From Public Plans to Global Solutions in Multiagent Planning

Jan Tožička(✉), Jan Jakubův, and Antonín Komenda

FEE, Department of Computer Science, Agent Technology Center,
Czech Technical University in Prague, Prague, Czech Republic
{jan.tozicka,jan.jakubuv,antonin.komenda}@agents.fel.cvut.cz

Abstract. Multiagent planning addresses the problem of coordinated sequential decision making of a team of cooperative agents. One possible approach to multiagent planning, which proved to be very efficient in practice, is to find an acceptable public plan. The approach works in two stages. At first, a public plan acceptable to all the involved agents is computed. Then, in the second stage, the public solution is extended to a global solution by filling in internal information by every agent. In the recently proposed distributed multiagent planner, the winner of the Competition of Distributed Multiagent Planners (CoDMAP 2015), this principle was utilized, however with unnecessary use of combination of both public and internal information for extension of the public solution.

In this work, we improve the planning algorithm by enhancements of the global solution reconstruction phase. We propose a new method of global solution reconstruction which increases efficiency by restriction to internal information. Additionally, we employ reduction techniques downsizing the input planning problem. Finally, we experimentally evaluate the resulting planner and prove its superiority when compared to the previous approach.

1 Introduction

Intelligent agents cooperatively solving a problem in a shared environment are required to coordinate their activity and preserve their local private knowledge. *Deterministic multiagent planning (DMAP)*, an established sub-area of the planning research, provides formal and practical tools to solve such problems.

The commonly used model for DMAP is MA-Strips [2] proposed by Brafman and Domshlak as a minimalistic extension of the classical planning model Strips [7]. MA-Strips begins with a set of cooperative agents, each capable of a different set of abilities described in the form of deterministic actions. The shared environment the agents act in is defined over a finite set of possible states, each state represented as a set of possibly holding facts. If a fact influences and/or is influenced only by a single agent, there is no need for the other agents to consider it, therefore it is *private* (or *internal*) for the given agent. Actions and states form a global transition system modeling the target planning problem. In order to execute an action, its preconditions have to be satisfied in the

M. Rovatsos et al. (Eds.): EUMAS 2015/AT 2015, LNAI 9571, pp. 21–33, 2016.
DOI: 10.1007/978-3-319-33509-4_2

current state of the environment. Conversely, after execution of an action, the environment is transformed into a new state according to effects of the action (under the close-world assumption). A solution of MA-STRIPS problem is an ordered sequence of agents' actions – a multiagent plan – which after execution transforms the environment from a predefined initial state to one of predefined goal states.

MA-STRIPS planning is domain independent, therefore the real-world motivation spans over a wide variety of problems [16], similarly to classical planning [14]. Representative examples, presented in our benchmarks are: the logistics domain modeling a heterogeneous fleet of vehicles transporting goods among predefined places; the rovers and satellites domains modeling teams of autonomous rovers and satellites conducing experiments around and on the surface of a distant planet; and a multi-robot variant of the classical Sokoban puzzle, where crates has to be pushed (pulls are not allowed) from their initial positions to predefined storage (goal) positions in a grid maze.

Following the historical development of classical planners, currently in DMAP, plan-space and state-space search techniques compete what approach is the more efficient. Our presented planning technique follows the principle of plan-space search to find a valid public plan, which can be consequently extended by local planning to a global solution of the planning problem. This principle was proposed in the first MA-STRIPS planner Planning First [17], using transformation of the plan-space search to a Distributed Constraint Satisfaction Problem, which solution represented the public plan. The transformation was superseded by Fabre et al. in [6] by public plans represented as Finite State Machines (FSMs). This idea inspired our line of work [11,19–21] on satisficing (i.e., a correct plan is sufficient, cf. optimal plan) DMAP by intersection of Planning State Machines (PSMs), which are FSMs representing compactly a set of plans of different agents.

DMAP problems are hard to solve. Particularly, planning of MA-STRIPS problems is exponential in (the tree-width of) the interaction graph among the agents [3] to the size of the planning problem. This inherent complexity cannot be in general tackled tractably (unless P = NP), therefore as in classical planning, we had to utilize automatically derived heuristics. In our recent work, we used the LAMA planner comprising forward-search planning heuristics (Fast-Forward [10] and LAMA landmarks [18]) to solve the local planning problems of particular agents. The local planning extends public plans towards a global solution. Additionally, we used the concept of planning landmarks to direct the plan-space search for the public plan in [19].

Besides the heuristics, which help to navigate the search, a special form of complexity can be reduced by an appropriate transformation of the planning problem. Such reductions can remove *accidental complexity* [8], i.e., superfluous complexity of planning caused by inappropriate formulation of problems. In classical planning, the most frequently used reduction technique is reachability analysis (e.g., in [9,10]), that is removal of a subset of actions which can be proved to be inapplicable during the search. Problem reduction by reachability analysis is one of transition system reduction techniques [1,4,5,8,13,15]. Recently, we have proposed first steps towards reductions for DMAP in [12].

In this paper, we propose a new method of global solution reconstruction, which increases efficiency by restriction to internal information. This combined with reductions for DMAP increases efficiency of the best performing distributed multiagent planner PSM-RVD [21] in the Competition of Distributed Multiagent Planners (CoDMAP 2015)[1]. On average, the improvement by proposed techniques in the coverage of solved problems is 13 %.

2 Multiagent Planning

This section provides a condensed formal prerequisites of multiagent planning based on MA-STRIPS formalism [2].

MA-STRIPS *planning problem* Π is a quadruple $\Pi = \langle F, \{\alpha_i\}_{i=1}^{n}, I, G\rangle$, where F is a set of facts, α_i is the set of actions of i-th agent[2], $I \subseteq F$ is an initial state, and $G \subseteq F$ is a set of goal facts. We define selector functions $\mathsf{facts}(\Pi)$, $\mathsf{agents}(\Pi)$, $\mathsf{init}(\Pi)$, and $\mathsf{goal}(\Pi)$ such that the following holds.

$$\Pi = \langle \mathsf{facts}(\Pi), \mathsf{agents}(\Pi), \mathsf{init}(\Pi), \mathsf{goal}(\Pi)\rangle$$

An *action* an agent can perform is a triple of subsets of $\mathsf{facts}(\Pi)$ called *preconditions*, *add effects*, *delete effects*. Selector functions $\mathsf{pre}(a)$, $\mathsf{add}(a)$, and $\mathsf{del}(a)$ are defined so that $a = \langle \mathsf{pre}(a), \mathsf{add}(a), \mathsf{del}(a)\rangle$.

A *planning state* s is a finite set of facts and we say that fact f *holds in* s, or that f *is valid in* s, iff $f \in s$. When $\mathsf{pre}(a) \subseteq s$ then *state progression* function γ is defined classically as $\gamma(s, a) = (s \setminus \mathsf{del}(a)) \cup \mathsf{add}(a)$.

In MA-STRIPS, out of computational or privacy concerns, each fact is classified either as *public* or as *internal*. A fact is *public* when it is mentioned by actions of at least two different agents. A fact is *internal for agent* α when it is not public but mentioned by some action of α. A fact is *relevant for* α when it is either public or internal for α. MA-STRIPS further extends this classification of facts to actions as follows. An action is *public* when it contains a public fact (as a precondition or effect), otherwise it is *internal*. An action from Π is *relevant for* α when it is either public or owned by (contained in) α.

We use $\mathsf{int\text{-}facts}(\alpha)$, $\mathsf{pub\text{-}facts}(\alpha)$, and $\mathsf{rel\text{-}facts}(\alpha)$ to denote in turn the set of internal, the set of public, and the set of relevant facts of agent α. Moreover, we write $\mathsf{pub\text{-}facts}(\Pi)$ to denote all the public facts of problem Π. We write $\mathsf{pub\text{-}actions}(\alpha)$ and $\mathsf{int\text{-}\ actions}(\alpha)$ to denote in turn the set of public, and the set of internal actions of agent α. Finally, we use $\mathsf{pub\text{-}actions}(\Pi)$ to denote all the public actions of all the agents in problem Π. The notation is summarized in Fig. 1.

In multiagent planning with external actions, a *local planning problem* is constructed for every agent α. Each local planning problem for α is a classical STRIPS problem where α has its own internal copy of the global state and where

[1] See http://agents.fel.cvut.cz/codmap for more info about CoDMAP'15.

[2] Note, that agents are defined only by their actions and thus α_i represents both the agent and the actions it can perform.

each agent is equipped with information about public actions of other agents called *external actions*. These local planning problems allow us to divide MA-STRIPS problem to several STRIPS problems which can be solved separately by a classical planner.

The *projection* $F \triangleright \alpha$ of set of facts $F \subseteq \mathsf{facts}(\Pi)$ to agent α is the restriction of F to the facts relevant for α, representing F as seen by α. The *projection* $a \triangleright \alpha$ of action a to agent α is obtained by restricting the facts in a to facts relevant for α, that is, hiding internal facts of other agents. The *public projection* $a \triangleright \star$ of action a is obtained by restricting the facts in a to public facts. Projections are extended to sets of actions element-wise.

A *local planning problem* $\Pi \triangleright \alpha$ of agent α, also called *projection of* Π to α, is a classical STRIPS problem containing all the actions of agent α together with external actions, that is, public projections other agents public actions. The local problem of α is defined only using the facts relevant for α. Formally,

$$\Pi \triangleright \alpha = \langle \mathsf{facts}(\Pi) \triangleright \alpha, \alpha \cup \mathsf{ext\text{-}actions}(\alpha), I \triangleright \alpha, G \rangle$$

where the set of external actions $\mathsf{ext\text{-}actions}(\alpha)$ is defined as follows.

$$\mathsf{ext\text{-}actions}(\alpha) = \bigcup_{\beta \neq \alpha} (\mathsf{pub\text{-}actions}(\beta) \triangleright \star)$$

In the above, β ranges over all the agents of Π. The set $\mathsf{ext\text{-}actions}(\alpha)$ can be equivalently described as $\mathsf{ext\text{-}actions}(\alpha) = (\mathsf{pub\text{-}actions}(\Pi) \setminus \alpha) \triangleright \star$. To simplify the presentation, we consider only problems with public goals and hence there is no need to restrict goal G.

3 Multiagent Planning by Plan Set Intersection

The previous section allows us to divide MA-STRIPS problem into several classical STRIPS local planning which can be solved separately by a classical planner. Recall that local planning problem of agent α contains all the actions of α together with α's external actions, that is, with projections of public actions of other agents. This section describe conditions which allow us to compute a solution of the original MA-STRIPS problem from solutions of local problems.

A *plan* π is a sequence of actions. The state progression function can then be iteratively extended to $\gamma^\star(s_0, \pi)$ defined on plans instead of actions. A *solution* of Π is a plan π whose execution transforms the initial state into a state in the set of goals, i.e. $\gamma^\star(I, \pi) \in G$. A *local solution* of agent α is a solution of $\Pi \triangleright \alpha$. Let $\mathsf{sols}(\Pi)$ denote the set of all the solutions of MA-STRIPS or STRIPS problem Π. A *public plan* σ is a sequence of public actions. The *public projection* $\pi \triangleright \star$ of plan π is the restriction of π to public actions. To avoid confusions possibly arising when two different actions have the same projection, we consider actions to have assigned unique ids which are preserved by projections. We omit ids from formal development in this work.

$\mathsf{facts}(a)$	$=$	$\mathsf{pre}(a) \cup \mathsf{add}(a) \cup \mathsf{del}(a)$	*facts of action a*
$\mathsf{facts}(\alpha)$	$=$	$\bigcup_{a\in\alpha} \mathsf{facts}(a)$	*facts of agent α*
$\mathsf{facts}(\Pi)$	$=$	$\bigcup_{\alpha\in\mathsf{agents}(\Pi)} \mathsf{facts}(\alpha)$	*all facts*
$\mathsf{actions}(\alpha)$	$=$	α	*actions of agent α*
$\mathsf{actions}(\Pi)$	$=$	$\bigcup_{\alpha\in\mathsf{agents}(\Pi)} \mathsf{actions}(\alpha)$	*all actions*
$\mathsf{pub\text{-}fact}(f)$	$\Leftrightarrow$	$\exists\alpha \neq \beta : \ f \in (\mathsf{facts}(\alpha) \cap \mathsf{facts}(\beta))$	*public fact (predicate)*
$\mathsf{pub\text{-}facts}(\Pi)$	$=$	$\{f \in \mathsf{facts}(\Pi) : \mathsf{pub\text{-}fact}(f)\}$	*all public facts*
$\mathsf{pub\text{-}action}(a)$	$\Leftrightarrow$	$(\mathsf{facts}(a) \cap \mathsf{pub\text{-}facts}(\Pi)) \neq \emptyset$	*public action (predicate)*
$\mathsf{pub\text{-}actions}(\Pi)$	$=$	$\{a \in \mathsf{actions}(\Pi) : \mathsf{pub\text{-}action}(a)\}$	*public actions*
$\mathsf{pub\text{-}facts}(\alpha)$	$=$	$\mathsf{facts}(\alpha) \cap \mathsf{pub\text{-}facts}(\Pi)$	*public facts of agent α*
$\mathsf{int\text{-}facts}(\alpha)$	$=$	$\mathsf{facts}(\alpha) \setminus \mathsf{pub\text{-}facts}(\Pi)$	*internal facts of agent α*
$\mathsf{rel\text{-}facts}(\alpha)$	$=$	$\mathsf{facts}(\alpha) \cup \mathsf{pub\text{-}facts}(\Pi)$	*relevant facts of agent α*
$\mathsf{pub\text{-}actions}(\alpha)$	$=$	$\mathsf{actions}(\alpha) \cap \mathsf{pub\text{-}actions}(\Pi)$	*public actions of agent α*
$\mathsf{int\text{-}actions}(\alpha)$	$=$	$\mathsf{actions}(\alpha) \setminus \mathsf{pub\text{-}actions}(\Pi)$	*internal actions of agent α*

Fig. 1. MA-Strips privacy classification of facts and actions of problem Π.

A public plan σ is *extensible* when there is $\pi \in \mathsf{sols}(\Pi)$ such that $\pi \triangleright \star = \sigma$. Extensible public plans are also called *public solutions*. Similarly, σ is α-*extensible* when there is $\pi \in \mathsf{sols}(\Pi \triangleright \alpha)$ such that $\pi \triangleright \star = \sigma$. Extensible public plans give us an order of public actions which is acceptable for all the agents. Thus extensible public plans are very close to solutions of Π and it is relatively easy to construct a solution of Π once we have an extensible public plan. That is why, in our previous work, the procedure of reconstruction of a global solution of Π from a public plan received little attention. In this work, we elaborate this procedure in detail.

The following Lemma [19] establishes the relationship between extensible and α-extensible plans. Its direct consequence is that to find a solution of Π it is enough to find a local solution $\pi_\alpha \in \mathsf{sols}(\Pi \triangleright \alpha)$ which is β-extensible for every agent β.

Lemma 1 **([19]).** *Public plan σ of Π is extensible if and only if σ is α-extensible for every agent α.*

Our previous multiagent planning algorithms [12,19,20] work in two-stages. In the first stage, a public solution is found, while, in the second stage, the public solution is extended to a global solution. Simple public solution search is described in Algorithm 1. Every agent executes the loop from Algorithm 1, possibly on a different machine. Every agent keeps generating new solutions of its local problem and stores public projections of local solutions set Φ_α. These sets are exchanged among all the agents so that every agent can compute their intersection Φ. Once the intersection Φ is non-empty, the algorithm terminates yielding Φ as the result. Hence Algorithm 1 yields a set of extensible public plans.

Algorithm 1. Public solution distributed search.

Function `MaPublicPlan`($\Pi \triangleright \alpha$) **is**
  $\Phi_\alpha \leftarrow \emptyset$;
  loop
    generate new $\pi_\alpha \in \mathsf{sols}(\Pi \triangleright \alpha)$;
    $\Phi_\alpha \leftarrow \Phi_\alpha \cup \{\pi_\alpha \triangleright \star\}$;
    exchange public plans Φ_β with other agents;
    $\Phi \leftarrow \bigcap_{\beta \in \mathsf{agents}(\Pi)} \Phi_\beta$;
    if $\Phi \neq \emptyset$ **then**
      return Φ;
    end
  end
end

Algorithm 2. Multiagent planning algorithm.

Function `MaPlan`(Π) **is**
  foreach $\alpha \in \mathsf{agents}(\Pi)$ **do**
    execute `MaPublicPlan`($\Pi \triangleright \alpha$); *// Algorithm 1*
  end
  $\Phi \leftarrow$ the result of `MaPublicPlan`($\Pi \triangleright \alpha$); *// of an arbitrary agent* α
  $\sigma \leftarrow$ any public solution from Φ;
  $\pi \leftarrow$ global solution reconstruction from σ; *// Sections 4 &5*
  return π;
end

Once an extensible public plan is found, it needs to be extended to a global solution. The reconstruction of a global solution of the original MA-STRIPS problem Π is described in details in the following sections. The overall procedure covering both public plan search and global solution reconstruction is depicted in Algorithm 2.

4 Global Solution from Local Solutions

This section summarizes methods of global solution reconstruction used in our previous work [12, 19, 20]. In multiagent planning algorithms based on the idea of plan set intersection sketched in Algorithm 2, every agent keeps generating local solutions until every agent generates a local solution with the same public projection as other agents. A global solution is then reconstructed from these local solutions utilizing Lemma 1. Its constructive proof [19, Lemma 1] suggests a method for reconstruction of a global solution from local solutions by their merging.

The above method can be used in situations when local solutions with an equal public projection were generated during the public plan search. Our successor planning algorithms [12, 20] can, however, arrive at a public solution σ

without generating local solutions with public projection σ. In [20], public projections of local solutions generated by individual agents are stored in structures called *Planning State Machines* (PSM). In some cases, plans stored by a PSM are combined together giving rise to new plans which were not explicitly generated. Moreover, analysis of dependencies of public actions [12] can yield a public solution without generating any local solution at all. Hence in these cases, a different approach needs to be used to reconstruct a global solution from σ.

Suppose we have a public plan σ and we know that σ is a public solution, that is, we know that σ is extensible. For every agent α, a local solution of $\Pi \rhd \alpha$ with public projection σ can be found by a method originally used to test α-extensibility [19]. For a public plan σ, we construct a classical STRIPS problem $\alpha \circ \sigma$ which contains public actions from σ together with α's internal actions. Public actions from σ are extended with special *mark facts* which ensure that every solution of $\alpha \circ \sigma$ contains all the actions from σ in the right order, possibly interleaved with α-internal actions. When $\sigma = \langle a_1, \ldots, a_k \rangle$, then we use $k + 1$ distinct mark facts $\{m_0, \ldots, m_k\}$. The meaning of fact m_i is that actions $a_1,\ldots,a_i$ has been used in the right order, and that action a_{i+1} should be used now. This behavior is achieved by adding m_{i-1} to the precondition and the delete effect of a_i, and by adding m_i to the add effect. For convenience, we define function $\mathsf{mark\text{-}act}(a, \mathit{from}, \mathit{to})$, which adds mark facts *from* and *to* to action a as follows.

$$\mathsf{mark\text{-}act}(a, \mathit{from}, \mathit{to}) = \langle \mathsf{pre}(a) \cup \{\mathit{from}\}, \mathsf{add}(a) \cup \{\mathit{to}\}, \mathsf{del}(a) \cup \{\mathit{from}\} \rangle$$

The following formally defines the STRIPS problem $\alpha \circ \sigma$. Note that the first mark m_0 is added to the initial state, and that m_k is added to the goal. The goal mark fact m_k ensures that all the actions were used. Also note that, as opposed to $\Pi \rhd \alpha$, problem $\alpha \circ \sigma$ does not contain external actions.

Definition 1. *Let* $\alpha \in \mathsf{agents}(\Pi)$ *and let* $\sigma = \langle a_1, \ldots, a_k \rangle$ *be a public plan. Let* $\mathsf{marks} = \{m_0, \ldots, m_k\}$ *be a set of facts distinct from* $\mathsf{facts}(\Pi)$*. The α-extensibility check of σ, denoted $\alpha \circ \sigma$, is the* STRIPS *problem* $\langle F_0, A_0, I_0, G_0 \rangle$ *where*

1. $F_0 = (\mathsf{facts}(\Pi) \rhd \alpha) \cup \mathsf{marks}$*, and*
2. $A_0 = \mathsf{int\text{-}\ actions}(\alpha) \cup \{\mathsf{mark\text{-}act}(a_i \rhd \alpha, m_{i-1}, m_i) : 0 < i \leq k\}$*, and*
3. $I_0 = (\mathsf{init}(\Pi) \rhd \alpha) \cup \{m_0\}$*, and*
4. $G_0 = G \cup \{m_k\}$*.*

The following lemma relates the STRIPS problem $\alpha \circ \sigma$ with α-extensibility of σ.

Lemma 2 ([19])**.** *Let* $\alpha \in \mathsf{agents}(\Pi)$ *and let σ be a public plan. Then σ is α-extensible iff* $\mathsf{sols}(\alpha \circ \sigma) \neq \emptyset$*.*

Suppose we have a public solution σ. It is easy to see that every solution of $\alpha \circ \sigma$ is also a local solution of $\Pi \rhd \alpha$, provided mark facts are removed. Hence problems $\alpha \circ \sigma$ can be used to generate local solutions with public projection σ. These local solutions can in turn be used to reconstruct a global solution as described above.

5 Global Solution from Public Solution

The previous section defines the α-extensibility check problem $\alpha \circ \sigma$ which can be used either to (1) verify that a public plan σ is α-extensible, or to (2) find a local solution of $\Pi \rhd \alpha$ with public projection σ. In this section we concentrate on task (2) in situations where an extensible public plan σ is given. The extensibility check problem $\alpha \circ \sigma$ contains public facts in public actions coming from σ. These public facts increase the complexity of planning task $\alpha \circ \sigma$. We propose an improved method for finding a local solution with a given public projection σ, provided we know that σ is (α-)extensible.

Every public solution σ is α-extensible and hence there is a solution π of $\Pi \rhd \alpha$ with public projection σ. Recall that π contains all the actions from σ possibly interleaved with α-internal actions. Public preconditions of public actions in π can not be affected by internal actions, and hence the public preconditions must be satisfied by actions coming from σ. Thus, when extending a public solution σ to a local solution, we can omit public facts and concentrate only on internal facts (public actions can have additional internal preconditions and effects).

Given a public solution σ, we define the *α-reconstruction problem*, denoted $\sigma \bullet \alpha$, similar to the α-extensibility check problem $\sigma \circ \alpha$. The only difference is that the reconstruction problem further restricts the facts to internal facts. Recall that α-projection ($\rhd$) is the restriction to the facts relevant for α. We define *internal α-projection* ($\blacktriangleright$) as the restriction to α-internal facts. For convenience, Fig. 2 summarizes definitions of different projections.

$F \rhd \alpha$	$= F \cap \mathsf{rel\text{-}facts}(\alpha)$	*facts α-projection*
$F \blacktriangleright \alpha$	$= F \cap \mathsf{int\text{-}facts}(\alpha)$	*facts internal α-projection*
$F \rhd \star$	$= F \cap \mathsf{pub\text{-}facts}(\Pi)$	*facts public projection*
$a \rhd \alpha$	$= \langle \mathsf{pre}(a) \rhd \alpha, \mathsf{add}(a) \rhd \alpha, \mathsf{del}(a) \rhd \alpha \rangle$	*action α-projection*
$a \blacktriangleright \alpha$	$= \langle \mathsf{pre}(a) \blacktriangleright \alpha, \mathsf{add}(a) \blacktriangleright \alpha, \mathsf{del}(a) \blacktriangleright \alpha \rangle$	*action internal α-projection*
$a \rhd \star$	$= \langle \mathsf{pre}(a) \rhd \star, \mathsf{add}(a) \rhd \star, \mathsf{del}(a) \rhd \star \rangle$	*action public projection*

Fig. 2. Different projections of facts and actions.

The α-reconstruction problem of σ is formally defined as follows. Note that only the last mark fact constitutes the goal because all the other goal facts are public.

Definition 2. *Let $\alpha \in \mathsf{agents}(\Pi)$ and let $\sigma = \langle a_1, \ldots, a_k \rangle$ be a public plan. Let $\mathsf{marks} = \{m_0, \ldots, m_k\}$ be a set of facts distinct from $\mathsf{facts}(\Pi)$. The α-reconstruction problem of σ, denoted $\alpha \bullet \sigma$, is the* Strips *problem $\langle F_0, A_0, I_0, G_0 \rangle$ where*

1. $F_0 = (\mathsf{facts}(\Pi) \blacktriangleright \alpha) \cup \mathsf{marks}$, *and*

2. $A_0 = \mathsf{int\text{-}\ actions}(\alpha) \cup \{\mathsf{mark\text{-}act}(a_i \blacktriangleright \alpha, m_{i-1}, m_i) : 0 < i \leq k\}$*, and*
3. $I_0 = (\mathsf{init}(\Pi) \blacktriangleright \alpha) \cup \{m_0\}$*, and*
4. $G_0 = \{m_k\}$.

The following lemma relates solutions of $\sigma \bullet \alpha$ with α-extensibility of σ. When compared with the similar result for the extensibility check problem $\sigma \circ \alpha$ (Lemma 2), only one implication can be proved.

Lemma 3. *Let* $\alpha \in \mathsf{agents}(\Pi)$ *and let* σ *be a public plan. If* σ *is* α*-extensible then* $\mathsf{sols}(\alpha \bullet \sigma) \neq \emptyset$*.*

Proof. By Lemma 2, there is $\pi \in \mathsf{sols}(\alpha \circ \sigma)$*. A solution of* $\alpha \bullet \sigma$ *can be obtained from* π *by internal* α*-restriction of actions (preserving mark facts).* □

There is a relationship between solutions of $\sigma \bullet \alpha$ and local solutions of agent α. When σ is α-extensible then every solution of $\sigma \bullet \alpha$ is also a solution of the local problem $\Pi \rhd \alpha$. Formally as follows.

Lemma 4. *Let* $\alpha \in \mathsf{agents}(\Pi)$ *and let* σ *be an* α*-extensible public plan. Then* $\mathsf{sols}(\sigma \bullet \alpha) \subseteq \mathsf{sols}(\Pi \rhd \alpha)$ *(up to the mark facts).*

Proof. Let $\pi \in \mathsf{sols}(\sigma \bullet \alpha)$*. Let* π' *be* π *with mark facts removed from the actions. Hence* π' *contains only actions from* $\Pi \rhd \alpha$*. Let* $I = \mathsf{init}(\Pi \rhd \alpha)$ *and let us prove that* $\gamma^\star(I, \pi')$ *is defined. Every* α*-internal precondition of every action from* π' *is satisfied because it was satisfied in* π*, and other actions in* π' *do not affect* α*-internal facts. Hence, it is enough to prove that public preconditions are satisfied. It, however, follows from extensibility of* σ*. Finally,* $\gamma^\star(I, \pi') \subseteq \mathsf{goal}(\Pi \rhd \alpha)$ *because all the goals are public and* σ *is* α*-extensible.* □

Hence α-reconstruction problems of σ can help us to construct local solutions with public projection σ. These local solutions can in turn be used to reconstruct a global solution. The following theorem put the pieces together, that is, it provides a constructive way to construct a global solution from a public solution. Given a public solution σ, a local solution with public projection σ is computed using α-reconstruction by every agent α. All these local solutions contain the same public actions given by σ. These public actions naturally split the plans into parts which are merged together giving rise to a global solution.

Theorem 1. *Let* $\mathsf{agents}(\Pi) = \{\alpha^1, \ldots, \alpha^n\}$ *and let* $\sigma = \langle a_1, \ldots, a_k \rangle$ *be an extensible plan of* Π*. Then*

1. there is $\pi^i \in \mathsf{sols}(\alpha^i \bullet \sigma)$ *for every* $0 < i \leq n$*, and*
2. for every π^i*, there are* $\pi^i_1, \ldots, \pi^i_k$ *such that* π^i *can be written as*

$$\pi^i = \pi^i_1 \cdot \langle a'_1 \rangle \cdot \cdots \cdot \pi^i_k \cdot \langle a'_k \rangle \quad \text{where} \quad a'_j = \mathsf{mark\text{-}act}(a_j \blacktriangleright \alpha^i) \quad (\text{for } 0 < j \leq k)$$

3. and, plan

$$\pi = \pi_1 \cdot \langle a_1 \rangle \cdot \ \cdots \ \cdot \pi_k \cdot \langle a_k \rangle \quad \text{where} \quad \pi_j = \pi^1_j \cdot \ \cdots \ \cdot \pi^n_j \quad (\text{for } 0 < j \leq k)$$

is a solution of Π*.*

Proof. *Claim (1) is by Lemmas 1 and 3. For Claim (2), let $\pi^i \in \mathsf{sols}(\alpha \bullet \sigma)$ be given. Due to the marks, π^i contains all the actions from σ in the right order. We can consider (α^i-projection of) a_k to be the last action of π^i because it is the only action fulfilling the goal of $\alpha^i \bullet \sigma$. Hence π^i_j are simply the (α^i-internal) actions between (projections of) a_j and a_{j-1} in π^i. For Claim (3), let $I = \mathsf{init}(\Pi)$. Now $\gamma^\star(I, \pi)$ is defined following the same arguments as in the Proof of Lemma 4. Finally, $\gamma^\star(I, \pi) \subseteq \mathsf{goal}(\Pi)$ because all goals are public and σ is extensible.* □

6 Experiments

This section experimentally evaluates the impact of improved global solution reconstruction. In order to undertake the experiments, we use our Psm-rvd planner [21] which performs public solution search utilizing Planning State Machines (PSM) [20] enhanced with analysis of internal dependencies of public actions [12]. In the Competition of Distributed Multiagent Planners (CoDMAP 2015), Psm-rvd solved 180 problems out of total 240 problems within the 30 min time limit, achieving the best results in the *distributed* track.

Planner Psm-rvd submitted to CoDMAP 2015 implemented global solution reconstruction using *extensibility check problems* (∘) described in Sect. 4. To undertake the experiments, we have implemented global solution reconstruction using *reconstruction problems* (•) from Sect. 5. We use 220 benchmark problems from 11 domains[3], with the time limit of 5 min.

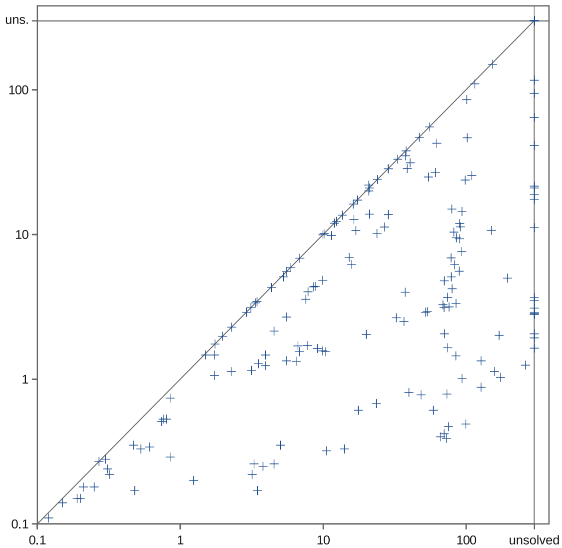

Fig. 3. Impact of improved global solution reconstruction on the runtime of Psm-rvd on CoDMAP benchmark problems (logarithmic scales).

[3] The last *Wireless* domain is not supported by the planner parser.

Table 1. Impact of improved global solution reconstruction on the coverage of PSM-RVD on CoDMAP benchmark problems.

Domain	Solved problems [count]		Reconstruction phase [avg. % of runtime]	
	(○)	(●)	(○)	(●)
Blocksworld (20)	18	**19**	92 %	**47 %**
Depot (20)	8	**15**	86 %	**24 %**
Driverlog (20)	19	**20**	18 %	**14 %**
Elevators (20)	7	**12**	3.0 %	**2.4 %**
Logistics (20)	12	**18**	86 %	**35 %**
Rovers (20)	6	**7**	2.3 %	**1.3 %**
Satellites (20)	8	8	4.1 %	4.1 %
Taxi (20)	20	20	82.8 %	**35.9 %**
Sokoban (20)	15	15	83 %	**29 %**
Woodworking (20)	18	18	35 %	**20 %**
Zenotravel (20)	10	10	5.5 %	**4.4 %**
Total (220)	141	**162**	45.2 %	**19.7 %**

Figure 3 evaluates the impact of improved global solution reconstruction on CoDMAP benchmark problems. For each problem, a point is drawn at the position corresponding to the runtime with extensibility check problems ○ (x-coordinate) and the runtime with reconstruction problems ● (y-coordinate). Hence points below a diagonal constitute improvements. We can see that for all the problems, the runtime was either improved or unchanged.

Table 1 shows (1) the impact of improved global solution reconstruction on the coverage of solved problems, and (2) the impact on a relative length of global solution reconstruction phase. The relative length of global reconstruction phase is measured in the percentage of runtime. We can see that the relative length of a reconstruction phase was shorten even in the cases where it has no effect on total coverage.

7 Conclusions

We have formally and practically enhanced the winning planner of recent competition of distributed and multiagent planners. The global solution reconstruction phase was limited to use only of private facts, which increased efficiency of the algorithm. We have formally proved that such narrowing preserves extensibility of the plan, and therefore the soundness and completeness of the planner. Additionally, we have used recently proposed static reductions of the planning problems for multiagent planning. The practical experiments show improvements of coverage of solved problems by 13 % and strong domination over to the original variant of the planner.

Acknowledgments. This research was supported by the Czech Science Foundation (grant no. 13-22125S) and by the Ministry of Education of the Czech Republic within the SGS project (no. SGS13/211/OHK3/3T/13). Access to computing and storage facilities owned by parties and projects contributing to the National Grid Infrastructure MetaCentrum, provided under the program "Projects of Large Infrastructure for Research, Development, and Innovations" (LM2010005), is greatly appreciated.

References

1. Bäckström, C., Jonsson, A., Jonsson, P.: Macros, reactive plans and compact representations. In: Raedt, L.D., Bessière, C., Dubois, D.,Doherty, P., Frasconi, P., Heintz, F., Lucas, P.J.F. (eds.) ECAI 2012. Frontiers in Artificial Intelligence and Applications, vol. 242, pp. 85–90. IOS Press (2012). http://dx.doi.org/10.3233/978-1-61499-098-7-85
2. Brafman, R., Domshlak, C.: From one to many: planning for loosely coupled multi-agent systems. In: Proceedings of ICAPS 2008, vol. 8, pp. 28–35 (2008)
3. Brafman, R.I., Domshlak, C.: From one to many: planning for loosely coupled multi-agent systems. In: Proceedings of ICAPS 2008, pp. 28–35 (2008)
4. Chen, Y., Yao, G.: Completeness and optimality preserving reduction for planning. In: Proceedings of 21st IJCAI, pp. 1659–1664 (2009)
5. Coles, A., Coles, A.: Completeness-preserving pruning for optimal planning. In: Proceedings of 19th ECAI, pp. 965–966 (2010)
6. Fabre, E., Jezequel, L., Haslum, P., Thiébaux, S.: Cost-optimal factored planning: promises and pitfalls. In: Proceedings of the 20th International Conference on Automated Planning and Scheduling, ICAPPS 2010, Toronto, Ontario, Canada, May 12–16, 2010, pp. 65–72 (2010)
7. Fikes, R., Nilsson, N.: STRIPS: a new approach to the application of theorem proving to problem solving. In: Proceedings of the 2nd International Joint Conference on Artificial Intelligence, pp. 608–620 (1971)
8. Haslum, P.: Reducing accidental complexity in planning problems. In: Proceedings of 20th IJCAI, pp. 1898–1903 (2007)
9. Helmert, M.: The fast downward planning system. J. Artif. Intell. Res. **26**, 191–246 (2006)
10. Hoffmann, J., Nebel, B.: The FF planning system: fast plan generation through heuristic search. JAIR **14**, 253–302 (2001)
11. Jakubův, J., Tožička, J., Komenda, A.: Multiagent planning by planset intersection and plan verification. In: ICAART 2015 - Proceedings of the International Conference on Agents and Artificial Intelligence, Lisbon, Portugal, 10–12 January, 2015, vol. 2, pp. 173–182 (2015)
12. Jakubův, J., Tožička, J., Komenda, A.: On internally dependent public actions in multiagent planning. In: Proceedings of 3rd DMAP Workshop (ICAPS 2015) (2015)
13. Jonsson, A.: The role of macros in tractable planning. J. Artif. Intell. Res. **36**, 471–511 (2009)
14. Nau, D., Ghallab, M., Traverso, P.: Automated Planning: Theory & Practice. Morgan Kaufmann Publishers Inc., San Francisco (2004)
15. Nissim, R., Apsel, U., Brafman, R.: Tunneling and decomposition-based state reduction for optimal planning. In: Proceedings of 20th ECAI, pp. 624–629 (2012). http://www.cs.bgu.ac.il/~apsel/files/decomp_ecai12.pdf

16. Nissim, R., Brafman, R.: Distributed heuristic forward search for multi-agent planning. JAIR **51**, 293–332 (2014)
17. Nissim, R., Brafman, R.I., Domshlak, C.: A general, fully distributed multi-agent planning algorithm. In: Proceedings of AAMAS, pp. 1323–1330 (2010)
18. Richter, S., Westphal, M.: The lama planner: guiding cost-based anytime planning with landmarks. J. Artif. Intell. Res. **39**(1), 127–177 (2010)
19. Tožička, J., Jakubův, J., Durkota, K., Komenda, A., Pechoucek, M.: Multiagent planning supported by plan diversity metrics and landmark actions. In: ICAART 2014 - Proceedings of the 6th International Conference on Agents and Artificial Intelligence, ESEO, Angers, Loire Valley, France, 6–8 March, 2014, vol. 1, pp. 178–189 (2014). http://dx.doi.org/10.5220/0004918701780189
20. Tožička, J., Jakubův, J., Komenda, A.: Generating multi-agent plans by distributed intersection of finite state machines. In: ECAI 2014 - 21st European Conference on Artificial Intelligence, 18–22 August 2014, Prague, Czech Republic - Including Prestigious Applications of Intelligent Systems (PAIS 2014), pp. 1111–1112 (2014)
21. Tožička, J., Jakubův, J., Komenda, A.: PSM-based planners description for CoDMAP 2015 competition. In: Proceedings of the Competition of Distributed and Multi-agent Planners (CoDMAP 2015), pp. 29–32 (2015)

http://www.springer.com/978-3-319-33508-7

Multi-Agent Systems and Agreement Technologies
13th European Conference, EUMAS 2015, and Third International Conference, AT 2015, Athens, Greece, December 17-18, 2015, Revised Selected Papers
Rovatsos, M.; Vouros, G.A.; Julian, V. (Eds.)
2016, XX, 474 p. 151 illus., Softcover
ISBN: 978-3-319-33508-7