Published in *Artificial Intelligence*, 66(1): 125-160, 1994.

On the Computational Complexity of Temporal Projection, Planning, and Plan Validation*

Bernhard Nebel
German Research Center for AI (DFKI)
Stuhlsatzenhausweg 3, D-6600 Saarbrücken 11, Germany
nebel@dfki.uni-sb.de

Christer Bäckström
Department of Computer and Information Science
Linköping University
S-581 83 Linköping, Sweden
cba@ida.liu.se

Abstract

One kind of temporal reasoning is *temporal projection*—the computation of the consequences of a set of events. This problem is related to a number of other temporal reasoning tasks such as *plan validation* and *planning*. We show that one particular, simple case of temporal projection on partially ordered events turns out to be harder than previously conjectured, while planning is easy under the same restrictions. Additionally, we show that plan validation is tractable for an even larger class of plans—the unconditional plans—for which temporal projection is NP-hard, thus indicating that temporal projection may not be a necessary ingredient in planning and plan validation. Analyzing the partial decision procedure for the temporal projection problem that has been proposed by other authors, we notice that it fails to be complete for unconditional plans, a case where we have shown plan validation tractable.

*This work was supported by the German Ministry for Research and Technology (BMFT) under grant ITW 8901 8 as part of the WIP project, the Swedish National Board for Technology Development (STU) under grant STU 90-1675 and the Swedish Research Council for Engineering Sciences (TFR) under grant Dnr. 92-143.

1 Introduction

The problem of *temporal projection* is to compute the consequences of a set of events. Dean and Boddy [15] formalize and analyze this problem for sets of partially ordered events assuming a propositional STRIPS-like [19] representation of events. They investigate the computational complexity of a number of restricted problems and conclude that even for severely restricted cases the problem is NP-hard, which motivated them to develop a partial decision procedure for the temporal projection problem.

It turns out that the temporal projection problem is even harder than it was originally believed. Among the restricted problems Dean and Boddy analyzed, there is a particular "simple" one they conjectured to be solvable in polynomial time. However, even in this case temporal projection is NP-hard, as is shown below.

The main motivation for the isolation and analysis of the temporal projection problem [15] is the hypothesis that "a significant part of this process [nonlinear planning] involves some means for predicting the consequences of actions and using these consequences to verify whether or not a given partially constructed plan is likely to succeed" [14, p. 196].[1] To verify this hypothesis, we have taken a closer look at the complexity of nonlinear planning in relation to that of temporal projection. Our analysis shows that temporal projection is *not* necessarily a useful subproblem for solving the nonlinear planning problem. In particular, we identify cases were nonlinear planning is computationally easy, whereas the corresponding temporal projection task is intractable (assuming $\mathsf{P} \neq \mathsf{NP}$).

The *planning* problem is defined as follows: Given an initial world state, a desired world state, and a set of possible actions, find a (partially or totally ordered) set of of actions that, if executed in the initial world state, will bring about the desired world state [33].[2] Planning is a very difficult problem [9, 11, 12, 18, 17, 22]. However, the planning problem turns out to be trivial if we apply the restrictions of the "simple" temporal projection problem to it. Plans of minimal length are derivable in polynomial time in this case. Thus, under these restrictions, planning is strictly easier than temporal projection.

We also considered the *plan validation* problem. This is the problem of verifying that given a plan, an initial state, and a desired state, all actions mentioned in the plan can be successfully executed, *i.e.*, their preconditions are satisfied, and the actions of the plan lead to the desired state [33, p. 29]. Since planning

[1] One of the main motivations for our analysis is the development of efficient methods for solving a generalized plan validation problem that comes up in the context of representing and managing plans in a terminological representation system [23], which is used in the multi-media presentation system WIP [35].

[2] This means, we adopt the "classical" perspective on planning, *i.e.*, we assume that there is complete knowledge and the world is only changed by the actions of the agent executing the plan.

proceeds incrementally, one is usually not only interested in deciding the validity of a plan, but also in finding the reason for a failure if the plan is not yet valid. These "diagnoses of failure" can then be used to further develop the plan. In our paper, we abstract from these more practical considerations, however.

In the general case, plan validation and temporal projection of necessary consequences in the form as defined by Dean and Boddy belong to the same complexity class, but there does not seem to exist a natural decomposition of validation problems into projection problems. In the special case where only context-independent effects of actions are allowed, there exists a straightforward decomposition of plan validation into temporal projection problems. However, from a complexity point of view, this decomposition does not make much sense. Plan validation is a polynomial-time problem in this case, as can be shown using the techniques developed by Chapman [11], while solving the temporal projection problems is NP-hard.

The key idea in proving tractability of plan validation for context-independent actions is that any valid plan must be *coherent*, *i.e.*, all preconditions must be necessarily satisfied. Based on the tentative assumption that a plan is coherent, it is easy to decide whether it is indeed coherent. This notion can be quite naturally applied to prove a *modified form* of temporal projection for context-independent actions to be tractable, provided we are only interested in necessary consequences. Further, the notion of coherence can also be applied to plan validation for more expressive action languages, which leads to tractable but incomplete plan validation criteria.

The remainder of the paper is structured as follows. Section 2 contains the definition of the general temporal projection problem for partially ordered events as formalized by Dean and Boddy [15]. In Section 3, a *simple* form of temporal projection that was conjectured to be tractable by Dean and Boddy [15] is shown to be NP-hard. The corresponding planning problem permits a polynomial-time planning algorithm, however, as is shown in Section 4. In Section 5, we show that plan validation is tractable if all events are unconditional and analyze the relationship between this result and Chapman's [11]. In order to put this result into perspective, we analyze in Section 6 why plan validation appears to be easier than projection in this special case and define an *alternative form* of temporal projection that is tractable for necessary consequences. In addition, we discuss in how far the tractability results could be exploited in more general causal structures. Finally, in Section 7, we examine the partial decision algorithm proposed by Dean and Boddy [15].

2 Temporal Projection

Given a description of the state of the world and a description of which events will occur, we are usually able to predict what the world will look like. This kind of reasoning is called *temporal projection*. It seems to be the easiest and most basic kind of temporal reasoning. Depending on the representation, however, there are subtle difficulties hidden in this reasoning task.

The formalization of the temporal projection problem for partially ordered events given below is equivalent to the formalization given by Dean and Boddy [15, Sect. 2] but more tailored to meet our needs for proving different properties about temporal projection. We start with the definition of what a *causal structure* is, which fixes our vocabulary to talk about states, event types, and rules of change. We confine ourselves to a particular simple form of causal structures, where world states are represented by sets of propositional atoms and rules of change are described as propositional STRIPS-like operators. As a second step, we introduce *sets of partially ordered events* over causal structures that denote all event sequences that satisfy the partial order over the event set. Finally, the notion of *event systems* will be introduced that consist of an initial state and a partially ordered event set. The problem of temporal projection is to decide whether a given propositional atom holds, possibly or necessarily, after or before a given event in an event system.

Definition 1 *A* **causal structure** *is given by a tuple* $\Phi = \langle \mathcal{P}, \mathcal{E}, \mathcal{R} \rangle$*, where*

- $\mathcal{P} = \{p_1, \ldots, p_n\}$ *is a set of propositional atoms, the* **conditions***,*
- $\mathcal{E} = \{\epsilon_1, \ldots, \epsilon_m\}$ *is a set of* **event types***,*
- $\mathcal{R} = \{r_1, \ldots, r_o\}$ *is a set of* **causal rules** *of the form* $r_i = \langle \epsilon_i, \varphi_i, \alpha_i, \delta_i \rangle$*, where*
  - $\epsilon_i \in \mathcal{E}$ *is the* **triggering event type***,*
  - $\varphi_i \subseteq \mathcal{P}$ *is a set of* **preconditions***,*
  - $\alpha_i \subseteq \mathcal{P}$ *is the* **add list***,*
  - *and* $\delta_i \subseteq \mathcal{P}$ *is the* **delete list***.*

In order to illustrate this definition, assume a toy scenario as depicted in Figure 1. There is a hall, a room A, and another room B. Room A contains a public phone, and room B contains an electric outlet. The robot Robby can be in the hall (denoted by the atom `h`), in room A (`a`), or in room B (`b`). Robby can have a phone card (`p`) or coins (`c`). Additionally, when Robby uses the phone, he can inform his master on the phone that he was finally successful in proving the difficult theorem (`i`). Robby's batteries can be fully charged (`f`), almost empty (`e`), or, in unlucky circumstances, be damaged (`d`). Summarizing, the set of conditions for our tiny causal structure is the following:

$$\mathcal{P} = \{\mathtt{a}, \mathtt{b}, \mathtt{h}, \mathtt{p}, \mathtt{c}, \mathtt{i}, \mathtt{d}, \mathtt{e}, \mathtt{f}\}.$$

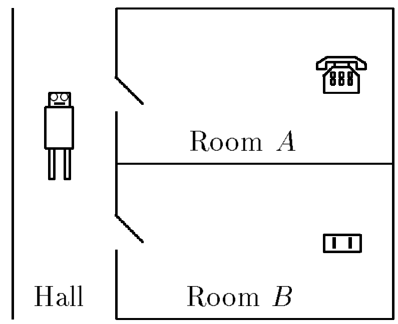

Figure 1: A toy scenario

Robby can do the following. He can move from the hall to either room ($\epsilon_{h\to a}$, $\epsilon_{h\to b}$) and *vice versa* ($\epsilon_{a\to h}$, $\epsilon_{b\to h}$). Provided he is in room A and he has a phone card or coins, he can call his master (ϵ_{call}). Additionally, if Robby is in room B, he can recharge himself (ϵ_{charge}). However, if Robby is already fully charged, this results in damaging his batteries. Summarizing, we have the following set of event types:

$$\mathcal{E} = \{\epsilon_{h\to a}, \epsilon_{h\to b}, \epsilon_{a\to h}, \epsilon_{b\to h}, \epsilon_{call}, \epsilon_{charge}\},$$

and the following set of causal rules:

$$\mathcal{R} = \Big\{ \begin{array}{llllllll} \langle \epsilon_{h\to a}, & \{\mathtt{h}\}, & \{\mathtt{a}\}, & \{\mathtt{h}\}\rangle, & \langle \epsilon_{h\to b}, & \{\mathtt{h}\}, & \{\mathtt{b}\}, & \{\mathtt{h}\}\rangle, \\ \langle \epsilon_{a\to h}, & \{\mathtt{a}\}, & \{\mathtt{h}\}, & \{\mathtt{a}\}\rangle, & \langle \epsilon_{b\to h}, & \{\mathtt{b}\}, & \{\mathtt{h}\}, & \{\mathtt{b}\}\rangle, \\ \langle \epsilon_{call}, & \{\mathtt{a},\mathtt{p}\}, & \{\mathtt{i}\}, & \emptyset\rangle, & \langle \epsilon_{call}, & \{\mathtt{a},\mathtt{c}\}, & \{\mathtt{i}\}, & \{\mathtt{c}\}\rangle, \\ \langle \epsilon_{charge}, & \{\mathtt{b},\mathtt{e}\}, & \{\mathtt{f}\}, & \{\mathtt{e}\}\rangle, & \langle \epsilon_{charge}, & \{\mathtt{b},\mathtt{f}\}, & \{\mathtt{d}\}, & \{\mathtt{f}\}\rangle \end{array} \Big\}.$$

In order to talk about sets of concrete events and temporal constraints over them, the notion of a *partially ordered event set* is introduced.[3]

Definition 2 *Assuming a causal structure $\Phi = \langle \mathcal{P}, \mathcal{E}, \mathcal{R}\rangle$, a* **partially ordered event set** *(POE) over Φ is a pair $\Delta_\Phi = \langle \mathcal{A}_\Phi, \prec\rangle$ consisting of*

1. *a set of* **actual events** *$\mathcal{A}_\Phi = \{e_1, \ldots, e_p\}$ with an associated function type: $\mathcal{A}_\Phi \to \mathcal{E}$, and*

2. *a strict partial order[4] $\prec$ over $\mathcal{A}_\Phi$.*

In the following, we will often drop the subscript Φ in Δ_Φ and $\mathcal{A}_\Phi$ if it is clear from the context which causal structure we mean. Continuing our example, we

[3] This notion is similar to the notion of a *nonlinear plan*.

[4] A strict partial order is a transitive and irreflexive relation.

assume a set of six actual events $\mathcal{A} = \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}, \mathtt{F}\}$, such that

$$\begin{array}{rclcrcl} type(\mathtt{A}) & = & \epsilon_{h\to a} & \quad & type(\mathtt{D}) & = & \epsilon_{h\to b} \\ type(\mathtt{B}) & = & \epsilon_{call} & & type(\mathtt{E}) & = & \epsilon_{charge} \\ type(\mathtt{C}) & = & \epsilon_{a\to h} & & type(\mathtt{F}) & = & \epsilon_{b\to h}, \end{array}$$

with the following temporal constraints

$$\mathtt{A} \prec \mathtt{B} \prec \mathtt{C} \text{ and } \mathtt{D} \prec \mathtt{E} \prec \mathtt{F}.$$

POEs denote sets of possible sequences of events satisfying the partial order. A **partial event sequence** of length m over such a POE $\langle \mathcal{A}, \prec\rangle$ is a sequence $\mathbf{f} = \langle f_1, \ldots, f_m\rangle$ such that (1) $\{f_1, \ldots, f_m\} \subseteq \mathcal{A}$, (2) $f_i \neq f_j$ if $i \neq j$, and (3) for each pair f_i, f_j of events appearing in $\mathbf{f}$, if $f_i \prec f_j$ then $i < j$. For instance, $\langle \mathtt{A}, \mathtt{B}, \mathtt{C}\rangle$ is a partial event sequence of length three over the POE given above, while $\langle \mathtt{A}, \mathtt{C}, \mathtt{B}\rangle$ is not. If the event sequence is of length $|\mathcal{A}|$, it is called a **complete event sequence** over the POE. The sequences $\langle \mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}, \mathtt{F}\rangle$ and $\langle \mathtt{A}, \mathtt{D}, \mathtt{B}, \mathtt{E}, \mathtt{C}, \mathtt{F}\rangle$ are complete event sequences, for instance. The set of all complete event sequences over a POE Δ is denoted by $CS(\Delta)$.

We say that a partial event sequence $\mathbf{f}$ can be **extended** to an event sequence $\mathbf{g}$ if $|\mathbf{f}| < |\mathbf{g}|$ and for all f_i, f_j with $i < j$ there exists $g_k = f_i$ and $g_l = f_j$ such that $k < l$. If $\mathbf{f} = \langle f_1, \ldots, f_k, \ldots, f_m\rangle$ is an event sequence, then $\langle f_1, \ldots, f_k\rangle$ is the initial sequence of $\mathbf{f}$ up to f_k, written $\mathbf{f}/f_k$. Similarly, $\mathbf{f}\backslash f_k$ denotes the initial sequence $\langle f_1, \ldots, f_{k-1}\rangle$ consisting of all events before f_k. Further, we write $\mathbf{f}; g$ to denote $\langle f_1, \ldots, f_m, g\rangle$.

Each event maps states (subsets of $\mathcal{P}$) to states. Let $S \subseteq \mathcal{P}$ denote a state and let e be an event. Then we say that the causal rule r is **applicable in state** S iff $r = \langle type(e), \varphi, \alpha, \delta\rangle$ and $\varphi \subseteq S$. Given e and S, $app(S, e)$ denotes the set of all **applicable rules** for e in state S. An event e is said to be **admissible** in a state S iff $app(S, e) \neq \emptyset$. In order to simplify notation, we write $\varphi(r)$, $\alpha(r)$, $\delta(r)$ to denote the sets φ, α, and δ, respectively, appearing in the rule $r = \langle \epsilon, \varphi, \alpha, \delta\rangle$. If there is only one causal rule associated with the event type $type(e)$, we will also use the notation $\varphi(e)$, $\alpha(e)$, and $\delta(e)$. Based on this notation, we define what we mean by the *result* of a sequence of events relative to a state S.

Definition 3 *The function "Result" from states and event sequences to states is defined recursively by:*

$$\begin{array}{rcl} Result\Big(S, \langle\rangle\Big) & = & S \\ Result\Big(S, (\mathbf{f}; g)\Big) & = & Result(S, \mathbf{f}) - \\ & & \{\delta(r) |\, r \in app(Result(S, \mathbf{f}), g)\} \ \cup \\ & & \{\alpha(r) |\, r \in app(Result(S, \mathbf{f}), g)\}. \end{array}$$

It is easy to verify that the following equation holds for our example scenario:

$$Result(\{\mathtt{h},\mathtt{e},\mathtt{c}\},\langle \mathtt{A},\mathtt{B},\mathtt{C},\mathtt{D},\mathtt{E},\mathtt{F}\rangle) = \{\mathtt{h},\mathtt{f},\mathtt{i}\}.$$

There are some points in Definition 3 that may appear to be problematical to the attentive reader. First, it is possible that two rules are applicable where one rules adds an atom and the other one deletes it. Although undesirable, this is permitted. In this case, we adopted in Definition 3 the (admittedly arbitrary) convention that the atom will be added (following Dean and Boddy [15]). A second arguable point is that the definition of the function *Result* permits sequences of events where events occur that are not admissible. For instance, it is possible to ask what the result of $\langle \mathtt{A},\mathtt{D},\mathtt{B},\mathtt{E},\mathtt{C},\mathtt{F}\rangle$ in state $\{\mathtt{h},\mathtt{e},\mathtt{c}\}$ will be:

$$Result(\{\mathtt{h},\mathtt{e},\mathtt{c}\},\langle \mathtt{A},\mathtt{D},\mathtt{B},\mathtt{E},\mathtt{C},\mathtt{F}\rangle) = \{\mathtt{h},\mathtt{e},\mathtt{i}\}.$$

Although perfectly well-defined, this result seems to be strange because the events `D, E`, and `F` occurred without being admissible in the states they occur in. In fact, it seems to be quite unintuitive that event `D`, *i.e.*, a movement of Robby from the hall to room B, can occur in a world state where Robby is in room A. In a natural language understanding context, one would take such a state of affairs as an incoherency, and perhaps attempt to fill in the missing event of Robby returning to the hall. In a planning context, the occurrence of `D` could be interpreted as a failed action attempt.

Definition 4 *An event sequence* $\mathbf{f} = \langle f_1,\ldots f_m\rangle$ *is called* **admissible in state** S *iff each event* f_i, $1 \leq i \leq m$, *is admissible in* $Result(S,\mathbf{f}\backslash f_i)$.

Depending on one's intuition, it may be preferable to define the function "*Result*" as a partial instead of a total function. The domain of "*Result*" would then be defined only over states and event sequences that are admissible in this state, a point we will return to in Section 6. The set of all complete event sequences over Δ that are admissible in S is denoted by $ACS(\Delta,S)$. If $CS(\Delta) = ACS(\Delta,S)$, we will say that Δ is **coherent** relative to S.

In the following, we will often talk about which consequences a POE will have on some initial state. For this purpose, the notion of an *event system* is introduced.

Definition 5 *An* **event system** Θ *is a pair* $\langle \Delta_\Phi, I\rangle$, *where* Δ_Φ *is a POE over the causal structure* $\Phi = \langle \mathcal{P},\mathcal{E},\mathcal{R}\rangle$, *and* $I \subseteq \mathcal{P}$ *is the* **initial state**.

In order to simplify notation, the functions CS and ACS are extended to event systems with the obvious meaning, *i.e.*, $CS(\langle\Delta,S\rangle) = CS(\Delta)$ and $ACS(\langle\Delta,S\rangle) = ACS(\Delta,S)$. Further, if $CS(\Theta) = ACS(\Theta)$, Θ is called **coherent**.

The problem of temporal projection as formulated by Dean and Boddy [15] is to determine whether some condition p holds, *possibly* or *necessarily*, after a particular event e of an event system Θ, written $p \in \mathit{Poss}^{+}(e,\Theta)$ and $p \in \mathit{Nec}^{+}(e,\Theta)$, respectively. We will also consider the problems of determining the sets of conditions that hold, possibly or necessarily, *before* a given event, written $\mathit{Poss}^{-}(e,\Theta)$ and $\mathit{Nec}^{-}(e,\Theta)$.

Definition 6 *Given an event system Θ, an event $e \in \mathcal{A}$, and a condition $p \in \mathcal{P}$:*

$$\begin{array}{rcl} p \in \mathit{Poss}^{+}(e,\Theta) & \mathit{iff} & \exists \mathbf{f} \in \mathit{CS}(\Theta)\text{: } p \in \mathit{Result}(I, \mathbf{f}/e) \\ p \in \mathit{Nec}^{+}(e,\Theta) & \mathit{iff} & \forall \mathbf{f} \in \mathit{CS}(\Theta)\text{: } p \in \mathit{Result}(I, \mathbf{f}/e) \\ p \in \mathit{Poss}^{-}(e,\Theta) & \mathit{iff} & \exists \mathbf{f} \in \mathit{CS}(\Theta)\text{: } p \in \mathit{Result}(I, \mathbf{f}\backslash e) \\ p \in \mathit{Nec}^{-}(e,\Theta) & \mathit{iff} & \forall \mathbf{f} \in \mathit{CS}(\Theta)\text{: } p \in \mathit{Result}(I, \mathbf{f}\backslash e). \end{array}$$

Hence, we have in fact four instead of one temporal projection problem. From a computational point of view, however, Nec^{+} and Nec^{-} are equivalent (under polynomial transformations), a property that also holds for Poss^{+} and Poss^{-}. Further, this property seems to extend to all restrictions on event systems.

Continuing our example, let us assume the initial state $I = \{h, e, c\}$. Then the following can be easily verified:

$$\begin{array}{lcl@{\qquad\qquad}lcl} \mathtt{i} & \in & \mathit{Poss}^{+}(\mathtt{B},\Theta) & \mathtt{i} & \notin & \mathit{Nec}^{+}(\mathtt{B},\Theta) \\ \mathtt{d} & \notin & \mathit{Poss}^{+}(\mathtt{E},\Theta) & \mathtt{d} & \notin & \mathit{Nec}^{+}(\mathtt{E},\Theta). \end{array}$$

In plain words, Robby is only possibly but not necessarily successful in informing his master about his success. On the positive side, however, we know that Robby's batteries will not be damaged, regardless of in which order the events happen.

3 A "Simple" Temporal Projection Problem

Given a set of conditions S and a sequence $\mathbf{f}$, $\mathit{Result}(S, \mathbf{f})$ can be computed in polynomial time by interpreting the definition of *Result* procedurally. Since the set $\mathit{CS}(\Theta)$ may contain exponentially many sequences, however, it is not obvious whether $p \in \mathit{Poss}^{+}(e,\Theta)$ and $p \in \mathit{Nec}^{+}(e,\Theta)$ can be decided in polynomial time.

In the general case, temporal projection as defined above is quite difficult. Dean and Boddy [15] show that the decision problems $p \in \mathit{Poss}^{+}(e,\Theta)$ and $p \in \mathit{Nec}^{+}(e,\Theta)$ are NP-complete and co-NP-complete, respectively, even under some severe restrictions, such as restricting α or δ to be empty for all rules, or requiring that there is only one causal rule associated with each event type. Considering the proofs of these results [15], they hold, quite obviously, also for the corresponding problems of deciding $p \in \mathit{Poss}^{-}(e,\Theta)$ and $p \in \mathit{Nec}^{-}(e,\Theta)$.

Definition 7 *A causal structure Φ is called* **unconditional** *iff for each $\epsilon \in \mathcal{E}$, there exists only one causal rule with the triggering event type ϵ. An event system $\langle \Delta_\Phi, I \rangle$ is called* **unconditional** *iff Φ is unconditional. An event system is called*

almost simple *iff it is unconditional and for each causal rule* $r = \langle \epsilon, \varphi, \alpha, \delta \rangle$*, the sets* α *and* δ *are singletons and* $\delta \subseteq \varphi$*. An event system is called* **simple** *iff it is unconditional,* I *is a singleton, and for each causal rule* $r = \langle \epsilon, \varphi, \alpha, \delta \rangle$*, the sets* φ*,* α*, and* δ *are singletons and* $\varphi = \delta$*.*

Dean and Boddy [15, Theorem 2.4] prove that the decision problem $p \in Poss^{+}(e, \Theta)$ is NP-complete for almost simple event systems and conjecture that it is a polynomial-time problem for simple event systems [15, p. 379]. As it turns out, however, also this problem is computationally difficult since the problem of *path with forbidden pairs* can be polynomially transformed to the simple temporal projection problem.

Definition 8 *An instance of the* **path with forbidden pairs** *(PWFP) problem is given by a directed directed graph* $G = (V, A)$*, two vertices* $s, t \in V$*, and a collection* $C = \Big\{\{a_1, b_1\}, \ldots, \{a_n, b_n\}\Big\}$ *of pairs of arcs from* A*. The question is: Does there exist a directed path from* s *to* t *in* G *that contains at most one arc from each pair in* C*?*

This problem is NP-complete as shown by Gabow *et al* [20], even if the graph is acyclic and all pairs are disjoint (see also [21, p. 203]).

Theorem 1 *Deciding* $p \in Poss^{+}(e, \Theta)$ *for simple event systems* Θ *is* NP*-complete.*

Proof Sketch. Membership in NP is obvious. Assume an acyclic directed graph with forbidden pairs of arcs such that all forbidden pairs are pairwise disjoint, and two nodes s and t. Assume without loss of generality that the final node t has no outgoing and only one incoming arc. In order to generate a temporal projection problem, nodes v_i are transformed to propositional atoms v_i, arcs $a_{i,j} = (v_i, v_j)$ are transformed to events $a_{i,j}$ with the associated causal rule $\langle type(a_{i,j}), \{v_i\}, \{v_j\}, \{v_i\} \rangle$, and the source node s is transformed to the initial state $\{s\}$. A forbidden pair $\{a, b\}$, where we assume without loss of generality that there is path from a to b, is interpreted as the temporal restriction $b \prec a$. Based on this transformation, it can be shown that there exists a path without forbidden pairs from s to t iff t is a possible consequence of the generated event system.[5] ■

In order to demonstrate the above sketched transformation, let us consider the graph with forbidden pairs in Figure 2. The temporal constraints in the generated event system would be

$$(\mathtt{x}, \mathtt{y}) \prec (\mathtt{v}, \mathtt{x}), \qquad (\mathtt{w}, \mathtt{y}) \prec (\mathtt{v}, \mathtt{w}).$$

[5] Full proofs are given in the appendix.

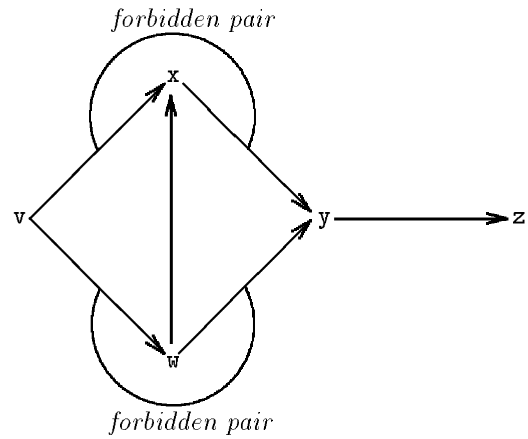

Figure 2: A graph with forbidden pairs

The only path from the source node `v` to the terminal node `z` that does not contain a forbidden pair is the path `v,w,x,y,z`. It is easy to see that this path could be used to generate a complete event sequence with `z` as its final consequence. As a first step, we use the partial event sequence consisting of all events corresponding to the arcs on this path:

$$\Big\langle (\mathtt{v},\mathtt{w}), (\mathtt{w},\mathtt{x}), (\mathtt{x},\mathtt{y}), (\mathtt{y},\mathtt{z}) \Big\rangle.$$

This sequence can be extended by the remaining events in a way such that they meet the temporal constraints and are not admissible in this sequence. These conditions can be easily satisfied because the temporal constraints involve only pairwise disjoint pairs of events. In our case the following complete sequence leads to the desired result:

$$\Big\langle (\mathtt{w},\mathtt{y}), (\mathtt{v},\mathtt{w}), (\mathtt{w},\mathtt{x}), (\mathtt{x},\mathtt{y}), (\mathtt{v},\mathtt{x}), (\mathtt{y},\mathtt{z}) \Big\rangle.$$

Conversely, it is easy to see that a complete event sequence leading to the atom corresponding to the terminal node implies the existence of a path without forbidden pairs. The subsequence consisting of all admissible events corresponds to a path from the source node to the terminal node (in our case, from `v` to `z`). Since this subsequence satisfies all temporal constraints, the corresponding path cannot contain forbidden paths.

Using a slight modification of the above sketched transformation, it can be easily shown that $p \in Nec^{+}(e, \Theta)$ is computationally equivalent to $p \notin Poss^{+}(e, \Theta)$, *i.e.*, it is co-NP-complete.

Corollary 2 *Deciding* $p \in Nec^{+}(e, \Theta)$ *for simple event systems* Θ *is co-*NP*-complete.*

From the above, it follows that the corresponding problems of deciding $p \in Poss^{-}(e, \Theta)$ and $p \in Nec^{-}(e, \Theta)$ are also complete for NP and co-NP, respectively.

An interesting observation in this context is that the sources of complexity identified by Dean and Boddy [15, p. 380], namely, conjunction by means of multiple preconditions ($|\varphi| > 1$) and disjunction in the form of multiple causal rules for one event, are not responsible for the intractability of the temporal projection problem. These sources of complexity are not present in our case. The sole source of complexity seems to be the partial ordering of events.

These results are somewhat surprising because one might suspect that planning and plan validation are easy under the restrictions imposed on the structure of event systems. We will analyze this point more thoroughly in the following sections.

4 Restricted Planning Problems

One reason for analyzing the temporal projection problem is that it seems to constitute the heart of nonlinear planning [14, p. 196]. If we now consider the restrictions placed on the simple temporal projection problem, it turns out that planning itself is quite easy under the same restrictions.

In the context of planning, events as introduced above are usually called **actions** and POEs are called **nonlinear plans**, or simply **plans**. In the following, we use these terms interchangeably.

Definition 9 *A* **planning task** Π *is given by* $\langle \Phi, I, G \rangle$, *where* $\Phi = \langle \mathcal{P}, \mathcal{E}, \mathcal{R} \rangle$ *is a causal structure as defined above, and* $I \subseteq \mathcal{P}$ *and* $G \subseteq \mathcal{P}$ *are the* **initial state** *and* **goal state**, *respectively. A* **plan** Δ_Φ **solves** Π *iff*

1. *the plan necessarily achieves the goal,* i.e., $G \subseteq Result(I, \mathbf{f})$ *for all* $\mathbf{f} \in CS(\Delta_\Phi)$, *and*

2. *the plan is coherent,* i.e., $ACS(\Delta_\Phi, I) = CS(\Delta_\Phi)$.

Note that we only allow plans where all actions are admissible (i.e. are guaranteed to have their preconditions satisfied), which coincides with the traditional definition [11, 33, 13]. Dean and Boddy [15], on the other hand, do not require valid plans to be coherent and define that non-admissible actions have no effect. Such a definition, however, makes very strong (implicit) assumptions about the underlying execution model, namely, that a failed action attempt does not lead to any unintended effects. Requiring coherence for validity is thus a *safe approach* since non-admissible actions are avoided in valid plans, making assumptions about the underlying execution model unnecessary.

The problem of planning is to decide whether there *exists* a solution for a planning task[6] (or, ultimately, to *find* a solution). If one is interested in plans of minimal length, the corresponding decision problem is to ask for the existence of a plan with a given length.

Definition 10 *An instance of the* **plan existence problem** *is a planning task* Π*. The question is: Does there exists a plan that solves* Π*?*

An instance of the **plan optimization problem** *is given by an integer* k *and a planning task* Π*. The question is: Does there exists a plan* $\Delta = \langle \mathcal{A}, \prec \rangle$ *such that* Δ *solves* Π *and* $|\mathcal{A}| \leq k$*?*

The computational complexity of planning has been investigated only recently. Bylander [9] analyzed the general problem of deciding the existence of a solution for a planning task in the context of propositional STRIPS-like representations[7] and showed that the general problem is PSPACE-complete. A number of restricted problems turn out to be tractable, however. For instance, plan existence for unconditional causal structures and causal rules restricted by $|(\alpha(r) \cup \delta(r))| = 1$ is tractable [9, Theorem 7]. Similarly, planning with causal rules such that the preconditions are always empty [9, Theorem 9] and planning with unconditional causal structures such that the goal state is restricted in size and all rules contain only one precondition [9, Theorem 8] are tractable. It should be noted, however, that Bylander [9] considers only the *existence* problem and not the associated *optimization* problem, which is often harder. For example, his Theorem 9 does not apply to the corresponding optimization problem because the *minimum cover* problem [21, p. 222] can be reduced to this planning problem [16, 10].

Proposition 3 *The plan optimization problem for planning tasks such that the preconditions of all causal rules are empty is* NP*-complete.*

Returning to the problem we analyzed in the previous section, similarly to *simple event systems* we define **simple planning tasks** to be planning tasks that meet the following restrictions: (1) there is only one causal rule associated with each event type, (2) for all causal rules $|\varphi| = |\alpha| = |\delta| = 1$ and $\varphi = \delta$, and (3) $|I| = 1$. Using Bylander's [9] Theorem 8, the tractability of the plan existence problem follows immediately. In this case, also plan optimization is tractable, however, since in this case planning can be reduced to a graph searching problem with a graph that is linearly bounded by the instance size.

[6]Note that we use the complexity-theoretic terminology here, where *problems* are sets of *instances*. In the terminology of planning research, instances of the planning problem are often called "planning problems." In order to avoid confusion, we called the latter *planning tasks*.

[7]Bylander allows incoherent plans and assumes that non-admissible actions have no effect. Since he considers only the existence of linear plans that solve a given task, such non-admissible actions can be safely removed from his solutions, however. Hence, Bylander's complexity results carry over also to the case where plans are required to be coherent.

Proposition 4 *For simple planning tasks, it can be decided in polynomial time whether there exists a solution. Further, plan optimization for simple planning tasks is also a polynomial-time problem.*

This result leads to the question why temporal projection, which is supposed to be the underlying problem in plan validation, is more difficult than planning itself in some cases. One explanation could be that a planner could create the complicated structure we used in the proof of Theorem 1, but it never would do so. Hence, the theoretical complexity never shows up in reality. In fact, all solutions of simple planning tasks are *linear* plans, *i.e.*, event sequences, for which temporal projection is tractable.

The natural question coming up is whether there are tractable planning problems that have truly nonlinear plans as solutions. Examples for such problems are the SAS-PUBS and SAS-PUS problems analyzed by Bäckström and Klein [6, 7].

The interesting point about these problems is that they are not defined by *local* restrictions on the causal rules, that the restrictions do not come up naturally in the formalism for specifying causal structures we use here, and that they are supposedly of more practical interest than the restricted classes of planning problems we have considered above. The SAS-PUBS and SAS-PUS planning problems are aimed at capturing planning tasks that come up in the domain of *sequential control*[8], where the action representation may be relatively simple, but the problem size makes computational complexity an important issue.

The SAS-PUS planning problem was originally formulated in the *simplified action structures* (SAS) formalism [6, 7], which is based on earlier work on action structures [3, 32]. In the following, we re-express the SAS-PUS restrictions in the formalism from Section 2 in order to facilitate a comparison with the planning problems we have considered so far.

Definition 11 *A planning task* $\Pi = \langle\langle \mathcal{P}, \mathcal{E}, \mathcal{R}\rangle, I, G\rangle$ *is* **SAS-PUS equivalent** *iff it satisfies the following restrictions:*

1. $\langle \mathcal{P}, \mathcal{E}, \mathcal{R}\rangle$ *is unconditional;*
2. $\mathcal{P}$ *can be partitioned into* m *disjoint subsets* $P_1, \ldots, P_m$ *s.t.* $|P_i| > 1$ *for* $1 \leq i \leq m$ *and for all causal rules* $\langle \epsilon, \varphi, \alpha, \delta\rangle \in \mathcal{R}$
   - *(a)* $\delta \subseteq \varphi$*,*
   - *(b)* $|\delta| = 1$*;*
   - *(c)* $|\varphi \cap P_i| \leq 1$ *for all* i*,*
   - *(d)* $|\alpha \cap P_i| = |\delta \cap P_i| \leq 1$ *for all* i*, and*

[8] Sequential control is a subdiscipline of *discrete event dynamical systems* within *automatic control.*

(e) $\alpha \cap \delta = \emptyset$.

3. $|I \cap P_i| = |G \cap P_i| = 1$ *for all* i.

4. *for all pairs of causal rules* $\langle \epsilon, \varphi, \alpha, \delta \rangle, \langle \epsilon', \varphi', \alpha', \delta' \rangle \in \mathcal{R}$

   (a) *if* $\varphi = \varphi'$, $\alpha = \alpha'$, *and* $\delta = \delta'$, *then* $\epsilon = \epsilon'$;

   (b) *if* $\epsilon \neq \epsilon'$, *then* $\alpha \cap \alpha' = \emptyset$; *and*

   (c) *for all* $1 \leq i \leq m$, *if* $(\varphi - \delta) \cap P_i \neq \emptyset$ *and* $(\varphi' - \delta') \cap P_i \neq \emptyset$ *then* $(\varphi - \delta) \cap P_i = (\varphi' - \delta') \cap P_i$.

The restrictions can be understood as follows. Each partition P_i can be viewed as the value domain of a state variable x_i, an action can change the value of a state variable only if it already has a defined value, an action can only change the value of one state variable, there must be no two different action types changing the same state variable to the same value (4b), and the initial state and the goal state are fully specified. Finally, restriction (4c) captures the notion of *single-valuedness* [6, 7]. Comparing these restrictions with the corresponding, but simpler restrictions in the SAS formalism [6, 7] it is easy to see why the SAS formalism was originally preferred for defining the SAS-PUS problem.

Theorem 5 *The plan optimization problem for* SAS-PUS *equivalent planning tasks is a polynomial-time problem.*

The SAS-PUS problem is not comparable with either Bylander's restricted problems or our simple planning problem, *i.e.*, the SAS-PUS problem is neither subsumed by nor does it subsume any of the those problems. In order to get an idea about the expressiveness of SAS-PUS planning tasks, it may be worthwhile to note that it permits formulating the *restricted primitive blocks world planning problem* [29], a problem Bylander used as an example for one of his restricted planning problems[9] [9, Theorem 10].

Although the SAS-PUS problem is incomparable to the other restricted planning problems mentioned in this section with respect to expressivity, we know that it presents an (almost) optimal tradeoff between expressivity and efficiency. Except for the conditions that the initial and goal states must be completely specified (condition (3)) and the requirement that an action may not change a state variable from the undefined (*i.e.* an arbitrary) value to a defined value (conditions (2a), (2b), and (2d)), which can be relaxed without endangering tractability

[9] To be precise, the blocks-world problem Bylander studied is slightly more expressive than the restricted primitive blocks-world problem. However, a blocks-world problem subsuming both these problems can be solved in polynomial time by encoding it as a SAS^{+}-PUS problem [4, 5].

[4, 5], all other conditions are necessary to guarantee tractability [8], provided we are interested in optimal plans.

While it is not obvious whether temporal projection as defined in Section 2 is NP-hard or not for SAS-PUS event systems, it is NP-hard for a slightly more general class of event systems. The SAS-US class of event systems/planning tasks, which may violate condition (4b), subsumes the class of simple event systems/planning tasks. Hence, SAS-US temporal projection is NP-hard, while SAS-US planning is solvable in polynomial time [8].

The restricted planning problems we have discussed here may appear to be expressively quite restricted. However, the research in identifying tractable planning problems is nevertheless one important aspect in "understanding the expressive and computational requirements for effective temporal reasoning" [14], we believe. Apart from the obvious advantage of identifying efficient algorithms for special cases, it also contributes to our understanding of where sources of complexity arise in planning. Interestingly, however, temporal projection in the general form as defined in Section 2 is not at all needed in these cases.

5 Temporal Projection and Plan Validation

As mentioned in the Introduction, the interest in the temporal projection problem stems from its assumed relevance to the plan validation problem. For this reason, it seems worthwhile to explore the relationships between temporal projection and plan validation.

Definition 12 *An instance of the* **plan validation** *problem is given by a planning task* Π *and a plan* Δ*. The question is: Does* Δ *solve* Π*?*

In the general case, *i.e.*, for unrestricted causal structures, it is well-known that plan validation is NP-hard [11, Intractability Theorem]. However, it is also not harder than the temporal projection of necessary consequences.

Proposition 6 *The plan validation problem for general causal structures is co-*NP*-complete.*

So, from a complexity-theoretic point of view, the two problems are simply equivalent. It may be the case, however, that from a conceptual point of view projection appears to be a subproblem of validation, *i.e.*, there exists a natural and elegant decomposition of the plan validation problem into subproblems that involve temporal projection.

Deciding whether a plan achieves the desired goals can be straightforwardly reduced to temporal projection. Given a planning task $\Pi = \langle \Phi, I, G \rangle$, and a plan Δ_Φ, we extend the plan by an event e_* that is not associated with any causal rule and occurs after all other events. The resulting plan is called Δ'_Φ. Now it is easy to see that Δ_Φ achieves G if, and only if, $G \subseteq Nec^-(e_*, \langle \Delta'_\Phi, I\rangle)$.

The second condition on a solution of a planning task (cf. Definition 9), namely, that all actions are executable in all linearizations of the nonlinear plan—that the plan is *coherent*—cannot be easily decomposed into temporal projection problems. Testing whether an action is executable amounts to testing whether *necessarily* at least one of the causal rules associated with the action can be applied. This cannot be expressed as a temporal projection problem as defined in Section 2 because it involves a *disjunction* over the preconditions of the rules associated with one event. In order to express this problem, we would have to extend the definition of temporal projection in a way such that one can test whether some (strictly positive) formula in disjunctive normal form holds necessarily before a given event.[10]

If we restrict ourselves to *unconditional* causal structures, coherence of a nonlinear plan can be easily reduced to a conjunction of temporal projection problems.

Proposition 7 *An unconditional event system Θ is coherent iff*

$$\forall e \in \mathcal{A}\colon \varphi(e) \subseteq Nec^-(e, \Theta).$$

Although this looks like an elegant *divide and conquer* strategy, it turns out to be just the opposite from a computational point of view. While it is NP-hard to solve the temporal projection problems for all events in the event system, the original problem of deciding coherence of a plan can be solved in polynomial time, as we show below. Further, since plan validation can easily be reduced to the coherence problem, the entire plan validation problem turns out to be solvable in polynomial time.

In order to simplify the following discussion, we will restrict ourselves to **consistent** unconditional event systems, which have to meet the restrictions that $\alpha(e) \cap \delta(e) = \emptyset$, for all $e \in \mathcal{A}$. Note that any unconditional event system Θ can be transformed into a consistent unconditional event system Θ' in polynomial time by setting

$$\begin{aligned} \varphi'(e) &= \varphi(e) \\ \alpha'(e) &= \alpha(e) \\ \delta'(e) &= \delta(e) - \alpha(e), \end{aligned}$$

for all $e \in \mathcal{A}$. Consulting the definition of *Result*, it is obvious that this modification does not change the outcome of $Result(S, \mathbf{f})$ for all $S \subseteq \mathcal{P}$ and all partial event sequences $\mathbf{f}$ over Θ.

[10] Of course, there exists a polynomial transformation from plan validation to temporal projection as defined in Section 2 because both problems belong to the same complexity class. However, this reduction is probably not a "natural" decomposition of the plan validation problem.

As a first step to specifying a polynomial-time algorithm that decides coherence for unconditional event systems, we define a simple syntactic criterion, written $\mathit{Maybe}^{-}(e, \Theta)$, that approximates $\mathit{Nec}^{-}(e, \Theta)$.

Definition 13 *Given a consistent, unconditional event system* Θ, *an atom* $p \in \mathcal{P}$, *and an event* $e \in \mathcal{A}$, $\mathit{Maybe}^{-}(e, \Theta)$ *is defined as follows:*

$$\begin{array}{lll} p \in \mathit{Maybe}^{-}(e,\Theta) & \textit{iff} & (1)\; p \in I \vee \exists e' \in \mathcal{A}\colon (e' \prec e \wedge p \in \alpha(e')) \wedge \\ & & (2)\; \neg\exists e' \in \mathcal{A} - \{e\}\colon (e' \not\prec e \wedge e \not\prec e' \wedge p \in \delta(e')) \wedge \\ & & (3)\; \forall e' \in \mathcal{A}\colon \big((e' \prec e \wedge p \in \delta(e')) \rightarrow \\ & & \qquad\qquad \exists e'' \in \mathcal{A}\colon (e' \prec e'' \prec e \wedge p \in \alpha(e''))\big). \end{array}$$

This definition resembles Chapman's [11] *modal truth criterion*. The first condition states that p has to be established before e. The second condition makes sure that there is no event unordered w.r.t. e that could delete p, and the third condition enforces that for all events that could delete p and that occur before e, some other event will reestablish p. It is obvious that this criterion can be checked in low-order polynomial time.

Proposition 8 $p \in \mathit{Maybe}^{-}(e, \Theta)$ *can be decided in polynomial time.*

Note that Maybe^{-} is neither sound nor complete w.r.t. Nec^{-} in the general case because we do not know whether the events referred to in the definition are admissible in all linearizations. However, Maybe^{-} coincides with Nec^{-} in the important special case that the event system is consistent and coherent.

Lemma 9 *Let* Θ *be an consistent unconditional event system. If* Θ *is coherent, then*

$$\forall e \in \mathcal{A}\colon \mathit{Nec}^{-}(e, \Theta) = \mathit{Maybe}^{-}(e, \Theta).$$

Proof Sketch. "$\subseteq$": Suppose $p \notin \mathit{Maybe}^{-}(e, \Theta)$. Then one of the conditions in the Definition of Maybe^{-} is not satisfied. Exploiting the fact that Θ is coherent, it is possible to show by case analysis that there exists always a sequence $\mathbf{f}$ such that $p \notin \mathit{Result}(I, \mathbf{f}\backslash e)$, hence $p \notin \mathit{Nec}^{-}(e, \Theta)$.

"$\supseteq$": Suppose $p \in \mathit{Maybe}^{-}(e, \Theta)$. Then by condition (1) it follows that for all event sequences $\mathbf{f}$, there is an event e' before e or identical to it such that p holds before e'. There must be a latest such event. By condition (2) and (3) it follows that the latest such event is identical to e. Hence, $p \in \mathit{Nec}^{-}(e, \Theta)$. ■

Now we can give a necessary and sufficient condition for coherence of consistent unconditional event systems.

Theorem 10 *A consistent unconditional event system* Θ *is coherent iff*

$$\forall e \in \mathcal{A}\colon \varphi(e) \subseteq \mathit{Maybe}^{-}(e, \Theta).$$

Proof Sketch. "$\Rightarrow$": Follows from Lemma 9.

"$\Leftarrow$": This is the tricky part. We want to derive that Θ is coherent without relying on the fact that Θ is already coherent. Using induction over the number of preconditions appearing in event systems solves the problem. ∎

By Proposition 7, Theorem 10, the fact that $p \in Maybe^{-}(e, \Theta)$ can be decided in polynomial time, and the fact that any unconditional event system can be transformed into a consistent one in polynomial time, it follows straightforwardly that coherence can be decided in polynomial time.

Corollary 11 *Coherence of unconditional event systems can be decided in polynomial time.*

Plan validation can easily be reduced to coherence, so it is a polynomial-time problem if the causal structure is unconditional.

Theorem 12 *Deciding whether a plan Δ_Φ is a solution for a planning task Π with an unconditional causal structure is a polynomial-time problem.*

The surprising point about this result is that it appears to be easier to solve a problem in its entirety than to decompose it into subproblems (temporal projection problems) and to solve these problems in isolation. There seems to be a certain synergy at work provided by the required coherence of an event system that allows us to solve the problem by deciding some simple syntactic conditions, which when taken together provide the solution.

Although maybe surprising, the essence of our result is not new. Chapman [11] used a similar technique to prove that deciding necessary truth in unconditional plans generated by the TWEAK planning system is a polynomial-time problem for a slightly different formalism. It should be noted, however, that Chapman's proof of the completeness and correctness of his *modal truth criterion* relies on the assumption that all events he refers to in his criterion are already (or will become eventually) necessarily admissible. Hence, Chapman's notion of necessary truth is not identical with Nec^{-}, but coincides with $Maybe^{-}$.

Since the planning strategy of TWEAK is aimed at satisfying all preconditions, this assumption seems to be reasonable. However, it sounds like a circular argument to base the decision of whether a plan is coherent on the property that it is already coherent. So, it seems to be the case that Chapman missed to prove something similar to our Theorem 10.

6 The Role of Coherence

There are at least two points which seem to deserve further analysis. First, the notions of *validation* and *projection* seem to be very closely related and the complexity results for unconditional causal structures may appear to be somehow surprising. In particular, it would be interesting to find out the reason why projection seems to be so much harder than validation. Second, the notion of *coherence* played an important rule in all proofs of the previous section and it seems to be interesting to explore whether and how this notion could be applied to more general causal structures.

Comparing the notions of validation and projection, the first difference one notes is that projection makes more fine grained distinctions than plan validation. While plan validation considers all event structures that contain just one event that is not admissible in one possible complete event sequence as invalid, temporal projection as defined in Section 2 gives results even in the presence of events that are not admissible. Consider, for instance, the following event system Θ:

$$
\begin{aligned}
\mathcal{P} &= \{\mathtt{p}, \mathtt{q}\} \\
\mathcal{E} &= \{\epsilon_1, \epsilon_2, \epsilon_*\} \\
\mathcal{R} &= \{\langle \epsilon_1, \{\mathtt{q}\}, \{\mathtt{p}\}, \emptyset \rangle, \langle \epsilon_2, \{\mathtt{p}\}, \emptyset, \{\mathtt{q}\} \rangle, \langle \epsilon_*, \emptyset, \emptyset, \emptyset \rangle\} \\
\mathcal{A} &= \{\mathtt{E}_1, \mathtt{E}_2, \mathtt{E}_*\} \\
I &= \{\mathtt{q}\}
\end{aligned}
$$

The types of the events and the partial order over the events is specified by Figure 3.

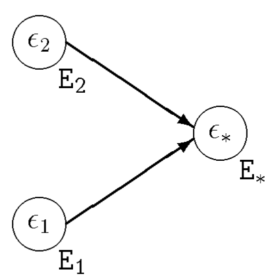

Figure 3: An incoherent structure

While plan validation would simply fail on Θ regardless of whether $\mathtt{p}$ or $\mathtt{q}$ is the desired goal, temporal projection yields $\mathtt{p} \in \mathit{Nec}^+(\mathtt{e}_*, \Theta)$ and $\mathtt{q} \notin \mathit{Nec}^+(\mathtt{e}_*, \Theta)$. Since there does not seem to be any obvious benefit in making these distinctions, one might look for an alternative definition of temporal projection that is more in line with the intuition that unadmissible event sequences lead to undefined states as spelled in Section 2.

When studying the NP-hardness proofs for projection problems over unconditional causal structures by Dean and Boddy [15] and our proof of Theorem 1, it turns out that all these proofs rely on event sequences that are not admissible. Hence, an alternative definition of temporal projection could perhaps be also more attractive from a computational point of view.

The answer we will give is quite interesting. Assuming a modified definition of temporal projection along the lines sketched above, *necessary* consequences can be easily computed by making again use of the synergy provided by the coherence of an event system. *Possible* consequences are, however, still difficult to compute. Hence, in this case the duality for the complexity of the temporal projection problems (NP-completeness for possible and co-NP-completeness for necessary consequences) does not hold any longer.

Instead of redefining the function *Result* as a partial function, we will redefine the projection problems to be based on admissible execution sequences. For this purpose, let us define a predicate *Adm* that is true just in case the first argument is a state and the second argument is an event sequence admissible in the first argument. Temporal projection can then be defined relative to admissible event sequences.

Definition 14 *Given an event system* Θ*, an event* $e \in \mathcal{A}$*, and a condition* $p \in \mathcal{P}$*:*

$$\begin{aligned}
p \in \mathit{Poss}^+_A(e,\Theta) \quad & \mathit{iff} \quad \exists \mathbf{f} \in \mathit{CS}(\Theta) : \mathit{Adm}(I, \mathbf{f}/e) \wedge p \in \mathit{Result}(I, \mathbf{f}/e) \\
p \in \mathit{Nec}^+_A(e,\Theta) \quad & \mathit{iff} \quad \forall \mathbf{f} \in \mathit{CS}(\Theta) : \mathit{Adm}(I, \mathbf{f}/e) \wedge p \in \mathit{Result}(I, \mathbf{f}/e) \\
p \in \mathit{Poss}^-_A(e,\Theta) \quad & \mathit{iff} \quad \exists \mathbf{f} \in \mathit{CS}(\Theta) : \mathit{Adm}(I, \mathbf{f}\backslash e) \wedge p \in \mathit{Result}(I, \mathbf{f}\backslash e) \\
p \in \mathit{Nec}^-_A(e,\Theta) \quad & \mathit{iff} \quad \forall \mathbf{f} \in \mathit{CS}(\Theta) : \mathit{Adm}(I, \mathbf{f}\backslash e) \wedge p \in \mathit{Result}(I, \mathbf{f}\backslash e).
\end{aligned}$$

This definition captures the intuition spelled out above, namely, that an event sequence should only have a result if all its events are admissible.[11] Consequently, a condition holds necessarily after or before a particular event e iff all possible event sequences up to this event e are in fact admissible and lead to the desired result. Similarly, possible consequences have to be based on possible event sequences that are admissible.

In order to show tractability of Nec^+_A and Nec^-_A, let us first consider a special case, namely, the projection of necessary consequences before an event which is always admissible and does not add or delete anything and which is a maximal element w.r.t. $\prec$.

Lemma 13 *Let* Θ *be an unconditional event system and let* $e_* \in \mathcal{A}$ *be an event such that* $e_* \not\prec e$ *for all* $e \in \mathcal{A}$ *and* $\varphi(e_*) = \alpha(e_*) = \delta(e_*) = \emptyset$*. Then*

$$p \in \mathit{Nec}^-_A(e_*,\Theta) \quad \mathit{iff} \quad p \in \mathit{Maybe}^-(e_*,\Theta) \text{ and } \mathit{CS}(\Theta) = \mathit{ACS}(\Theta).$$

[11] Interestingly, this modified definition of temporal projection seems also to be more in line with the informal definition of temporal projection given in a survey paper by Tate *et al* [33].

Proof Sketch. "$\Leftarrow$": Follows by Lemma 9.

"$\Rightarrow$": Since e_* is always admissible, does not change anything, and is a maximal element w.r.t. $\prec$, every complete sequence must be admissible, *i.e.*, Θ is coherent. By Lemma 9 it follows that $p \in \mathit{Maybe}^-(e, \Theta)$. ■

This special case can quite obviously be used to solve the general problem. Since events that appear necessarily after the event e_* cannot influence the state before e_*, it suffices to consider a sub-event system that contains only events that are not constrained to happen after the particular event e_*. Further, because deciding the coherence of an unconditional event system and deciding Maybe^- are polynomial-time problems, the entire problem can be decided in polynomial time.

Theorem 14 *Deciding* $p \in \mathit{Nec}_A^-(e, \Theta)$ *and* $p \in \mathit{Nec}_A^+(e, \Theta)$ *are polynomial-time problems.*

As mentioned above, all the NP-hardness proofs of temporal projection for possible consequences in unconditional event systems make use of event sequences that are not admissible. Hence, one may hope to carry over the positive result for Nec_A^+ to Poss_A^+. However, for possible consequences the modification of the definition of temporal projection does not result in a tractable problem. The main reason for the difficulty seems to be that it is already a difficult problem to decide whether an event system permits some admissible sequence.

Theorem 15 *The problem of deciding* $\mathit{ACS}(\Theta) \neq \emptyset$ *is* NP*-complete, even if* Θ *is an unconditional event system.*

From that it follows straightforwardly that the computation of possible consequences under the modified definition of temporal projection is NP-complete.

Corollary 16 *Deciding* $p \in \mathit{Poss}_A^-(e, \Theta)$ *or* $p \in \mathit{Poss}_A^+(e, \Theta)$ *is* NP*-complete.*[12]

Another interpretation of this result is that although it is easy to determine the *validity* of an unconditional plan, it is hard to check whether a plan is *satisfiable*, *i.e.*, whether it has an admissible and successful execution sequence, even if the goal is a singleton set.

As mentioned in the beginning of this section and emphasized by the theorems above, *coherence* of event structures does seem to play a very prominent role for the tractability of temporal reasoning tasks. It is obvious, however, that in the context of more general causal structure the coherence of event structures is not sufficient for tractability. Every event system over conditional causal

[12]Note that this does not contradict Chapman's claim that his modal truth criterion is also tractable in its dual form (for possibility). Since he makes the assumption that the plan is already (or will become eventually) coherent, his criterion for possibility differs from $\mathit{Poss}_A^-(e, \Theta)$.

structures could be made coherent simply by adding vacuous causal rules (see also Proposition 6).

However, it seems to be possible to formulate validation criteria that are sufficient but not necessary for the success of a plan that are based on the key idea of the tractability proofs in this and the previous section. This key idea is apparently to verify some simple condition for an action that guarantees the "correct behavior" of this action—provided this condition holds also for all other actions.

Such a criterion might, for instance, require that in all linearizations the same causal rules for each event get applied, a requirement one could call *strong coherence*. As a matter of fact, Pednault's [31] approach to nonlinear planning could be understood in this way.

This means, of course, that some correct plans may not be recognized as valid plans. However, as pointed out by McAllester and Rosenblitt [25], in order to drive the planning process it may be undesirable to use a plan validation criterion that is complete—even in the case where plan validation is tractable—because a complete criterion may not lead to a systematic exploration of the search space. In fact, using a complete criterion may be considerably less efficient than using an incomplete but (almost) systematic one [26].

When investigating this problem in the formal framework of our paper, one will note that the above described notion of strong coherence—the same set of causal rules for each event gets applied in every linearization—is again an NP-hard problem. The main problem is that deciding whether a set of atoms could hold possibly *simultaneously* before an event is NP-hard, even if the event system is unconditional and coherent.[13] Hence, in order to achieve tractability an even stronger (and less complete) validation criterion has to be used. One way could be to add a "safeness" condition that requires that for each rule that is not applied, there exists one atom in its precondition that never even possibly holds before the event.[14]

As a final remark, it should be noted that neither the complete plan validation method for unconditional plans described in the previous section nor the incomplete methods we have sketched here rely on temporal projection as defined in Section 2. Rather, computationally they are based on simple syntactic checks. Assuming that all events satisfy these checks, a global property of the plan can be derived. This is, of course, also a way of predicting the consequences of actions, but it may be *incorrect* as long as there are actions that do not satisfy the syntactic criterion. Nevertheless, it can be employed to incrementally generate a nonlinear plan.

[13] The proof of this claim is left as a not completely trivial exercise to the reader.

[14] As can be shown, this leads indeed to a polynomial-time and sound plan validation criterion. Further, this criterion is more general than that sketched in [27] because in our case the effects of rules that are not applied are ignored.

7 Approximate Temporal Projection

Based on the observation that temporal projection is difficult even for severely restricted cases, Dean and Boddy [15] developed an incomplete decision procedure that computes its results in polynomial time. Reconsidering the reflections from the previous sections, a natural question is whether the assumptions behind the design of the incomplete decision procedure led to a procedure that gives reasonable results. Such a judgement is, of course, quite difficult and depends heavily on the application setting.

In the area of reasoning about temporal relations between events [2], it was possible to identify tractable special cases that are natural for uncertain observations and text understanding [30, 34]. Further, the incomplete decision procedure for the full problem turned out to be complete for the tractable special case. Thus, we have a good justification for using the incomplete algorithm in this case.

If we consider the incomplete decision procedure for temporal projection, there is the question what the interesting special cases are where we want the procedure to be complete. Dean and Boddy [15, Theorem 3.4] prove their procedure to be complete if the events are totally ordered, which gives us one characterization of the behavior of the procedure.

It is, of course, interesting to characterize the procedure by additional cases for which it is complete. Such a characterization of an incomplete decision procedure gives the user of such a procedure some feeling of what he can expect. Under the assumption that the validation of nonlinear planning is the main application, the case of nonlinear plans containing only unconditional actions seems to be a nontrivial special case that deserves some attention.

From the discussions in the previous sections one is probably inclined to conjecture that the incomplete decision procedure by Dean and Boddy is not able to deal with this case in a complete manner. All in all, the procedure is based on the formalization of Section 2, which leads to computational problems in this case. Indeed, when tracing the procedure specified by Dean and Boddy, it turns out that the procedure does not lead to the projection of propositions that necessarily hold. The main reason for this failure is that the procedure considers all events *unordered* with respect to a given event as equally likely to appear. Condition (3) in the definition of $Maybe^-$, however, tells us that sometimes the deletion of an atom can be ignored.

Since we cannot reproduce the entire procedure because of space limitations, the reader is referred to the presentation in the original article [15, p. 380-392]. Here we will only sketch the ideas of the procedure. For every event e, two sets are computed, namely, $Strong(e, \Theta)$ and $Weak(e, \Theta)$, such that

$$Strong(e, \Theta) \subseteq Nec^+(e, \Theta) \subseteq Poss^+(e, \Theta) \subseteq Weak(e, \Theta),$$

where $Strong(e, \Theta)$ is intended to contain only conditions that hold after e in *all* complete event sequences, while $Weak(e, \Theta)$ is meant to contain all conditions that might hold after e in *some* complete event sequence.

In addition, the sets *S-Strong*(e, Θ) and *S-Weak*(e, Θ) are computed. The first set contains all of *Strong*(e, Θ) except those conditions that could be deleted by an event unordered with respect to e. Similarly, *S-Weak*(e, Θ) contains all of *Weak*(e, Θ) plus those conditions that could be added by events unordered with respect to e.

Consider now the following unconditional event system:

$$
\begin{aligned}
\mathcal{P} &= \{\mathtt{p}, \mathtt{q}, \mathtt{r}\} \\
\mathcal{E} &= \{\epsilon_a, \epsilon_b, \epsilon_c\} \\
\mathcal{R} &= \{\langle \epsilon_a, \{\mathtt{q}\}, \{\}, \{\mathtt{r}\}\rangle, \\
&\quad\ \langle \epsilon_b, \{\mathtt{q}\}, \{\mathtt{r}\}, \{\}\rangle, \\
&\quad\ \langle \epsilon_c, \{\mathtt{q}, \mathtt{r}\}, \{\mathtt{p}\}, \{\}\rangle\} \\
\mathcal{A} &= \{\mathtt{A}, \mathtt{B}, \mathtt{C}, \mathtt{D}, \mathtt{E}\} \\
I &= \{\mathtt{q}\}
\end{aligned}
$$

The types of the events and the partial order is given in Figure 4.

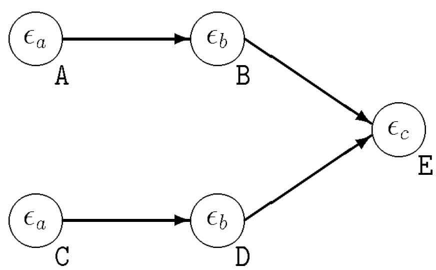

Figure 4: A valid nonlinear plan

It is easy to see that this unconditional event system is coherent and achieves $\{\mathtt{p}, \mathtt{q}, \mathtt{r}\}$. Using Theorem 12, this could be easily checked. However, the incomplete decision procedure is too conservative. It misses to report that `r` and `p` are among the necessary consequences, as can be seen from Table 1.[15]

In the computation of *S-Strong*(`B`) and *S-Strong*(`D`), the procedure is overly pessimistic with respect to the occurrence of the events `A` and `C`. Since these could delete the condition `r`, it may be the case that `r` does not hold before the occurrence of the event `E`. However, it is easy to see that `r` is necessarily added before occurrence of `E`.

[15]As the attentive reader will notice, there are some unmentioned assumptions in the specification of the partial decision procedure [15], e.g., that there exists an initial event before all other events—symbolized by the first row in our table. Besides that, we have to admit that the procedure is highly nontrivial and that we were unable to understand the procedure in all its details.

Event	Type	*S-Strong*	*Strong*	*Nec*$^+$	*Poss*$^+$	*Weak*	*S-Weak*
		{q}	{q}	{q}	{q}	{q}	{q}
A	ϵ_a	{q}	{q}	{q}	{q}	{q}	{q, r}
B	ϵ_b	{q}	{q, r}	{q, r}	{q, r}	{q, r}	{q, r}
C	ϵ_a	{q}	{q}	{q}	{q}	{q}	{q, r}
D	ϵ_b	{q}	{q, r}	{q, r}	{q, r}	{q, r}	{q, r}
E	ϵ_c	{q}	$\boxed{\{q\}}$	$\boxed{\{p, q, r\}}$	{p, q, r}	{p, q, r}	{p, q, r}

Table 1: Results of the incomplete decision procedure

Summarizing, we note that the procedure is not designed to handle some special case where plan validation is tractable. Although this is not surprising given our observations in the previous sections, it nevertheless provides a characterization of this procedure. In the case that only unconditional actions are of interest, the procedure is incomplete. One of the open problems we see here—as with other incomplete decision procedures—is to give an easy to understand characterization of when the procedure is complete and when and why incompleteness arises.

8 Conclusions

Reconsidering the problem of temporal projection for sets of partially ordered events as defined by Dean and Boddy [15], we noted that this problem is harder than originally believed. A particular, simple special case conjectured to be tractable turns out to be NP-complete. This result demonstrates that the only source of complexity for the temporal projection problem is the partial ordering of events and not, as conjectured by Dean and Boddy [15], multiple causal rules or multiple preconditions.

Since the original interest in the analysis of the computational properties of temporal projection originates from the hypothesis that temporal projection is a *significant* part of planning and plan validation [14, p. 196], we took a closer look at these problems. It turned out that planning is tractable under some restrictions on the representation of causal structures where we have shown temporal projection to be intractable. Turning to plan validation, we noted that in the general case (w.r.t. the framework set up by Dean and Boddy [14, 15]) temporal projection is not of much help. Its complexity is identical to the complexity of plan validation and there does not appear to be an elegant and natural decomposition of the plan validation problem that involves temporal projection problems.

Considering the special case of plans over unconditional structures, plan validation turns out to be decomposable into temporal projection problems. However, what looks like a *divide and conquer* strategy at a first glance is rather the opposite. Plan validation is a polynomial-time problem in this case, as we have shown, while the corresponding temporal projection problems are NP-hard.

Since temporal projection and plan validation seem to be very closely related, this result may appear to be counter-intuitive because it implies that it is impossible to reduce temporal projection to plan validation in case of unconditional causal structures. Analyzing the reasons for this result, it turns out that one particular assumption can be blamed for the difference in complexity, as long as we consider only the computation of necessary consequences. The original formulation of the temporal projection problem by Dean and Boddy permits event systems that are not *coherent*, *i.e.*, systems containing events that are not executable in some linearizations of the partial ordering. If we regard the result in such cases as undefined, temporal projection becomes tractable for necessary consequences. Computing possible consequences is still NP-hard, however.

Although the tractability results described above apply only to unconditional causal structures, the techniques used in the proofs might be used for designing tractable plan validation criteria for more general causal structures that are correct but incomplete. In exploring this issue we note again that temporal projection as defined in Section 2 is not essential for verifying that such a criterion is satisfied.

These observations lead to the question in how far the formalization of the temporal projection problem influenced the design of the partial decision procedure for temporal projection developed by Dean and Boddy [15]. As we demonstrate, the procedure fails to be complete on cases where we have shown plan validation to be tractable.

Summarizing, in the context of classical planning the hypothesis that temporal projection over partially ordered events as defined in Section 2 is a significant part of nonlinear planning and plan validation turns out to be invalid in some special cases. It is an interesting open problem, however, to determine whether the hypothesis holds under a modified definition of the temporal projection problem or for other forms of planning.

Acknowledgements

We would like to thank Gerd Brewka, Gerd Große, Bart Selman, Bill Swartout, and the two anonymous reviewers, who provided helpful comments on earlier versions of this paper.

Appendix: Proofs

Theorem 1 *Deciding* $p \in Poss^{+}(e, \Theta)$ *for simple event systems* Θ *is* NP*-complete.*

Proof. Membership in NP is obvious. Guess an event sequence $\mathbf{f}$ and verify in polynomial time that $\mathbf{f} \in CS(\Theta)$ and $p \in Result(I, \mathbf{f}/e)$.

In order to prove NP-hardness, we give a polynomial transformation from *path with forbidden pairs* (see Definition 8), where we assume that the graph is acyclic and all forbidden pairs are disjoint.

First of all, we specify a transformation from *directed acyclic graphs* (DAG) to simple event systems. Let $G = (V, A)$ be a DAG, where $V = \{v_1, \ldots, v_k\}$. Then define

$$\begin{array}{rcl} \mathcal{P} & = & \{v_1, \ldots, v_k\} \cup \{*\} \\ \mathcal{E} & = & \{\epsilon_{i,j} \mid (v_i, v_j) \in A\} \cup \{\epsilon_*\} \\ \mathcal{R} & = & \{\langle \epsilon_{i,j}, \{v_i\}, \{v_j\}, \{v_i\}\rangle \mid (v_i, v_j) \in A\} \cup \\ & & \{\langle \epsilon_*, \{*\}, \{*\}, \{*\}\rangle\} \\ \mathcal{A} & = & \{e_{i,j} \mid (v_i, v_j) \in A\} \cup \{e_*\} \\ type(e_{i,j}) & = & \epsilon_{i,j} \text{ for all } e_{i,j} \in \mathcal{A} - \{e_*\} \\ type(e_*) & = & \epsilon_* \\ e \prec e_* & & \text{for all } e \in \mathcal{A} - \{e_*\}. \end{array}$$

Note that such event systems, which we will call DAG event systems, are simple, provided $|I| = 1$.

Let $G = (V, A)$ be a DAG, let $C = \Big\{\{a_1, b_1\}, \ldots, \{a_n, b_n\}\Big\}$ be a collection of "forbidden pairs" of arcs from A such that each pair consists of different arcs and the pairs are pairwise disjoint. Further, let s and t be two nodes from V and assume without loss of generality that there is no arc $(t, v_i) \in A$.

Let Θ be the corresponding DAG event system with $I = \{s\}$. For each pair of arcs $\{(v_i, v_j), (v_k, v_l)\} \in C$,

1. if there is a (possibly empty) path from v_j to v_k in G add $e_{k,l} \prec e_{i,j}$ as a temporal constraint to Θ,

2. if there is a (possibly empty) path from v_l to v_i in G, add $e_{i,j} \prec e_{k,l}$ as a temporal constraint to Θ.

Since the graph is acyclic, it is impossible that (1) and (2) apply simultaneously to a pair of arcs. Further, since the forbidden pairs are pairwise disjoint, there is no set of events $\{f_1, f_2, f_3\} \subseteq \mathcal{A} - \{e_*\}$ such that $f_1 \prec f_2 \prec f_3$. Note that the entire transformation can obviously be computed in polynomial time.

For the resulting event system, we claim that there is a path from s to t in G that contains at most one arc from each pair in C iff $t \in Poss^+(e_*, \Theta)$.

"$\Rightarrow$": Let $v_1, \ldots, v_m$, $1 \leq m \leq |V|$, be a path in G, where $v_1 = s$ and $v_m = t$, without forbidden pairs from C. Then by construction of Θ, there exists a sequence of events $\mathbf{g} = \langle g_1, \ldots, g_{m-1}\rangle$ such that $\langle type(g_i), \{v_i\}, \{v_{i+1}\}, \{v_i\}\rangle \in \mathcal{R}$. Note that this sequence is indeed a partial event sequence over Θ because the path does not contain forbidden pairs, and, hence there are no temporal constraints for the events appearing in $\mathbf{g}$. Furthermore, we have for $\alpha(g_{m-1}) = \{t\}$. By the construction of Θ, it holds that

$$Result(I, (\mathbf{g}; e_*)/e_*) = \{t\}.$$

The sequence $\mathbf{g}; e_*$ can be extended to a complete event sequence $\mathbf{h}$ over Θ in the following way:

1. add all events f that are not temporally constrained and do not appear in $\mathbf{g}$ immediately before e_*,
2. add all pairs of events f, f' such that $f \prec f'$ and such that f and f' do not appear in $\mathbf{g}$ immediately before e_* respecting $\prec$,
3. add all events f that do not appear in $\mathbf{g}$ and $f \prec g_i$ for some g_i appearing in $\mathbf{g}$ immediately before g_i,
4. add all events f that do not appear in $\mathbf{g}$ and $g_i \prec f$ for some g_i appearing in $\mathbf{g}$ immediately after g_i.

Note that for extensions of the forms (1) and (2) it holds trivially that

$$Result(I, \mathbf{h}/e_*) = \{t\} \text{ iff } Result(I, (\mathbf{g}; e_*)/e_*) = \{t\}$$

since no precondition of any rule contains t by assumption. For extensions of the form (3) it holds that $e_{i,j} \prec e_{k,l}$ only if there is path from v_l to v_i in G. Hence, if $e_{i,j}$ is placed immediately before $e_{k,l}$, the precondition of the causal rule associated with $e_{i,j}$ cannot be satisfied. Thus, the above equivalence also holds for case (3). Since (4) is the converse case, the equivalence also holds.

Summarizing, we have for the complete event sequence $\mathbf{h}$

$$Result(I, \mathbf{h}/e_*) = \{t\}.$$

Thus, $t \in Poss^+(e_*, \Theta)$.

"$\Leftarrow$": Assume $t \in Poss^+(e_*, \Theta)$. Then there exists a complete event sequence $\mathbf{g}$ such that

$$Result(I, \mathbf{g}/e_*) = \{t\}.$$

Consider the subsequence $\mathbf{h}$ containing only admissible events:

$$\mathbf{h} = \langle h_1, h_2, \ldots, h_{m-1} \rangle.$$

By the construction of the causal rules in Θ and the form of the initial state it is evident that each event in the subsequence $\mathbf{h}$ has an add list that is identical to the precondition of the immediately following event. Since the initial state is $I = \{s\}$ and $Result(I, \mathbf{h}) = \{t\}$, there must be a path $v_1, v_2, \ldots, v_m$ in G, where $s = v_1$ and $t = v_m$.

Finally, this path cannot contain any forbidden pair. Suppose the contrary, *i.e.*, the path is of the form $s, \ldots, v_i, v_j, \ldots, v_k, v_l, \ldots, t$ and $\{(v_i, v_j), (v_k, v_l)\} \in C$. Thus, there is a path from v_j to v_k. In this case, however, we have $e_{k,l} \prec e_{i,j}$ by the construction of $\prec$ in Θ. This means, however, that $\mathbf{h}$ cannot be a possible event sequence over Θ. Hence, there cannot be any event sequences leading to t that contain forbidden pairs. ■

Corollary 2 *Deciding $p \in \mathit{Nec}^{+}(e, \Theta)$ for simple event systems Θ is co-*NP*-complete.*

Proof. We show that $p \notin \mathit{Nec}^{+}(e, \Theta)$ is NP-complete. Membership in NP is obvious. For the NP-hardness part, we start with the same transformation as in the proof of Theorem 1. We add to Θ a new condition p and a number of events f_v with associated causal rules of the form:

$$\langle \mathit{type}(f_v), \{v\}, \{p\}, \{v\} \rangle,$$

for all $v \in V - \{t\}$. These events are constrained to happen before e_* and after all other events constructed in the above reduction.

Now, it follows by the same arguments as in the proof of Theorem 1 that $p \notin \mathit{Nec}^{+}(e_*, \Theta)$ iff there is a path from s to t without forbidden pairs. ■

Theorem 5 *The plan optimization problem for* SAS-PUS *equivalent planning tasks is a polynomial-time problem.*

Proof Sketch. Define a transformation between sets of propositions and partial states in the SAS formalism and also map action conditions in the obvious way. Prove that a SAS-PUS-equivalent planning task Π can be transformed into a SAS-PUS planning task Π' in this way s.t. the solutions for Π' are exactly the solutions for Π. Since the plan optimization problem for SAS-PUS tasks is a polynomial-time problem [6, Theorem 4.2], the theorem follows. ■

Lemma 9 *Let Θ be an consistent unconditional event system. If Θ is coherent, then*

$$\forall e \in \mathcal{A}\colon \mathit{Nec}^{-}(e, \Theta) = \mathit{Maybe}^{-}(e, \Theta).$$

Proof. "$\subseteq$": We will show that all three conditions of $p \in \mathit{Maybe}^{-}(e, \Theta)$ in Definition 13 are true for all $e \in \mathcal{A}$ and all $p \in \mathit{Nec}^{-}(e, \Theta)$.

Suppose that the first condition does not hold for some event e and atom $p \in \mathit{Nec}^{-}(e, \Theta)$, *i.e.*, $p \notin I$ and $\neg \exists e'\colon e' \prec e \wedge p \in \alpha(e')$. Since Θ is coherent, we can construct an admissible complete event sequence $\mathbf{f} = \langle f_1, \ldots, e, \ldots \rangle$ such that $\mathbf{g} = \mathbf{f} \backslash e$ contains only events g_i such that $g_i \prec e$. By induction over the length of the length of $\mathbf{f} \backslash e$, we get $p \notin \mathit{Result}(I, \mathbf{f} \backslash e)$, hence $p \notin \mathit{Nec}^{-}(e, \Theta)$, which is a contradiction.

Suppose that the second condition does not hold for some event e and atom $p \in \mathit{Nec}^-(e, \Theta)$, *i.e.*, there exists an event e' unordered with respect to e such that $p \in \delta(e')$. Since e' is unordered with respect to e, there exists a complete event sequence $\mathbf{f} = \langle f_1, \ldots, e', e, \ldots \rangle$. Since Θ is coherent, and thus e' necessarily admissible, it is obvious that $p \notin \mathit{Result}(I, \mathbf{f}/e') = \mathit{Result}(I, \mathbf{f}\backslash e) \supseteq \mathit{Nec}^-(e, \Theta)$, which is a contradiction.

Suppose the third condition is not satisfied, *i.e.*, there exists $p \in \mathit{Nec}^-(e, \Theta)$ and an event $e' \prec e$ such that $p \in \delta(e')$, but there is no e'' such that $e' \prec e'' \prec e$ and $p \in \alpha(e'')$. Consider a complete event sequence $\mathbf{f} = \langle f_1, \ldots, e', \ldots, e, \ldots \rangle$ such that there are only events f_i between e' and e that have to occur between them. Because $p \notin \mathit{Result}(I, \mathbf{f}/e')$ and there are no events after e' that have p in the add list, using induction on the length of $\mathbf{f}\backslash e$, we can infer $p \notin \mathit{Result}(I, \mathbf{f}\backslash e) \supseteq \mathit{Nec}^-(e, \Theta)$, which is again a contradiction.

"$\supseteq$": Assume $p \in \mathit{Maybe}^-(e, \Theta)$. We will show that also $p \in \mathit{Nec}^-(e, \Theta)$. Consider any complete event sequence $\mathbf{g} \in \mathit{CS}(\Theta)$. We want to show that $p \in \mathit{Result}(I, \mathbf{g}\backslash e)$. By condition (1) of the definition of Maybe^- and the fact that all complete event sequences are admissible, we know that there exists $g_i \in \mathcal{A}$ such that $|\mathbf{g}\backslash g_i| \leq |\mathbf{g}\backslash e|$ and $p \in \mathit{Result}(I, \mathbf{g}\backslash g_i)$. Consider the latest such event, *i.e.*, g_i with a maximal i. Since all event sequences are finite, such an event must exist. If $g_i = e$, we are ready. Otherwise, we will show that i cannot be maximal.

Since g_i is the latest event in $\mathbf{g}$ such that $p \in \mathit{Result}(I, (\mathbf{g}\backslash e)\backslash g_i)$, it must be the case that $p \in \delta(g_i)$. By condition (2) in the definition of Maybe^-, we know that g_i cannot be unordered with respect to e. By condition (3), we know that there exists an event g_j such that $g_i \prec g_j \prec e$ and $p \in \alpha(g_j)$. Since $\mathbf{g}$ is admissible by assumption, it must be the case that $p \in \mathit{Result}(I, \mathbf{g}/g_j)$ and $|\mathbf{g}\backslash g_i| < |\mathbf{g}/g_j| \leq |\mathbf{g}\backslash e|$. Hence, g_i cannot be the latest event before e and different from e such that p holds before the occurrence of g_i. Hence, $p \in \mathit{Result}(I, \mathbf{g}\backslash e)$.

Since $\mathbf{g}$ was an arbitrary element of $\mathit{CS}(\Theta)$, this holds for all complete event sequences. Hence, $p \in \mathit{Nec}^-(e, \Theta)$. ■

Theorem 10 *A consistent unconditional event system Θ is coherent iff*

$$\forall e \in \mathcal{A}\colon \varphi(e) \subseteq \mathit{Maybe}^-(e, \Theta).$$

Proof. "$\Rightarrow$": Since Θ is coherent, we know that $\forall e \in \mathcal{A}\colon \varphi(e) \subseteq \mathit{Nec}^-(e, \Theta)$. Further, by Lemma 9, $\mathit{Maybe}^-(e, \Theta) = \mathit{Nec}^-(e, \Theta)$, for all $e \in \mathcal{A}$. Hence, $\forall e \in \mathcal{A}\colon \varphi(e) \subseteq \mathit{Maybe}^-(e, \Theta)$.

"$\Leftarrow$": For the converse direction, we use induction on the number of conditions appearing in the preconditions of events over the entire event system: $\sum_{e \in \mathcal{A}} |\varphi(e)|$. As the base step, we assume, that for all events $e \in \mathcal{A}$, $\varphi(e) = \emptyset$. Clearly, $\varphi(e) \subseteq \mathit{Maybe}^-(e, \Theta)$ and $\varphi(e) \subseteq \mathit{Nec}^-(e, \Theta)$, for all $e \in \mathcal{A}$. Hence, the hypothesis holds for $k = 0$.

Now assume that our claim holds for all event systems with k or less preconditions. We will show that it also holds for event systems with $k+1$ preconditions.

Consider an event system Θ with $k+1$ preconditions such that $\varphi(e) \subseteq \mathit{Maybe}^-(e,\Theta)$ for all $e \in \mathcal{A}$. Choose one event f that has a nonempty set of preconditions and replace the associated causal rule $\langle \mathit{type}(f), \varphi, \alpha, \delta\rangle$ by the rule $\langle \mathit{type}(f), \emptyset, \alpha, \delta\rangle$. This new event system is called Θ'. We will write $\varphi'(e)$, $\alpha'(e)$, and $\delta'(e)$ in order to refer to the preconditions, add lists, and delete lists in Θ', respectively. Note that for all $e \in \mathcal{A}$ it holds that $\mathit{Maybe}^-(e,\Theta') = \mathit{Maybe}^-(e,\Theta)$ because the Maybe^- conditions do not refer to the preconditions. Since the only change from Θ to Θ' was the removal of the preconditions of f, we clearly have $\varphi'(e) \subseteq \mathit{Maybe}^-(e,\Theta')$ for all $e \in \mathcal{A}$. Because $k \geq \sum_{e\in\mathcal{A}'} |\varphi'(e)|$, we can apply our induction hypothesis and know that Θ' is coherent. Finally note that by Lemma 9, we have $\mathit{Maybe}^-(f,\Theta') = \mathit{Nec}^-(f,\Theta')$ for our special event f. Hence, any sequence $\mathbf{g} \in \mathit{CS}(\Theta')$ that contains f would be an admissible sequence even if we assume that the causal rule associated with f has the original precondition $\varphi(f)$ because we assumed that $\varphi(f) \subseteq \mathit{Maybe}^-(f,\Theta)$, where $\mathit{Maybe}^-(f,\Theta) = \mathit{Maybe}^-(f,\Theta')$. Since we have $\mathit{CS}(\Theta) = \mathit{CS}(\Theta')$, it follows that all sequences $\mathbf{h} \in \mathit{CS}(\Theta)$ are admissible. Hence, Θ is coherent, whence, the induction hypothesis holds for $k+1$ preconditions. ■

Theorem 12 *Deciding whether a plan Δ_Φ is a solution for a planning task Π with an unconditional causal structure is a polynomial-time problem.*

Proof. Follows immediately from Corollary 11 and the fact that plan validation can be reduced to coherence in linear time as follows: Add an extra event e_* s.t. $\varphi(e_*)$ is the intended effects of the plan and e_* is constrained to occur after all other events. ■

Lemma 13 *Let Θ be an unconditional event system and let $e_* \in \mathcal{A}$ be an event such that $e_* \not\prec e$ for all $e \in \mathcal{A}$ and $\varphi(e_*) = \alpha(e_*) = \delta(e_*) = \emptyset$. Then*

$$p \in \mathit{Nec}^-_A(e_*,\Theta) \quad \textit{iff} \quad p \in \mathit{Maybe}^-(e_*,\Theta) \textit{ and } \mathit{CS}(\Theta) = \mathit{ACS}(\Theta).$$

Proof. "$\Leftarrow$": Suppose $p \in \mathit{Maybe}^-(e_*,\Theta)$ and $\mathit{CS}(\Theta) = \mathit{ACS}(\Theta)$. By Lemma 9 it follows that $p \in \mathit{Nec}^-(e_*,\Theta)$, hence, for all complete event sequences $\mathbf{f}$ over Θ, we have $p \in \mathit{Result}(I, \mathbf{f}\backslash e_*)$. Further, since all sequences $\mathbf{f} \in \mathit{CS}(\Theta)$ are admissible, all sequences $\mathbf{f}\backslash e_*$ must be admissible. Hence, $p \in \mathit{Nec}^-_A(e_*,\Theta)$.

"$\Rightarrow$": Suppose $p \in \mathit{Nec}^-_A(e_*,\Theta)$, *i.e.*, for all complete sequences $\mathbf{f} \in \mathit{CS}(\Theta)$, the event sequences $\mathbf{f}\backslash e_*$ are admissible. Now suppose that Θ is not coherent, *i.e.*, there exists a sequence $\mathbf{g} = \langle g_1,\ldots,g_i,e_*,g_{i+2},\ldots,g_m\rangle$ such that $\mathbf{g}$ is not admissible, but $\mathbf{g}\backslash e_*$ is. Hence, $\langle e_*, g_{i+2},\ldots g_m\rangle$ is not admissible in $\mathit{Result}(I, \mathbf{g}\backslash e_*)$. Since by assumption e_* is always admissible and does not add or delete anything, and since $\langle e_*, g_{i+2},\ldots g_m\rangle$ is not admissible in $\mathit{Result}(I, \langle g_1,\ldots g_i\rangle)$, $\langle g_1,\ldots g_i, g_{i+2},\ldots g_m\rangle$ cannot be admissible in I. Further, since e_* is a maximal element with respect to $\prec$, $g_{i+2},\ldots,g_m$ must be unordered with respect to e_*. For this reason $\mathbf{h} = \langle g_1,\ldots,g_i,g_{i+2},\ldots,g_m,e_*\rangle$ must also be an element of $\mathit{CS}(\Theta)$. However, that

$\langle g_1, \ldots, g_i, g_{i+2}, \ldots, g_m \rangle = \mathbf{g} \backslash e_*$ is not admissible in I contradicts the assumption that all complete sequences up to e_* are admissible. Hence, Θ must be coherent, and for this reason we have $Nec_A^-(e_*, \Theta) = Nec^-(e_*, \Theta)$. Because of Lemma 9, it follows then that we must have $p \in Maybe^-(e_*, \Theta)$. ■

Theorem 14 *Deciding $p \in Nec_A^-(e, \Theta)$ and $p \in Nec_A^+(e, \Theta)$ are polynomial-time problems.*

Proof. In the following we consider only Nec_A^-. The proof for Nec_A^+ is similar.

Consider the event system Θ' that is identical to Θ except that the set of actual event $\mathcal{A}'$ is a subset of the original set $\mathcal{A}$ defined in the following way: $\mathcal{A}' = \{f \in \mathcal{A} | \, e \not\prec f\}$. Now we claim that

$$p \in Nec_A^-(e, \Theta) \text{ iff } p \in Nec_A^-(e, \Theta').$$

"$\Rightarrow$": Suppose $p \notin Nec_A^-(e, \Theta')$, *i.e.*, there exists a sequence $\mathbf{g} \in CS(\Theta')$ such that $\mathbf{g} \backslash e$ is not admissible or $p \notin Result(I, \mathbf{g} \backslash e)$. Since it is possible to extend $\mathbf{g}$ to a complete event sequence $\mathbf{f}$ over Θ by adding the events from $\mathcal{A} - \mathcal{A}'$ to the end of $\mathbf{g}$ without violating temporal constraints, it must be the case that $\mathbf{f} \backslash e$ is not admissible or $p \notin Result(I, \mathbf{f} \backslash e)$, hence $p \notin Nec_A^-(e, \Theta)$.

"$\Leftarrow$": Suppose $p \notin Nec_A^-(e, \Theta)$, *i.e.*, there is a sequence $\mathbf{f} \in CS(\Theta)$ such that $\mathbf{f} \backslash e$ is not admissible or $p \notin Result(I, \mathbf{f} \backslash e)$. Consider the sequence $\mathbf{g}$ that is identical to $\mathbf{f}$ except that all events from $\mathcal{A} - \mathcal{A}'$ have been removed. This sequence is obviously a member of $CS(\Theta')$. Now it is easy to see that $\mathbf{f} \backslash e = \mathbf{g} \backslash e$ because all events of $\mathcal{A} - \mathcal{A}'$ have to appear after e. Hence, $\mathbf{g} \backslash e$ is either unadmissible or does not lead to p, *i.e.*, $p \notin Nec_A^-(e, \Theta')$.

Hence, we can apply Lemma 13 to solve the problem stated in the Theorem, and as an immediate consequence of Proposition 8 and Corollary 11, we get that $p \in Nec_A^-(e, \Theta)$ can be decided in polynomial time. ■

Theorem 15 *The problem of deciding $ACS(\Theta) \neq \emptyset$ is* NP*-complete, even if Θ is an unconditional event system.*

Proof. Membership in NP is obvious. For the hardness part we use a straightforward reduction from SAT [21, p. 259]. Let $X = \{x_1, \ldots, x_n\}$ be a set of boolean variables and let $C = \{c_1, \ldots, c_m\}$ be a set of clauses over X. Define an event system Θ as follows:

$$\begin{aligned}
\mathcal{P} &= X \cup C \\
\mathcal{E} &= \{\epsilon_{-n}, \ldots, \epsilon_n\} \\
\mathcal{R} &= \{\langle \epsilon_0, C, X, \emptyset \rangle\} \cup \\
&\quad \{\langle \epsilon_i, \{x_i\}, \{c_j \in C | x_i \in c_j\}, \{x_i\} \rangle | \, x_i \in X\} \cup \\
&\quad \{\langle \epsilon_{-i}, \{x_i\}, \{c_j \in C | \neg x_i \in c_j\}, \{x_i\} \rangle | \, x_i \in X\} \\
\mathcal{A} &= \{e_{-n}, \ldots, e_n\} \\
type(e_k) &= \epsilon_k \text{ for all } e_k \in \mathcal{A}, \epsilon_k \in \mathcal{E} \\
I &= X
\end{aligned}$$

It is obvious that the set of clauses C is satisfiable iff there exists a complete event sequence over Θ that is admissible. ■

References

[1] *Proceedings of the 9th National Conference of the American Association for Artificial Intelligence*, Anaheim, CA, July 1991. MIT Press.

[2] James F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26(11):832–843, November 1983.

[3] Christer Bäckström. *Reasoning about Interdependent Actions.* Licentiate thesis, Linköping University, Linköping, Sweden, 1988.

[4] Christer Bäckström. *Computational Complexity of Reasoning about Plans.* Doctoral dissertation, Linköping University, Linköping, Sweden, June 1992.

[5] Christer Bäckström. Equivalence and tractability results for SAS$^+$ planning. In Nebel et al. [28], pages 126–137.

[6] Christer Bäckström and Inger Klein. Parallel non-binary planning in polynomial time. In IJCAI-91 [24], pages 268–273. Extended version available as Research Report LiTH-IDA-R-91-11, Department of Computer and Information Science, Linköping University, Linköping, Sweden.

[7] Christer Bäckström and Inger Klein. Planning in polynomial time: The SAS-PUBS class. *Computational Intelligence*, 7(3):181–197, August 1991.

[8] Christer Bäckström and Bernhard Nebel. Complexity results for SAS$^+$ planning. Research report, German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany, and Department of Computer and Information Science, Linköping University, Linköping, Sweden, 1993. In preparation.

[9] Tom Bylander. Complexity results for planning. In IJCAI-91 [24], pages 274–279.

[10] Tom Bylander. The computational complexity of propositional STRIPS planning. Technical report, Laboratory for Artificial Intelligence Research, Department of Computer and Information Science, Ohio State University, Columbus, OH, 1992.

[11] David Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 32(3):333–377, July 1987.

[12] Stephen V. Chenoweth. On the NP-hardness of blocks world. In AAAI-91 [1], pages 623–628.

[13] Ernest Davis. *Representation of Commonsense Knowledge*. Morgan Kaufmann, San Mateo, CA, 1990.

[14] Thomas L. Dean and Mark Boddy. Incremental causal reasoning. In *Proceedings of the 6th National Conference of the American Association for Artificial Intelligence*, pages 196–201, Seattle, WA, July 1987.

[15] Thomas L. Dean and Mark Boddy. Reasoning about partially ordered events. *Artificial Intelligence*, 36(3):375–400, October 1988.

[16] Kutluhan Erol, Dana S. Nau, and V. S. Subrahmanian. Complexity, decidability and undecidability results for domain-independent planning. Technical Report CS-TR-2797, UMIACS-TR-91-154, Department of Computer Science, University of Maryland, College Park, MD, 1991.

[17] Kutluhan Erol, Dana S. Nau, and V. S. Subrahmanian. On the complexity of domain-independent planning. In *Proceedings of the 10th National Conference of the American Association for Artificial Intelligence*, pages 381–386, San Jose, CA, July 1992. MIT Press.

[18] Kutluhan Erol, Dana S. Nau, and V. S. Subrahmanian. When is planning decidable? In *Proceedings of the 1st International Conference on Artificial Intelligence Planning Systems*, pages 222–227, Washington, D.C., 1992. Morgan Kaufmann.

[19] Richard E. Fikes and Nils Nilsson. STRIPS: A new approach to the application of theorem proving as problem solving. *Artificial Intelligence*, 2:198–208, 1971.

[20] H. N. Gabow, S. N. Maheshwari, and L. Osterweil. On two problems in the generation of program test paths. *IEEE Transactions on Software Engineering*, SE-2, 1976.

[21] Michael R. Garey and David S. Johnson. *Computers and Intractability—A Guide to the Theory of NP-Completeness*. Freeman, San Francisco, CA, 1979.

[22] Naresh Gupta and Dana S. Nau. On the complexity of blocks-world planning. *Artificial Intelligence*, 56:223–254, 1992.

[23] Jochen Heinsohn, Daniel Kudenko, Bernhard Nebel, and Hans-Jürgen Profitlich. RAT: Representation of actions using terminological logics. DFKI Research Report, German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany, 1993. To appear.

[24] *Proceedings of the 12th International Joint Conference on Artificial Intelligence*, Sydney, Australia, August 1991. Morgan Kaufmann.

[25] David McAllester and David Rosenblitt. Systematic nonlinear planning. In AAAI-91 [1], pages 634–639.

[26] Steve Minton, John Bresnia, and Mark Drummond. Commitment strategies in planning: A comparative analysis. In IJCAI-91 [24], pages 259–265.

[27] Steve Minton, Mark Drummond, John L. Bresnia, and Andrew B. Philips. Total order *vs.* partial order planning: Factors influencing performance. In Nebel et al. [28], pages 83–92.

[28] B. Nebel, W. Swartout, and C. Rich, editors. *Principles of Knowledge Representation and Reasoning: Proceedings of the 3rd International Conference*, Cambridge, MA, October 1992. Morgan Kaufmann.

[29] Bernhard Nebel and Christer Bäckström. On the computational complexity of temporal projection and some related problems. Research Report RR-91-34 (DFKI) and LiTH-IDA-R-91-34 (Univ. Linköping), German Research Center for Artificial Intelligence (DFKI), Saarbrücken, Germany, and Department of Computer and Information Science, Linköping University, Linköping, Sweden, October 1991.

[30] Klaus Nökel. Convex relations between time intervals. In J. Rettie and K. Leidlmair, editors, *Proceedings der 5. Österreichischen Artificial Intelligence-Tagung*, pages 298–302. Springer-Verlag, Berlin, Heidelberg, New York, 1989.

[31] Edwin P. D. Pednault. Generalizing nonlinear planning to handle complex goals and actions with context-dependent effects. In IJCAI-91 [24], pages 240–245.

[32] Erik Sandewall and Ralph Rönnquist. A representation of action structures. In *Proceedings of the 5th National Conference of the American Association for Artificial Intelligence*, pages 89–97, Philadelphia, PA, August 1986.

[33] Austin Tate, James Hendler, and Mark Drummond. A review of AI planning techniques. In James F. Allen, James Hendler, and Austin Tate, editors, *Readings in Planning*, pages 26–49. Morgan Kaufmann, San Mateo, CA, 1990.

[34] Marc B. Vilain, Henry A. Kautz, and Peter G. van Beek. Constraint propagation algorithms for temporal reasoning: A revised report. In D. S. Weld and J. de Kleer, editors, *Readings in Qualitative Reasoning about Physical Systems*, pages 373–381. Morgan Kaufmann, San Mateo, CA, 1989.

[35] Wolfgang Wahlster, Elisabeth André, Wolfgang Finkler, Hans-Jürgen Profitlich, and Thomas Rist. Plan-based integration of natural language and graphics generation. *Artificial Intelligence*, 1993. To appear.