

# Universal and Uniform Action Reversibility

Lukáš Chrpa<sup>1</sup>, Wolfgang Faber<sup>2</sup>, Michael Morak<sup>2</sup>

<sup>1</sup>Faculty of Electrical Engineering, Czech Technical University in Prague

<sup>2</sup>Department of Artificial Intelligence and Cybersecurity, University of Klagenfurt

chrpaluk@fel.cvut.cz, {wolfgang.faber,michael.morak}@aau.at

## Abstract

The problem of action reversibility studies whether effects of a given action can be reversed (or undone) by a sequence of (other) actions. For example, actions whose effects can be reversed cannot lead to dead-ends. In the usual settings, the problem of action reversibility is PSPACE-complete, that is, as hard as deciding plan existence. In this paper, we focus on subclasses of the action reversibility problem, *universal* and *uniform* action reversibility, where the former considers all states in which the action in question is applicable, while the latter requires a single reverting action sequence, independent of the considered states. Specifically, we study the relations between projection abstractions and the subclasses of the action reversibility problem and we show that universal uniform reversibility of a given action can be decided on projection consisting of only the variables present in the schema of the action in question.

## 1 Introduction

Traditionally, Automated Planning deals with the problem of generating a sequence of actions that transforms an initial state of the environment to some goal state (Ghallab, Nau, and Traverso 2004; Ghallab, Nau, and Traverso 2016). One interesting question is whether the effects of an action are reversible (by other actions), or in other words, whether the action effects can be undone. Notions of reversibility have been investigated by, e.g., Eiter, Erdem, and Faber (2008), Daum et al. (2016) and Morak et al. (2020).

Studying action reversibility is important for several reasons. Intuitively, reversible actions cannot lead to dead-end states from which the goal state is no longer reachable. Concerning online planning, we can observe that applying reversible actions is safe and hence we might not need to explicitly provide information about safe states of the environment (Cserna et al. 2018). In non-deterministic planning, action reversibility may allow us to recover from undesirable effects of a non-deterministic action, which is important for efficient computation of strong (cyclic) plans (Carmacho, Muise, and McIlraith 2016). Action reversibility can be useful while reasoning in more complex structures such as Agent Planning Programs (De Giacomo et al. 2016), which represent networks of planning tasks where a goal state of one task is an initial state of another, as preferring reversible actions for solving particular planning tasks

mitigates the risk of dead-ends (De Giacomo et al. 2016; Chrpa, Lipovetzky, and Sardiña 2017).

Eiter, Erdem, and Faber (2008) introduced the concept of *reverse plans*, which corresponds to uniform reversibility in this paper. Morak et al. (2020) presented a general framework concerning the problem of action reversibility. One can investigate action reversibility on a subset of states that can be described by a logical formula and/or investigate whether a single “reverse plan” undoes effects of the action applied in these states. If the “single reverse plan” property is satisfied, we talk about *uniform* action reversibility. If (uniform) action reversibility applies to all states, then we talk about *universal* (uniform) action reversibility. It has been shown that the problem of (uniform) action reversibility is PSPACE-hard (Morak et al. 2020), and therefore at least as hard as the planning problem itself.

In this paper, we focus on the notions of universal and uniform action reversibility and investigate several interesting properties arising from their definitions. We show that an action  $a$  can only be universally uniformly reversible if the reverse plan contains only variables present in the precondition of action  $a$ . Hence, such reverse plans can be found in an abstracted state space which is a projection of the original state space to the set of variables present in the precondition of the action to be reversed. Also, we show that if there is no path to the state in which the action was applied in such a projection abstraction, then the action is not universally uniformly reversible, that is, its effects cannot be undone for at least one state. We also show that given a formula  $\phi$  specifying a subset of states (as a conjunction of literals), we can leverage universal uniform reversibility (of a dummy action whose precondition represents  $\phi$ ) to determine uniform  $\phi$ -reversibility.

Even though universal uniform action reversibility is a very specific subclass of action reversibility, it can be commonly found in existing planning domains (e.g. loading or unloading in Logistics, moving blocks in BlocksWorld, moving parts of Rubik’s cube). Daum et al. (2016) address a more general problem than uniform action reversibility by means of contingent planning, which is, in general, EXPSPACE-hard. Therefore, our concept of universal uniform action reversibility poses an alternative that is computationally cheaper.

## 2 Classical Planning

Classical planning seeks to find a sequence of actions that transforms some initial state to some goal state (Ghallab, Nau, and Traverso 2004). We consider the Finite Domain Representation (FDR) of planning tasks (Helmert 2009).

Let  $V$  be a set of *variables* where each variable  $v \in V$  is associated with its domain  $D(v)$ . An *assignment* of a variable  $v \in V$  is a pair  $(v, val)$ , with value  $val \in D(v)$ . Hereafter, an assignment of a variable is also called a *fact*. A (partial) *variable assignment*  $p$  over  $V$  is a set of assignments of individual variables from  $V$ , where each variable is assigned at most once,  $vars(p)$  is the set of all variables in  $p$  and  $p[v]$  represents the value of  $v$  in  $p$ . A *state* is a complete variable assignment (over  $V$ ). We say that a (partial) variable assignment  $q$  *holds* in a (partial) variable assignment  $p$ , denoted as  $p \models q$ , iff  $vars(q) \subseteq vars(p)$  and for each  $v \in vars(q)$  it is the case that  $q[v] = p[v]$ . A *formula* is a propositional formula over facts.

An *action* is a pair  $a = (pre(a), eff(a))$ , where  $pre(a)$  is a partial variable assignment representing  $a$ 's precondition and  $eff(a)$  is a partial variable assignment representing  $a$ 's effects. We say that an action  $a$  is *applicable* in state  $s$  iff  $s \models pre(a)$ . The *result* of applying  $a$  in  $s$ , denoted as  $\gamma(s, a)$ , is a state  $s'$  such that for each variable  $v \in V$ ,  $s'[v] = eff(a)[v]$  if  $v \in vars(eff(a))$  while  $s'[v] = s[v]$  otherwise. If  $a$  is not applicable in  $s$ ,  $\gamma(s, a)$  is undefined. Note that we can generalise  $\gamma$  for action sequences, i.e.,  $\gamma(s, \langle a_1, a_2, \dots, a_n \rangle) = \gamma(\gamma(s, a_1), \langle a_2, \dots, a_n \rangle)$ .

A *planning domain* is a pair  $\mathcal{D} = (V, A)$  and a *planning task* is a triple  $\mathcal{P} = (\mathcal{D}, I, G)$ , where  $V$  is a set of variables,  $A$  a set of actions,  $I$  a complete variable assignment representing the *initial state* and  $G$  a partial variable assignment representing the *goal*. A *plan* (for the planning task) is a sequence of actions  $\pi = \langle a_1, \dots, a_n \rangle$  whose application in  $I$  results in a state satisfying  $G$ , i.e.,  $\gamma(I, \pi) \models G$ .

Note that by  $vars(a)$  we represent the set of variables associated with an action  $a$ , i.e.,  $vars(a) = vars(pre(a)) \cup vars(eff(a))$ . We denote as  $ha(a)$  a (partial) variable assignment representing values of variables that must *hold after applying* an action  $a$ , i.e., for each  $v \in vars(eff(a))$  :  $ha(a)[v] = eff(a)[v]$  and for each  $v' \in vars(pre(a)) \setminus vars(eff(a))$  :  $ha(a)[v'] = pre(a)[v']$ . Note that for variables not associated with  $a$ ,  $ha(a)$  is undefined.

### 2.1 Projection Abstraction

Abstraction is a technique for simplifying the state space into a simpler homomorphic one, which can be used, for instance, for computing admissible heuristics (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007).

Initially, we define the notion of *state space* for a planning domain  $\mathcal{D}$  as an edge-labelled directed graph  $S = (S, E)$ , where  $S$  is the set of states over  $V$  (all possible complete variable assignments) and  $E = \{(s, a, s') \mid s' \text{ is the result of applying } a \text{ in } s\}$ . We can abstract the state-space for a planning domain  $\mathcal{D}$  by projecting it to a subset of variables  $V' \subset V$ . We say that  $\Pi_{V'} : S \rightarrow S_{V'}$  is a *projection mapping*, where for each  $s \in S$  it is the case that  $\Pi_{V'}(s) = s_{V'}$  with  $s_{V'}[v] = s[v]$  for each  $v \in V'$

---

### Algorithm 1 Uniform $\phi$ -reversibility of an action $a$ .

---

**Input:** planning domain  $\mathcal{D} = (V, A)$ , an action  $a \in A$   
**Output:** formula  $\phi$ ,  $\phi$ -reverse plan  $\pi$

- 1:  $I \leftarrow pre(a), S \leftarrow \text{undefined}, \pi = \langle \rangle$
- 2:  $\forall v \in vars(ha(a)) : S[v] \leftarrow ha(a)[v]$
- 3: **while**  $S \not\models I$  **do**
- 4:   non-deterministically select  $a' \in A$  such that  $\nexists v \in vars(pre(a')) : pre(a')[v] \neq S[v]$
- 5:   **if**  $a'$  does not exist **then**
- 6:     **return**  $\perp, \langle \rangle$
- 7:   **end if**
- 8:    $\forall v \in vars(pre(a')) \setminus vars(I) : I[v] \leftarrow pre(a')[v]$
- 9:    $\forall v \in vars(ha(a')) : S[v] \leftarrow ha(a')[v]$
- 10:    $\pi = \pi \cdot a'$
- 11: **end while**
- 12:  $\phi = \bigwedge_{S[v]=val} (v, val)$
- 13: **return**  $\phi, \pi$

---

(for variables not present in  $V'$ ,  $s_{V'}$  is undefined). Then, we define a *projection abstraction* for  $\mathcal{D}$  and  $V'$  ( $V' \subseteq V$ ) as  $S_{V'} = (S_{V'}, E_{V'})$ , where  $S_{V'} = \{\Pi_{V'}(s) \mid s \in S\}$  and  $E_{V'} = \{(\Pi_{V'}(s), a, \Pi_{V'}(s')) \mid (s, a, s') \in E\}$ . We define  $\gamma_{V'}$  such that  $\gamma_{V'}(s_{V'}, a) = s'_{V'}$  iff  $(s_{V'}, a, s'_{V'}) \in E_{V'}$ . Note that each projection abstraction is homomorphic.

## 3 Action Reversibility

As intuition suggests, an action is reversible if there is a way to reverse all its effects. When formally defining this concept, we also need to take several other factors into account; in particular, whether the action is always reversible or only in a subset of states represented by a formula, or whether one reverse plan applies for all states. Note that our definitions of action reversibility depend only on the domain, not on a specific planning task. The following definition is adapted from (Morak et al. 2020). For universal reversibility we also include a formula in order to allow for relating to the non-universal case: if the set of states defined by the formula is large enough to contain all states in which the action is applicable, we have universal reversibility.

**Definition 1.** Let  $\mathcal{D} = (V, A)$  be a planning domain. An action  $a \in A$  is  $\phi$ -**reversible** for a formula  $\phi$  iff for each state  $s$  where  $a$  is applicable and  $s \models \phi$  there exists a sequence of actions  $\pi = \langle a_1, \dots, a_k \rangle$ , where  $a_1, \dots, a_k \in A$ , such that  $s = \gamma(\gamma(s, a), \pi)$ . Action  $a \in A$  is **uniformly  $\phi$ -reversible** for a formula  $\phi$  iff there exists a sequence of actions  $\pi = \langle a_1, \dots, a_k \rangle$ , where  $a_1, \dots, a_k \in A$  such that for each state  $s$ , where  $a$  is applicable in  $s$  and  $s \models \phi$ ,  $s = \gamma(\gamma(s, a), \pi)$ . We call  $\pi'$  a  **$\phi$ -reverse plan** for  $a$ . Action  $a \in A$  is **universally (uniformly) reversible** iff  $a$  is (uniformly)  $\phi$ -reversible for a formula  $\phi$  and for each state  $s$  over  $V$  in which  $a$  is applicable we have  $s \models \phi$ . In the uniform case,  $\pi'$  is called a **reverse plan** for  $a$ .

Analogously, we can define action irreversibility which says that there is no way to undo the effects of the action.

**Definition 2.** Let  $\mathcal{D} = (V, A)$  be a planning domain. An action  $a \in A$  is  $\phi$ -**irreversible** for a formula  $\phi$  iff for each

state  $s$  where  $s \models \phi$  and  $a$  is applicable there does not exist a sequence of actions  $\pi = \langle a_1, \dots, a_k \rangle$  with  $a_1, \dots, a_k \in A$  such that  $s = \gamma(\gamma(s, a), \pi)$ . Action  $a \in A$  is **universally irreversible** iff  $a$  is  $\phi$ -irreversible for a formula  $\phi$  and for each state  $s$  over  $V$  in which  $a$  is applicable we have  $s \models \phi$ .

Algorithm 1 is an adaptation of the algorithm presented by Morak et al. (2020) for the FDR formalism. The algorithm is inspired by plan generation in conformant planning (Grastien and Scala 2017). The idea is to keep the partial variable assignment  $I$  which has to hold prior to the application of  $a$  and the partial variable assignment  $S$  referring to the evolution of partial states during reverse plan generation. Finally, the formula  $\phi$  is generated as a conjunction of literals representing variable assignments from  $S$  for which at this stage it is the case that  $S \models I$ . The outputs  $\phi, \pi$  represent that  $a$  is uniformly  $\phi$ -reversible with  $\pi$  as the reverse plan. If  $\perp, \langle \rangle$  is returned, then  $a$  is universally irreversible.

It can be observed that the more iterations the algorithm performs, the larger the set of defined variables in a partial state  $S$  becomes. In other words, a variable defined in  $S$  cannot become undefined in later iterations of the algorithm. If the “to be reversed” action  $a$  is in normal form, i.e., variables present in its effects are also present in its precondition, the following theorem shows that existence of a reverse plan in which every action consists only of variables present in the precondition of  $a$  implies universal uniform reversibility of  $a$ . We can demonstrate that the opposite implication also holds if the domain for each variable defined in the planning domain has at least two values.

**Theorem 3.** *Let  $\mathcal{D} = (V, A)$  be a planning domain, where for each  $v \in V$  it is the case that  $|D(v)| \geq 2$ , and  $a \in A$  be an action such that  $\text{vars}(\text{eff}(a)) \subseteq \text{vars}(\text{pre}(a))$ . The action  $a$  is universally uniformly reversible if and only if there exists a reverse plan for  $a$ ,  $\pi = \langle a_1, \dots, a_k \rangle$ , such that  $\forall i \in \{1, \dots, k\} : \text{vars}(a_i) \subseteq \text{vars}(\text{pre}(a))$ .*

*Proof.* **The “if” part:**

Without loss of generality let us assume that  $\phi, \pi$  is the output of Algorithm 1 with  $\mathcal{D}$  and  $a$  as its inputs. We can observe that initially  $\text{vars}(S) = \text{vars}(\text{pre}(a))$ . Then, given the theorem assumption it is the case for each action  $a'$  selected in Line 4 that  $\text{vars}(a') \subseteq \text{vars}(\text{pre}(a))$ . Hence, after updating  $S$  in Line 9, it still holds that  $\text{vars}(S) = \text{vars}(\text{pre}(a))$ . Also,  $\text{vars}(\text{pre}(a')) \setminus \text{vars}(I) = \emptyset$  holds in each iteration and thus  $I = \text{pre}(a)$  remains unchanged. The while loop terminates only if  $S \models I$  which in this case happens only when  $S = I = \text{pre}(a)$ . Hence,  $\text{pre}(a) \models \phi$ , implying universal uniform reversibility of  $a$ .

**The “only if part”:**

We will prove the “only if part” by contradiction. Note that if  $\text{pre}(a) = V$ , the claim trivially holds. Without loss of generality let us assume that  $v \in V \setminus \text{pre}(a)$ . Since  $a$  is universally uniformly reversible by the assumption, there exists a reverse plan  $\pi = \langle a_1, \dots, a_k \rangle$ . Let us assume that there exists  $a_i \in \pi$  such that  $v \in \text{vars}(a_i)$ . Hence also  $v \in \text{vars}(ha(a))$  which implies that  $S[v]$  becomes defined after  $a_i$  is processed in Algorithm 1. Without loss of generality we assume that after the while loop of Algorithm 1 terminates,  $S[v] = \text{val}$  and  $(v, \text{val})$  becomes a part of  $\phi$ , which is

a conjunction of defined variable assignments from  $S$ . Since  $|D(v)| \geq 2$ , there exists  $\text{val}' \in D(v)$  such that  $\text{val}' \neq \text{val}$ . Hence, there exists a state  $s'$  such that  $s' \models \text{pre}(a)$  and  $s'[v] = \text{val}'$ . We can also observe that  $s' \not\models \phi$  which however contradicts with the assumption that  $a$  is universally uniformly reversible (see Def. 1).  $\square$

It should be noted that the “if” part of Theorem 3 can be proven even without assuming that the size of domain of each variable is at least two (i.e.,  $|D(v)| \geq 2$  for each  $v \in V$ ). The claim immediately follows from the above proof.

## 4 Action Reversibility and Abstractions

Projection abstraction in (classical) planning is used for computing admissible heuristics, as solving abstracted problems is much simpler (Edelkamp 2001; Helmert, Haslum, and Hoffmann 2007). With regards to the problem of action reversibility or irreversibility, projection abstractions can also be leveraged.

It can be straightforwardly observed that for a given planning domain  $\mathcal{D} = (V, A)$  and the projection abstraction  $\mathcal{S}_{\text{vars}(\text{pre}(a))}$  there is exactly one state in which  $a$  is applicable, i.e., there is exactly one edge labelled  $a$ . For the sake of simplicity, the notation  $\mathcal{S}_a = (S_a, E_a)$  will be equivalent to  $\mathcal{S}_{\text{vars}(\text{pre}(a))} = (S_{\text{vars}(\text{pre}(a))}, E_{\text{vars}(\text{pre}(a))})$ .

With the projection abstraction, we can simplify the computation of universal uniform reversibility as follows:

**Proposition 4.** *Let  $\mathcal{D}$ ,  $a$  be defined as in Theorem 3,  $\mathcal{S}_a = (S_a, E_a)$  be the projection abstraction for  $\mathcal{D}$  and  $\text{vars}(\text{pre}(a))$ , and  $(s, a, s') \in E_a$  be the only edge labelled  $a$ . There exists a path from  $s'$  to  $s$  in  $\mathcal{S}_a$  where all labels  $a'$  satisfy  $\text{vars}(a') \subseteq \text{vars}(\text{pre}(a))$  iff  $a$  is universally uniformly reversible.*

*Proof.* Since the actions on the path contain only variables being part of the projection abstraction, no information is abstracted out. We can observe that the abstract states visited along the path correspond with the evolution of the partial states  $S$  in Algorithm 1. Hence, the sequence of actions being the labels on the path forms a reverse plan for  $a$ .  $\square$

Proposition 4 allows us to restrict the domain to only variables from  $\text{vars}(\text{pre}(a))$  and to actions operating on them, specializing Algorithm 1 to universal uniform reversibility.

Projection abstraction can also uncover cases of universal action irreversibility as stated in the following proposition.

**Proposition 5.** *Let  $\mathcal{D} = (V, A)$  be a planning domain and  $a \in A$  be an action. Let  $\mathcal{S}_a = (S_a, E_a)$  be the projection abstraction for  $\mathcal{D}$  and  $\text{vars}(\text{pre}(a))$  and  $(s, a, s') \in E_a$  be the (only) edge labelled  $a$ . If there is no path from  $s'$  to  $s$  in  $\mathcal{S}_a$ , then  $a$  is universally irreversible.*

*Proof.* Since  $\mathcal{S}_a$  is homomorphic to the state space  $\mathcal{S}$  for  $\mathcal{D}$ , if there is no path from  $s'$  to  $s$  in  $\mathcal{S}_a$ , then there is no path from  $s'_*$  to  $s_*$  in  $\mathcal{S}$  for all pairs  $s'_*, s_*$  such that  $\Pi_a(s'_*) = s'$  and  $\Pi_a(s_*) = s$  ( $\Pi_a$  is the projection mapping from the set of states in  $\mathcal{S}$  to  $\mathcal{S}_a$ ).  $\square$

For example, in a variant of the Logistics domain we have trucks that can move from one location to another if connected by a one-way road, and packages that can be at some location or inside the truck (by loading or unloading packages). To determine universal uniform reversibility of the  $\text{drive}(A,B)$  action, which changes the value of  $\text{at-truck}$  from  $A$  to  $B$ , we can leverage Proposition 4 (and also Theorem 3) by projecting the state space to the values of  $\text{at-truck}$  and considering only actions over these variables, i.e., only the  $\text{drive}$  actions (disregarding the load or unload actions). Hence, if there is a path from  $\text{at-truck}=B$  to  $\text{at-truck}=A$ , e.g., via  $\text{at-truck}=C$ , then  $\text{drive}(A,B)$  is universally uniformly reversible (e.g.,  $\text{drive}(B,C)$ ,  $\text{drive}(C,A)$  is a reverse plan). If there is no path from  $\text{at-truck}=B$  to  $\text{at-truck}=A$ , then  $\text{drive}(A,B)$  is universally irreversible by Proposition 5.

## 5 Uniform Reversibility

Although the definition of (uniform)  $\phi$ -reversibility (Def. 1) allows us to consider states satisfying  $\phi$  in which the action in question is not applicable, we will, in this section, focus on formulas  $\phi$  such that only states in which the action in question is applicable satisfy  $\phi$ . This also correlates with the output of Algorithm 1. That said, for an action  $a$  we will consider formulas  $\phi$  that are conjunctions of positive literals representing variable assignments with  $\phi \models \text{pre}(a)$ .

According to the above specification of  $\phi$ , the action is universally (uniformly) reversible iff  $a$  is (uniformly)  $\phi$ -reversible with  $\phi = \bigwedge_{\text{pre}(a)[v]=\text{val}}(v, \text{val})$ . For another action  $a'$  and a formula  $\psi$  such that  $\psi = \bigwedge_{\text{pre}(a')[v]=\text{val}}(v, \text{val})$  with  $\phi \models \psi$ , we can observe that  $a'$  is universally (uniformly) reversible iff  $a$  is (uniformly)  $\psi$ -reversible.

**Proposition 6.** *Let  $\mathcal{D} = (V, A)$  be a planning domain and  $a \in A$  be an action. Let formula  $\phi$  be a conjunction of positive literals, such that  $\phi \models \text{pre}(a)$ . We define an action  $a'$  such that  $\text{eff}(a') = \text{eff}(a)$  and for each  $v$  with  $\phi \models (v, \text{val})$  it holds  $\text{pre}(a')[v] = \text{val}$  (for other variables  $\text{pre}(a')$  is undefined). Then,  $a'$  is universally (uniformly) reversible in  $\mathcal{D}' = (V, A \cup \{a'\})$  iff  $a$  is (uniformly)  $\phi$ -reversible in  $\mathcal{D}$ .*

*Proof.* It is clear that for each state  $s$  the action  $a'$  is applicable in, it holds that  $a$  is also applicable in  $s$ . Definition 1 states that for (uniform)  $\phi$ -reversibility we consider states in which the action in question is applicable and they satisfy  $\phi$  while for universal (uniform) reversibility we require that each state in which the action in question satisfies  $\phi$ . It can be observed that it is the case that each state in which the action  $a'$  also satisfies  $\phi$ . Since  $\phi \models \text{pre}(a)$  it can also be observed that the set of states in which  $a$  is applicable and each of the states satisfies  $\phi$  is the same as the set of states in which  $a'$  is applicable. That said, considering also that  $\text{eff}(a') = \text{eff}(a)$ , for a sequence of actions  $\pi$  (from  $A$ ) and a state  $s$ , where  $a'$  is applicable,  $s = \gamma(\gamma(s, a'), \pi)$  iff  $s = \gamma(\gamma(s, a), \pi)$ . Hence,  $a'$  is universally (uniformly) reversible iff  $a$  is (uniformly)  $\phi$ -reversible.  $\square$

The above proposition gives us a blueprint as to how universal uniform reversibility can be exploited to determine

uniform  $\phi$ -reversibility for known  $\phi$ . This is complementary to Algorithm 1 which generates  $\phi$  alongside the reverse plan.

Extending the  $\text{drive}$  action from our Logistics example, by requiring that the destination location must be free while making it occupied and the location of origin free afterwards, causes that  $\text{drive}(A,B)$  will no longer be universally uniformly reversible (e.g. it has to consider the  $\text{free}(C)$  variable which is not part of the precondition of  $\text{drive}(A,B)$ ). We can, however, leverage Proposition 6 by constructing a dummy action  $\text{dummy-drive}(A,B)$  requiring  $A$  occupied and  $C$  free (on top of the precondition of  $\text{drive}(A,B)$ ).  $\text{dummy-drive}(A,B)$  is then universally uniformly reversible yielding  $\text{drive}(A,B)$  uniformly  $\phi$ -reversible (where  $\phi$  represents the precondition of  $\text{dummy-drive}(A,B)$ ).

In the non-uniform case, we can split the problem of universal action reversibility into a set of problems of uniform action  $\phi$ -reversibility, i.e., the set of states can be partitioned via a set of formulas  $\phi_1, \dots, \phi_n$ , such that for each formula uniform action  $\phi_i$ -reversibility can be established. By ensuring that, for an action  $a$ , each state  $s$  in which  $a$  is applicable satisfies  $\phi_1 \vee \dots \vee \phi_n$ , universal (non-uniform) action reversibility can be established by ensuring that  $a$  is uniformly  $\phi_i$ -reversible for each partition of states induced by  $\phi_i$ ,  $0 < i \leq n$ . Note that, in general, such an approach is analogous to contingent planning as used for determining action reversibility by Daum et al. (2016), despite restricting only to states reachable from the initial state.

## 6 Discussion and Conclusion

In this paper, we have focused on special cases of action reversibility, universal and uniform action reversibility. Universal action reversibility means that for each state the action is applied in, its effects can be undone. Uniform action reversibility, on the other hand, means that for a subset of states in which the action is applied, the same action sequence reverts the effects of the action in question. A combination of both, i.e., universal uniform action reversibility, means having a single reverse plan for all possible states in which the action is applied.

Our main contribution, Theorem 3, establishes an equivalence between universal uniform reversibility of an action and the fact that the reverse plan does not contain actions with variables not present in the schema of the action in question. In consequence, such reverse plans can be found in the abstract state space that only considers variables occurring in the action to be reversed. Hence such plans can be found more easily as long as the number of such variables is low. The same abstract state space can be analysed to determine universal action irreversibility (i.e., whenever the action is applied, there is no way to undo its effects) as summarised in Proposition 5. Lastly, we have investigated how to leverage universal uniform reversibility to decide whether an action is uniformly  $\phi$ -reversible given the formula  $\phi$  up front (Proposition 6).

As future work, we would like to investigate action reversibility for actions in lifted form (i.e., with free variables that can be substituted by constants referring to problem-specific objects).

## Acknowledgements

Supported by the S&T Cooperation CZ 05/2019 “Identifying Undoable Actions and Events in Automated Planning by Means of Answer Set Programming”, by the Czech Ministry of Education, Youth and Sports under the Czech-Austrian Mobility programme (project no. 8J19AT025) and by the OP VVV funded project CZ.02.1.01/0.0/0.0/16\_019/0000765 “Research Center for Informatics”.

## References

- Camacho, A.; Muise, C. J.; and McIlraith, S. A. 2016. From FOND to robust probabilistic planning: Computing compact policies that bypass avoidable deadends. In *Proc. ICAPS*, 65–69.
- Chrpa, L.; Lipovetzky, N.; and Sardiña, S. 2017. Handling non-local dead-ends in agent planning programs. In *Proc. IJCAI*, 971–978.
- Cserna, B.; Doyle, W. J.; Ramsdell, J. S.; and Ruml, W. 2018. Avoiding dead ends in real-time heuristic search. In *Proc. AAAI*, 1306–1313.
- Daum, J.; Torralba, Á.; Hoffmann, J.; Haslum, P.; and Weber, I. 2016. Practical undoability checking via contingent planning. In *Proc. ICAPS*, 106–114.
- De Giacomo, G.; Gerevini, A. E.; Patrizi, F.; Saetti, A.; and Sardiña, S. 2016. Agent planning programs. *Artif. Intell.* 231:64–106.
- Edelkamp, S. 2001. Planning with pattern databases. In *Proceedings of the European Conference on Planning*, 13–24.
- Eiter, T.; Erdem, E.; and Faber, W. 2008. Undoing the effects of action sequences. *J. Applied Logic* 6(3):380–415.
- Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated planning, theory and practice*. Morgan Kaufmann Publishers.
- Ghallab, M.; Nau, D. S.; and Traverso, P. 2016. *Automated Planning and Acting*. Cambridge University Press.
- Grastien, A., and Scala, E. 2017. Intelligent belief state sampling for conformant planning. In *Proc. IJCAI*, 4317–4323.
- Helmert, M.; Haslum, P.; and Hoffmann, J. 2007. Flexible abstraction heuristics for optimal sequential planning. In *Proceedings of the Seventeenth International Conference on Automated Planning and Scheduling, ICAPS 2007*, 176–183.
- Helmert, M. 2009. Concise finite-domain representations for PDDL planning tasks. *Artif. Intell.* 173(5-6):503–535.
- Morak, M.; Chrpa, L.; Faber, W.; and Fiser, D. 2020. On the reversibility of actions in planning. In *Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020*, 652–661.