Journal of Artificial Intelligence Research 48 (2013) 783-812 Submitted 08/13; published 11/13

The Complexity of Optimal Monotonic Planning: The Bad, The Good, and The Causal Graph

Carmel Domshlak DCARMEL@IE.TECHNION.AC.IL
Anton Nazarenko ANTON.NAZARENKO@GMAIL.COM
Faculty of Industrial Engineering & Management,
Technion - Israel Institute of Technology,
Haifa, Israel

Abstract

For almost two decades, monotonic, or "delete free," relaxation has been one of the key auxiliary tools in the practice of domain-independent deterministic planning. In the particular contexts of both satisficing and optimal planning, it underlies most state-of-the-art heuristic functions. While satisficing planning for monotonic tasks is polynomial-time, optimal planning for monotonic tasks is NP-equivalent. Here we establish both negative and positive results on the complexity of some wide fragments of optimal monotonic planning, with the fragments being defined around the causal graph topology. Our results shed some light on the link between the complexity of general optimal planning and the complexity of optimal planning for the respective monotonic relaxations.

1. Introduction

In domain-independent deterministic (or "classical") planning, the world states are represented by complete assignments to a set of variables, the operators allow for deterministic modifications of these assignments, and the objective is to find a sequence of operators that sequentially modifies a given initial assignment to an assignment that satisfies a certain predefined goal property. In the last two decades, solvers for this problem have made spectacular advances in their empirical efficiency, and this especially in the context of state-space heuristic search planning techniques. This efficiency has been made possible largely by the ability to exploit *monotonic*, or "delete-free," *relaxations* of the planning tasks (McDermott, 1999; Bonet & Geffner, 2001; Hoffmann & Nebel, 2001).

At a high level, monotonic relaxation replaces the regular value switching semantics of planning operators with the value accumulating semantics. That is, if an operator switches the value of a variable v from x to y, then the relaxed version of that operator "extends" the value of v from $\{x\}$ to $\{x, y\}$. The key point here is that applying operators under value accumulating semantics does not reduce the applicability of operators in the future. Two properties of monotonic relaxation make it especially valuable to automated planning. First, while deterministic planning is PSPACE-complete even for rather conservative propositional formalisms, planning for monotonic tasks is polynomial-time (Bylander, 1994), and thus can be exploited for deriving heuristic estimates. Second, in numerous problems of practical interest, plans for monotonic relaxations are not that distant from the true plans for these problems (Hoffmann, 2005; Helmert & Mattmüller, 2007; Helmert & Domshlak, 2009; Bonet & Helmert, 2010). Hence, starting from the seminal HSP (Bonet & Geffner, 2001) and FF (Hoffmann & Nebel, 2001) planning systems, exploiting, and in particular, explicitly planning for, monotonic relaxations became an important ingredient of most systems for domain-independent deterministic planning. (For a comprehensive survey, see, e.g., Betz & Helmert, 2009.)

Ideally, a planner reasoning about the cost of the plans for monotonic relaxations should reason about the cost of *optimal plans for monotonic relaxations*. Unfortunately, while regular planning for monotonic tasks is polynomial-time, optimal planning for these tasks is NP-equivalent (Bylander, 1994), and constant-factor approximations for this problem are provably hard as well (Betz & Helmert, 2009). Still, admissible heuristic estimates can in principle exploit tractable fragments of optimal planning for monotonic relaxations (Katz & Domshlak, 2010). However, to the best of our knowledge, no substantial fragments of tractability have been revealed for monotonic optimal planning to date.

Identifying significant fragments of tractability for optimal monotonic planning is precisely our focus here. Our special interest is in establishing connections between the complexity of optimal planning and that of optimal planning for the monotonic relaxations of the respective planning tasks. This interest is motivated by the new and important role of methods that combine tractable fragments of regular deterministic planning with monotonic relaxation heuristics (Helmert, 2004; Keyder & Geffner, 2008a; Katz & Domshlak, 2010; Katz, Hoffmann, & Domshlak, 2013a, 2013b; Katz & Hoffmann, 2013). In turn, this comparative perspective has brought us to consider planning tasks in terms of *finite-domain* representations that go beyond the standard propositional representation formalisms such as STRIPS (Fikes & Nilsson, 1971) and ADL (Pednault, 1989). Some explanation, and possibly even justification, for this choice of analysis are in place here.

Due to their close relationship to first-order and propositional logics, propositional representations have dominated the area of automated planning since the early days of AI research. For instance, the propositional PDDL language is still a *de facto* standard problem description language in the planning community (Fox & Long, 2003). However, propositional languages blur a lot of important structure that is present in typical planning tasks of interest. As we discuss later on, finite-domain representations (FDR) that go beyond propositional state variables (Bäckström & Klein, 1991; Bäckström & Nebel, 1995; Helmert, 2009) allowed for much deeper and much more discriminative analysis of automated planning complexity. In turn, some of these formal developments have already been translated into practical advances in planning, allowing for the introduction of effective enhancements of monotonic relaxation heuristics (Fox & Long, 2001; Helmert, 2004; Helmert & Geffner, 2008; Keyder & Geffner, 2008a; Cai, Hoffmann, & Helmert, 2013; Katz et al., 2013a, 2013b), abstraction heuristics (Edelkamp, 2001; Helmert, Haslum, & Hoffmann, 2007; Katz & Domshlak, 2010), decomposition-based planning (Nissim, Brafman, & Domshlak, 2010; Nissim & Brafman, 2012), search-topology analysis (Hoffmann, 2011), and many others.

Nonetheless, under the value accumulating semantics of monotonic relaxation, FDR no longer maintains its key advantage over propositional representations, because there no longer seems to be any reason to prefer explicit representation of certain mutual exclusion relationships between the propositions. Therefore, in principle, the results presented in what follows can be phrased, and sometimes even extended, in the context of propositional representations such as STRIPS. Why, then, have we chosen to view the complexity of optimal relaxed planning through the lens of FDR? The primary reason is our interest in *comparative* complexity analysis of optimal planning and optimal planning for the respective monotonic relaxations. Departing from previously discovered fragments of tractability for both satisficing and optimal planning, we approach the following two high-level questions:

1. For what fragments of deterministic planning, if any, is optimal planning hard but optimal (monotonically) relaxed planning easy?

2. For what fragments of deterministic planning, if any, is optimal planning easy yet optimal (monotonically) relaxed planning hard?

For regular planning, the best classification we have these days for its worst-case time complexity exploits the properties of some graphical structures induced by the planning tasks, *together* with various properties of the FDR state variables such as the size of their domains. Hence, discussing optimal relaxed planning for FDR tasks keeps us in direct relation to that well-explored complexity map of regular deterministic planning. Moreover, we show that some known links between the planning complexity and graph-topological properties of the FDR tasks are even stronger in the case of optimal relaxed planning.

The second reason for our choice is that recent work has already revealed interesting interplays between (either complete or partial) monotonic relaxation of finite-domain variables and graphical structures induced by the FDR tasks. The respective results are in the context of computational complexity of non-optimal planning (Katz et al., 2013b), heuristic estimates (Keyder & Geffner, 2008b; Katz et al., 2013a; Katz & Hoffmann, 2013), and search-topology analysis (Hoffmann, 2011). For instance, Hoffmann (2011) showed that, if the causal graph of the FDR task is acyclic, and every variable transition is invertible, then the h^+ heuristic induced by optimal relaxed plans from the evaluated states has no local minima. This result in particular testifies that examining monotonic relaxations through the lens of FDR can lead to some crisp and concisely formulated results. The same can be found in our work here: While we show how some of our results can be easily reformulated, and sometimes even generalized, in terms of STRIPS, some of our other results do not conform so easily to STRIPS reformulation.

Finally, while the immediate value of our results in this work is mostly theoretical, we would like to see them as a step towards exploiting optimal relaxed planning to devise heuristic functions for deterministic planning. Some of our results, such as Theorem 4, can in fact be directly used within the framework of implicit abstractions (Katz & Domshlak, 2010), while some other results can possibly be exploited within various frameworks of partial monotonic relaxation such as, e.g., the recent red-black planning framework of Katz et al. (2013b, 2013a). To what extent this will actually happen remains, of course, to be seen. However, the very focus of both implicit abstractions and red-black planning on finite-domain task representations should make it much easier to assess the relevance of our results to these frameworks.

2. Formalism, Background, and Related Results

We use the notation $[n]$ to refer to the set $\{1, \ldots, n\}$. In directed graphs, the edge from x to y is denoted by (x, y), and in undirected graphs, the edge between x and y is denoted by $\{x, y\}$. By $||x||$ we refer to the representation size of object x, not to be confused with $|x|$, which denotes the number of elements in set x.

2.1 FDR, MFDR, and Monotonic Relaxation

Here we adopt the terminology and notation of Katz et al. (2013b). A planning task in **finite-domain representation (FDR)** is given by a quintuple $\Pi = \langle V, A, I, G, cost \rangle$, where:

- V is a set of *state variables*, with each $v \in V$ being associated with a finite domain $\mathcal{D}(v)$. A partial variable assignment p is a function on a variable subset $\mathcal{V}(p) \subseteq V$ that assigns each $v \in \mathcal{V}(p)$ a value $p[v] \in \mathcal{D}(v)$ of its domain. A partial variable assignment s is called a *state* if $\mathcal{V}(s) = V$.

- I is an *initial state*. The *goal* G is a partial variable assignment to V.

- A is a finite set of *actions*. Each action a is a pair $\langle \mathsf{pre}(a), \mathsf{eff}(a) \rangle$ of partial variable assignments to V called *precondition* and *effect*, respectively.

- $cost : A \to \mathbb{R}^{0+}$ is a real-valued, nonnegative *action cost* function.

Auxiliary notation:

— For a partial assignment p and a variable subset $V' \subseteq \mathcal{V}(p)$, by $p[V']$ we denote the assignment provided by p to V'. For ease of presentation, we sometimes also specify partial assignments as sets of constructs $v \leftarrow d$, in which $v \in V$ and $d \in \mathcal{D}(v)$.

— For a variable $v \in V$, by $A_v \subseteq A$ we denote the actions affecting the value of v, that is, $A_v = \{a \mid v \in \mathcal{V}(\mathsf{eff}(a))\}$. For a sequence of actions ρ and a state variable $v \in V$, by $\rho_{\downarrow v}$ we denote the restriction of ρ to the actions in A_v.

The semantics of FDR tasks is as follows. An action a is applicable in a state s iff $s[v] = \mathsf{pre}(a)[v]$ for all $v \in \mathcal{V}(\mathsf{pre}(a))$. Applying a in state s changes the value of every $v \in \mathcal{V}(\mathsf{eff}(a))$ to $\mathsf{eff}(a)[v]$; the resulting state is denoted by $s[\![a]\!]$. By $s[\![\langle a_1, \ldots, a_k \rangle]\!]$ we denote the state obtained from sequential application of the (respectively applicable) actions $a_1, \ldots, a_k$ starting at state s. Such an action sequence is an *s-plan* if $s[\![\langle a_1, \ldots, a_k \rangle]\!][\mathcal{V}(G)] = G$, and it is an *optimal* s-plan if the sum of its action costs is minimal among all s-plans. The computational task of **(optimal) planning** is finding an (optimal) I-plan. In what follows, (optimal) I-plans are often referred simply as *(optimal) plans for* Π.

A **monotonic finite-domain representation (MFDR)** planning task is given by a quintuple $\Pi = \langle V, A, I, G, cost \rangle$ exactly as for FDR tasks, but the semantics is different.[1] Informally, in MFDR tasks the state variables accumulate their values, rather than switching between them. More specifically, an MFDR state s is a function that assigns each $v \in V$ a non-empty subset $s[v] \subseteq \mathcal{D}(v)$ of its domain. An MFDR action a is applicable in state s iff $\mathsf{pre}(a)[v] \in s[v]$ for all $v \in \mathcal{V}(\mathsf{pre}(a))$. Applying an MFDR action a in state s changes the value of $v \in \mathcal{V}(\mathsf{eff}(a))$ from $s[v]$ to $s[v] \cup \{\mathsf{eff}(a)[v]\}$. Respectively, an MFDR action sequence $\langle a_1, \ldots, a_k \rangle$ applicable in state s is an s-plan if $G[v] \in s[\![\langle a_1, \ldots, a_k \rangle]\!][v]$ for all $v \in \mathcal{V}(G)$. In all other respects, MFDR and FDR semantics are identical.

1. It is not entirely clear to whom the original formulation of monotonic relaxation for multi-valued variable domains should be attributed, but it can be traced back at least to the work of Helmert (2006) on the Fast Downward planning system.

While FDR planning is PSPACE-complete even for propositional state variables, planning for MFDR tasks is polynomial time (Bylander, 1994). Starting with the HSP (Bonet & Geffner, 2001) and FF (Hoffmann & Nebel, 2001) planning systems, exploiting this attractive property of MFDR for deriving heuristic estimates via the notion of monotonic relaxation became a key ingredient of many planning systems. Given an FDR planning task $\Pi = \langle V, A, I, G \rangle$, the **monotonic relaxation** of Π is the MFDR task $\Pi^+ = \Pi$. For any state s of Π, the *optimal relaxation heuristic* $h^+(s)$ is defined to be the cost of the optimal plan for the MFDR task $\langle V, A, s, G \rangle$, and by **(optimal) relaxed planning** for Π we refer to (optimal) planning for Π^+. If ρ^+ is a plan for Π^+, then ρ^+ is referred to as a **relaxed plan** for Π.

Finally, for both FDR and MFDR, we sometimes distinguish between planning tasks in terms of a pair of standard graphical structures induced by the description of these tasks.

- The **causal graph** CG_Π of Π is a digraph over nodes V. An arc (v, v') is in CG_Π iff $v \neq v'$ and there exists an action $a \in A$ such that $(v, v') \in \mathcal{V}(\mathsf{eff}(a)) \cup \mathcal{V}(\mathsf{pre}(a)) \times \mathcal{V}(\mathsf{eff}(a))$. In this case, we say that (v, v') is *induced* by a. By $\mathsf{succ}(v)$ and $\mathsf{pred}(v)$ we respectively denote the sets of immediate successors and predecessors of v in CG_Π.

- The **domain transition graph** $DTG(v, \Pi)$ of a variable $v \in V$ is an arc-labeled digraph over the nodes $\mathcal{D}(v)$ such that an arc (d, d') labeled with $\mathsf{pre}(a)[V \setminus \{v\}]$ and $cost(a)$ belongs to the graph iff both $\mathsf{eff}(a)[v] = d'$, and either $\mathsf{pre}(a)[v] = d$ or $v \notin \mathcal{V}(\mathsf{pre}(a))$.

2.2 Causal Graph Treewidth and Planning Complexity

Introduced by Halin (1976), tree-width went unnoticed until it was independently rediscovered by Robertson and Seymour (1984) and Arnborg, Cornell, and Proskurowski (1987). It has since received widespread attention due to its numerous graph-theoretic and algorithmic applications. Informally, the tree-width of a graph is a measure of how close the structure of the graph is to a tree. For example, the tree-width of a tree is 1, regardless of its size, whereas the tree-width of a complete graph over n nodes is $n-1$. Formally, the tree-width of a graph is defined via the notion of tree decomposition as follows.

A *tree decomposition* of a connected undirected graph $G = (V, E)$ is a pair $\langle T, \alpha \rangle$, where $T = (V^T, E^T)$ is a tree, i.e., a connected acyclic graph, and $\alpha : V^T \mapsto 2^V$ such that:

1. For every $v \in V$, the set $\{t \in V^T \mid v \in \alpha(t)\}$ in T is non-empty and connected.

2. For every $(v, u) \in E$ there is a $t \in V^T$ such that $\{v, u\} \subseteq \alpha(t)$.

The width of a tree decomposition $\langle T, \alpha \rangle$ of G is $\max\{|\alpha(t)| \mid t \in V^T\} - 1$, and the **tree-width** of G, $tw(G)$, is the minimum width over all tree decompositions of G. Following what appears to be standard terminology, by tree-width of a digraph G we refer to the tree-width of the undirected graph induced by G (Berwanger, Dawar, Hunter, Kreutzer, & Obdržálek, 2012).

With the development of parametrized complexity analysis (Downey & Fellows, 1999; Flum & Grohe, 2006), it has been shown that many NP-hard problems can be solved in polynomial time when restricted to induce certain problem-specific graphical structures of a fixed tree-width. In particular, constraint satisfaction and constraint optimization problems over finite-domain variables can be solved in time polynomial in the size of the explicit description of these problems, and exponential only in the tree-width of the induced constraint graph (Dechter, 2003). Since the causal graph captures a high-level structure of the planning problems, one would expect its tree-width to play a similar role in the worst-case time complexity of both satisficing and optimal FDR planning. Unfortunately, the results in this direction have mostly been negative.

- Under a standard assumption on parametric complexity hierarchy that $\mathsf{W}[1] \not\subseteq \mathsf{nu\text{-}FPT}$ (Flum & Grohe, 2006), Chen and Gimenez (2010) proved that, for any family of digraphs $\mathcal{C}$, FDR planning for tasks inducing causal graphs in $\mathcal{C}$ is polynomial-time if and only if the size of connected components in $\mathcal{C}$ is bounded by a constant. As the family of all digraphs with tree-width of 1 trivially fails to satisfy the latter condition, the immediate corollary of this result is that even satisficing FDR planning restricted to causal graphs with the tree-width of 1 is not polynomial.

- While the construction in the proof of Chen and Gimenez (2010) uses FDR tasks with variable domains of parametric size, the work of Gimenez and Jonsson (2009b) shows that the negative result for causal graphs with the tree-width of 1 holds even if restricted to such planning tasks with fixed variable domains. Specifically, Gimenez and Jonsson show that FDR planning over chain causal graphs is NP-hard even if restricted to variables with domains of size 5.

These negative results on the role of the causal graph's tree-width in computational tractability of FDR planning are strong, but apparently tell only part of the story. As shown by Brafman and Domshlak (2006, 2013), the causal graph's tree-width does play a role in the worst-case time complexity of FDR planning, but in a tight interplay with another parameter called the task's *local depth*. Informally, local depth of an FDR task Π captures a minmax amount of work required on a single variable in order to solve Π. Since later we refer to this result of Brafman and Domshlak, a precise specification of local depth is warranted here: denoting by $Plans(\Pi)$ the (possibly infinite) set of all plans for an FDR task Π, the local depth of Π is

$$\delta_\Pi = \min_{\rho \in Plans(\Pi)} \max_{v \in V} \{|\rho_{\downarrow v}|\},$$

that is, δ_Π is the maximal number of value changes of a single state variable, along a plan that minimizes that quantity among the plans for Π. By Theorem 6 of Brafman and Domshlak (2013), FDR tasks Π can be solved in time polynomial in $||\Pi||$ and exponential only in $O(tw(CG_\Pi) \cdot \delta_\Pi)$. Up to some possible stratifications based on succinct representation of internal variable domain dynamics, such as those suggested by Fabre, Jezequel, Haslum, and Thiébaux (2010), this appears to be the strongest link discovered so far between the complexity of general FDR planning and the graph-topological properties of the causal graphs. Note also that this positive result applies only to satisficing planning; it is applicable to optimal planning only in very limited settings (Fabre et al., 2010; Brafman & Domshlak, 2013).

3. Negative Results: The Bottleneck of Variable Domains

Our focus here is on connections between the worst-case time complexity of optimal relaxed planning and the structure of the problems' causal graphs. Note that the causal graphs of FDR tasks are trivially invariant under monotonic relaxation: since $\Pi^+ = \Pi$, we have $CG_{\Pi^+} = CG_{\Pi}$. As we just mentioned, previous works have already revealed certain connections between the structure of the causal graphs, and in particular its tree-width, and the complexity of FDR planning. In what follows, we show that this link is even stronger and somewhat more intriguing in the case of optimal relaxed planning. Having said that, we begin with a set of *negative* results which, at least at first glance, suggest that no such link is actually likely.

Definition 1 *A connected digraph $G = (N, E)$ is a* **fork** *if it contains exactly one node $r \in N$ with non-zero out-degree, that is, $E = \{(r, n) \mid n \in N \setminus \{r\}\}$. Similarly, G is an* **inverted fork** *if it contains exactly one node $r \in N$ with non-zero in-degree, that is, $E = \{(n, r) \mid n \in N \setminus \{r\}\}$. The respective special nodes r in fork and inverted-fork graphs are called the* roots *of the graphs.*

Considering FDR planning tasks with fork and inverted fork causal graphs, the first impression might be that these FDR fragments are restricted enough to allow for polynomial-time planning. This, however, is not the case: even non-optimal planning for FDR tasks with such simple causal graphs is hard, and this even if all variables but the roots are further restricted to be binary-valued (Domshlak & Dinitz, 2001). On the other hand, especially since non-optimal relaxed planning for FDR is polynomial-time, these results have no direct influence on the complexity of optimal relaxed planning for the respective FDR fragments. Nonetheless, surprisingly or not, this problem *is* hard.

Theorem 1 *Optimal relaxed planning is* **NP-equivalent** *even if restricted to* FDR *tasks with fork and inverted-fork structured causal graphs. Moreover, the result holds even if all state variables but the roots are restricted to binary domains.*

Proof: The proof is by polynomial reductions from the NP-equivalent problems of (minimum) Directed Steiner Tree and (minimum) Set Cover (Karp, 1972).

Directed Steiner Tree**:** Given a digraph $\mathcal{G} = (N, E)$ with arc weights $w : E \to \mathbb{R}^{0+}$, a set of terminals $Z \subset N$, and a root vertex n_r, find a minimum weight arborescence (directed tree) T rooted in $n_r \in N$ such that all terminals Z are included in T.

Set Cover**:** Given a collection C of subsets of a finite set S, find a minimum cardinality subset $C' \subseteq C$ such that every element of S belongs to at least one member of C'.

Fragment I: Given a Directed Steiner Tree problem $\langle \mathcal{G} = (N, E), w, Z, n_r \rangle$, the corresponding fork-structured FDR task $\Pi = \langle V, A, I, G, cost \rangle$ is constructed as follows. The variable set V contains a variable per terminal node in $\mathcal{G}$, plus an extra variable r, that is, $V = \{r\} \cup V_Z$ where $V_Z = \{v_z \mid z \in Z\}$. The domain of r, $\mathcal{D}(r) = N$, corresponds to the nodes of $\mathcal{G}$, and all other variables are binary-valued, with $\mathcal{D}(v_z) = \{0, 1\}$. In the initial state, $I[r] = n_r$ and $I[v] = 0$ for all $v \in V_Z$. The goal is to achieve value 1 for all $v \in V_Z$. For each arc $e = (x, y) \in E$, the action set A contains a root-changing action

a_e with $\mathsf{pre}(a_e) = \{r \leftarrow x\}$, $\mathsf{eff}(a_e) = \{r \leftarrow y\}$, and $cost(a_e) = w(e)$. Likewise, for each terminal $z \in Z$, A contains a v_z-changing action a_z with $\mathsf{pre}(a_z) = \{r \leftarrow z, v_z \leftarrow 0\}$, $\mathsf{eff}(a_z) = \{v_z \leftarrow 1\}$, and $cost(a_z) = 0$. This construction is clearly polynomial, the causal graph of Π forms a fork rooted at r, and the variable domains in Π are as required by the theorem. It also holds that:

(i) For any *relaxed* plan ρ for Π, the set of arcs $\{e \mid a_e \in \rho_{\downarrow r}\}$ in $\mathcal{G}$ induces a connected sub-graph $\mathcal{G}'$ containing n_r such that all terminals Z are included in $\mathcal{G}'$ (or otherwise at least one of the leaf variables could not have been changed by ρ to its goal value). Likewise, there is a directed path in $\mathcal{G}'$ from n_r to every other node n in $\mathcal{G}'$ (or otherwise the respective value n of the root variable r could not have been achieved along ρ). Hence, in particular, $\mathcal{G}'$ contains an arborescence rooted in n_r that includes all terminals Z.

(ii) Vice versa, let T be an arborescence of $\mathcal{G}$ rooted in n_r that includes all terminals $Z = \{z_1, \dots, z_m\}$, and let $\{e_1, \dots, e_k\}$ be a topological ordering of the arcs of T. Then $\langle a_{e_1}, \dots, a_{e_k}, a_{z_1}, \dots, a_{z_m}\rangle$ is a relaxed plan for Π, and the cost of ρ is precisely the weight of T.

Hence, optimal relaxed plans for Π induce minimum directed Steiner trees for $\langle \mathcal{G} = (N, E), w, Z, n_r\rangle$, and vice versa.

Fragment II: Given a Set Cover problem $\langle S, C\rangle$ with $S = \{1, 2, \dots, m\}$ and $|C| = n$, the corresponding inverted-fork structured FDR task $\Pi = \langle V, A, I, G, cost\rangle$ is constructed as follows. The variable set V contains a variable per member of C, plus an extra variable r, that is, $V = \{r\} \cup \{v_c \mid c \in C\}$. The domain of r is $\mathcal{D}(r) = \{0\} \cup S$, and all other variables are binary-valued, with $\mathcal{D}(v_c) = \{0, 1\}$. In the initial state, $I[v] = 0$ for all $v \in V$. The goal is to achieve value m for the special variable r. For $i \in S$, and each subset $c \in C$ such that $i \in c$, the action set A contains a root-changing action $a_{i;c}$ with $\mathsf{pre}(a_{i;c}) = \{r \leftarrow (i-1), v_c \leftarrow 1\}$, $\mathsf{eff}(a_{i;c}) = \{r \leftarrow i\}$, and $cost(a_{i;c}) = 0$. Likewise, for each $c \in C$, A contains a v_c-changing action a_c with $\mathsf{pre}(a_c) = \{v_c \leftarrow 0\}$, $\mathsf{eff}(a_c) = \{v_c \leftarrow 1\}$, and $cost(a_c) = 1$. The construction is polynomial, the causal graph of Π forms an inverted fork rooted at r, and the variable domains in Π are as required by the theorem. Note that, due to the chain-like structure of the domain transition graphs in Π, there is no difference between the plans for Π and for Π^+, and it is easy to verify that any plan for Π induces a cover of S of the same cost, and vice versa. Hence, optimal relaxed plans for Π induce minimum set covers for $\langle S, C\rangle$, and vice versa. $\square$

Corollary 1 *Optimal relaxed planning is* **NP-equivalent** *even if restricted to* FDR *tasks with the causal-graph tree-width of* 1.

This corollary is immediate from Theorem 1 as the undirected graphs induced by both forks and inverted forks are a special case of trees and thus have tree-width of 1. At first glance, the message of Corollary 1 is discouraging with respect to our agenda: the structure of the causal graphs does not seem to play a major role in the complexity of optimal relaxed planning for FDR. Our next result, however, seems to be even more discouraging with respect to the prospects of tractability of optimal relaxed planning for FDR.

Theorem 2 *Optimal relaxed planning is* **NP-equivalent** *even if restricted to* FDR *tasks with two state variables.*

Proof: The proof is by a polynomial reduction from the (minimum) directed Steiner tree problem, and in fact, the proof is very similar to that of the fork case of Theorem 1. Given a Directed Steiner Tree problem $\langle \mathcal{G} = (N, E), w, Z, n_r \rangle$ with $Z = \{z_1, \dots, z_m\}$, we compile it to an FDR task $\Pi = \langle \{v_1, v_2\}, A, I, G, cost \rangle$ as follows. The domain of v_1 corresponds to the nodes of $\mathcal{G}$, and the domain of v_2 corresponds to the terminal nodes and the root node, that is, $\mathcal{D}(v_1) = N$ and $\mathcal{D}(v_2) = Z \cup \{n_r\}$. In the initial state, both $I[v_1] = n_r$ and $I[v_2] = n_r$, and the goal is to achieve value z_m for v_2. For each arc $e = (x, y) \in E$, the action set A contains a v_1-changing action a_e with $\mathsf{pre}(a_e) = \{v_1 \leftarrow x\}$, $\mathsf{eff}(a_e) = \{v_1 \leftarrow y\}$, and $cost(a_e) = w(e)$. Likewise, denoting n_r by z_0, for $1 \le i \le m$, A contains a v_2-changing action a_{z_i} with $\mathsf{pre}(a_{z_i}) = \{v_1 \leftarrow z_i, v_2 \leftarrow z_{i-1}\}$, $\mathsf{eff}(a_{z_i}) = \{v_2 \leftarrow z_i\}$, and $cost(a_{z_i}) = 0$. The construction is polynomial, and its correctness stems from analysis identical to that of the proof of Theorem 1. $\square$

Theorem 2 shows that even the *dimensionality* of the FDR state spaces plays a secondary role, if any, in the complexity of optimal relaxed planning. By that, however, it answers one of the two macro-questions on our agenda:

Corollary 2 *There exist fragments of* FDR *for which optimal planning is polynomial-time, while optimal relaxed planning is NP-equivalent.*

This corollary is immediate from Theorem 2 and polynomial-time solvability of optimal planning for FDR tasks with a *fixed* number of state variables.

4. Positive Results I: State Variables with Fixed-Size Domains

Depending on the reader's background and intuitions, Theorem 2 can either surprise or seem somewhat predictable. In any case, it was Theorem 2 that led us to consider a different (and this time fruitful) fragmentation of optimal relaxed planning.

A closer look at Theorem 1 and Lemma 2 reveals that the *size of the* FDR *variable domains* might be crucial in the complexity of optimal relaxed planning: While the proofs of these two claims rely heavily on the parametric domain size of some of the state variables, these proofs also imply that optimal relaxed planning is hard even if only a single FDR variable comes with a parametric domain size. Departing from this observation, we now show that, after all, the topology of the causal graph *does* play an interesting role in worst-case time complexity classification of optimal relaxed planning for FDR tasks. In particular, it turns out that bounding the tree-width of the causal graph by a constant is all it takes to achieve polynomial-time optimal relaxed planning for FDR tasks with fixed-size variable domains.

Theorem 3 *For any family of directed graphs $\mathcal{C}$, if the tree-width in $\mathcal{C}$ is bounded by a constant, then optimal relaxed planning for* FDR *tasks with fixed-size variable domains and causal graphs in $\mathcal{C}$ is* **polynomial-time**.

Proof: The proof of Theorem 3 is inspired by and closely resembles the approach of Brafman and Domshlak (2013) discussed in Section 2.2. Given an FDR task $\Pi = \langle V, A, I, G, cost\rangle$, we compile it into a constraint optimization problem $\mathsf{COP}_{\Pi^+} = (\mathcal{X}, \Phi)$ over finite-domain variables $\mathcal{X}$, functions Φ, and the global objective to minimize $\Sigma_{\varphi\in\Phi}\varphi(\mathcal{X})$. If Π^+ is unsolvable, then all the assignments to $\mathcal{X}$ evaluate the objective function to ∞. Otherwise, the optimum of the objective is obtained on and only on the assignments to $\mathcal{X}$ that correspond to optimal plans for Π^+, that is, to optimal relaxed plans for Π.

Let $|V| = n$, $\Delta = \max_{v\in V} |\mathcal{D}(v)|$, and recall that $\mathsf{pred}(v)$ for state variable $v \in V$ denotes the set of v's immediate predecessors in the causal graph.

- For each state variable $v \in V$, $\mathcal{X}$ contains a variable x_v that represents the choice of a subset of actions from A_v to participate in the plan we are looking for. These possible choices form the domain $\mathscr{D}(x_v)$ of x_v, and each such choice is represented by a set of size smaller or equal to Δ, with each element of that set being a quadruple

$$(d, \mathsf{id}, a, t),$$

with $d \in \mathcal{D}(v)$, $\mathsf{id} \in \{v\} \cup \mathsf{pred}(v)$, $t \in \{1, \dots, n\Delta\}$, and, if $\mathsf{id} = v$, then $a \in \{a' \in A_v \mid \mathsf{eff}(a')[v] = d\}$ and otherwise $a = \perp$. At a high level, if we view state variables as active decision makers, then d is a value that the relaxed variable v aims to achieve and accumulate at time point t, either by itself using action a, or by delegating this task to another variable id. As we show later on, no variable should accumulate more than Δ values, and since the accumulated values are never lost, optimal plans for Π^+ cannot be longer than $n\Delta$ actions.

- For each state variable $v \in V$, Φ contains a non-negative, extended real-valued function φ_v from $\mathscr{D}(x_v)$. Likewise, for each pair of state variables $\{v, w\}$ such that the causal graph CG_Π contains either arc (v, w) or arc (w, v), Φ contains an indicator function $\varphi_{v,w} : \mathscr{D}(x_v) \times \mathscr{D}(x_w) \to \{0, \infty\}$. To simplify the specification of $\varphi_{v,w}$, we define a set of auxiliary constraints as follows.

(S1) [Precondition Constraint] An assignment $\langle\theta_{v_1}, \dots, \theta_{v_n}\rangle$ to $\mathcal{X}$ satisfies S1 iff, for all $v \in V$, $(d, v, a, t) \in \theta_v$ implies that, for each $w \in \mathcal{V}(\mathsf{pre}(a))$,

$$\mathsf{pre}(a)[w] \in \{I[w]\} \cup \{d' \mid (d', \cdot, \cdot, t') \in \theta_w, t' < t\}. \tag{1}$$

(S2) [Delegation Constraint] An assignment $\langle\theta_{v_1}, \dots, \theta_{v_n}\rangle$ to $\mathcal{X}$ satisfies S2 iff, for all $v \in V$, $(d, w, \perp, t) \in \theta_v$ implies that, for some $a \in A_v \cap A_w$ with $d \in \mathsf{eff}(a)[v]$, $(\cdot, w, a, t) \in \theta_w$.

(S3) [Goal Achievement Constraint] An assignment $\langle\theta_{v_1}, \dots, \theta_{v_n}\rangle$ to $\mathcal{X}$ satisfies S3 iff, for all $v \in \mathcal{V}(G)$,

$$G[v] \in \{I[v]\} \cup \{d \mid (d, \cdot, \cdot, \cdot) \in \theta_v\}. \tag{2}$$

Constraint S1 ensures that preconditions of actions to which a variable is committed are provided on time. Constraint S2 ensures that the "outsourced" value achievements are fulfilled at the required time points. Finally, constraint S3 simply verifies that the value of v induced by θ_v is a goal value. Importantly, S3 corresponds to a set of n *unary* constraints, and S1 and S2 can both be represented as a set of *binary* constraints over $\mathcal{X}$. Given that, functions Φ are specified as

$$\varphi_v(\theta_v) = \sum_{(\cdot,v,a,\cdot)\in\theta_v} cost(a),$$
$$\varphi_{v,w}(\theta_v,\theta_w) = \begin{cases} 0, & \{\theta_v,\theta_w\} \text{ satisfies S1}(x_v,x_w),\ \text{S2}(x_v,x_w),\ \text{S3}(x_v) \text{ and S3}(x_w) \\ \infty, & \text{otherwise} \end{cases}, \tag{3}$$

where $\text{S1}(x_v, x_w)$, $\text{S2}(x_v, x_w)$, and $\text{S3}(x_v)$ correspond to the binary and unary constraints induced respectively by S1, S2, and S3 on the COP variables x_v and x_w.

Let us now take a closer look at COP_{Π^+} constructed as above for the problems in the scope of Theorem 3.

(1) The constraint network of COP_{Π^+} corresponds to the undirected graph induced by the causal graph CG_{Π^+} ($= CG_{\Pi}$). Hence, since the tree-width of the latter is bounded by a constant by the scope of the theorem, so is the tree-width of the constraint network of COP_{Π^+}. While finding an optimal tree decomposition of a graph $\mathcal{G}$ is NP-hard, a tree decomposition of $\mathcal{G}$ with width $c \cdot tw(\mathcal{G})$ for a low constant c can be found in time polynomial in the size of $\mathcal{G}$ (Robertson & Seymour, 1991; Becker & Geiger, 1996; Amir, 2010). Hence, COP_{Π^+} can be solved in time polynomial in the size of its representation using the standard message-passing algorithm for constraint optimization over trees (Dechter, 2003).

(2) Recall that the values of the COP variable x_v are sets of quadruples (d, id, a, t) of size $\leq \Delta$. Then, the size of the x_v's domain $\mathscr{D}(x_v)$ is upper-bounded by

$$(\underbrace{|\mathcal{D}(v)|}_{d} \cdot \underbrace{n}_{\mathsf{id}} \cdot \underbrace{(|A_v|+1)}_{a} \cdot \underbrace{(n\Delta)}_{t})^{\Delta}, \tag{4}$$

and since $\Delta = O(1)$, we have $|\mathscr{D}(x_v)| = O(poly(||\Pi||))$. Together with (1), that implies that COP_{Π^+} can be solved in time $O(poly(||\Pi||))$.

We now prove the correctness of this compilation by showing that, if Π^+ is unsolvable, then all the assignments to $\mathcal{X}$ evaluate the objective function $\Sigma_{\varphi\in\Phi}\varphi(\mathcal{X})$ to ∞, and otherwise, the objective is minimized on and only on the assignments to $\mathcal{X}$ that correspond to optimal plans for Π^+.

First, given an assignment $\boldsymbol{\theta} = \langle\theta_{v_1}, \ldots, \theta_{v_n}\rangle$ to $\mathcal{X}$ such that $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta}) < \infty$, we show that $\boldsymbol{\theta}$ induces a valid plan $\rho_{\boldsymbol{\theta}}$ for Π^+ of cost $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta})$. Note that, since $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta}) < \infty$, by Eq. 3 we have $\boldsymbol{\theta}$ satisfying constraints S1-S3.

Consider the multi-set $Z = \bigcup_{v\in V}\{(a,t) \mid (\cdot, v, a, t) \in \theta_v\}$ induced by $\boldsymbol{\theta}$, and let

$$Z = \{(a_1, t_1), \ldots, (a_m, t_m)\}, \ \ m = |Z|,$$

be an arbitrary ordering of Z such that, for $1 \leq j < i \leq m$, it holds that $t_j \leq t_i$. For $1 \leq i \leq m$, let v be the variable "in charge of performing the action a_i" in $\boldsymbol{\theta}$, that is, $(\mathsf{eff}(a_i)[v], v, a_i, t_i) \in \theta_v$. For each $w \in \mathcal{V}(\mathsf{pre}(a_i))$, Eq. 1 in constraint S1 implies that either $\mathsf{pre}(a_i)[w] = I[w]$ or there is some $(\mathsf{pre}(a_i)[w], \mathsf{id}, a, t) \in \theta_w$ with $t < t_i$. In the latter case, if $\mathsf{id} = w$, then by the construction of $\rho_{\boldsymbol{\theta}}$, $a = a_j$ for some $j < i$, and by the definition of $\mathscr{D}(x_w)$, $\mathsf{pre}(a_i)[w] = \mathsf{eff}(a_j)[w]$. Otherwise, if $\mathsf{id} = w'$ for some $w' \neq w$, then by the definition of S2, $\mathsf{pre}(a_i)[w] = \mathsf{eff}(a')[w]$ for some $(\cdot, w', a', t) \in \theta_{w'}$. Thus, $a' = a_j$ for some $j < i$, and $\mathsf{pre}(a_i)[w] = \mathsf{eff}(a_j)[w]$. Therefore, the action sequence

$$\rho_{\boldsymbol{\theta}} = \langle a_1, \ldots, a_m \rangle$$

is applicable in the initial state I of Π^+, and given that, $\boldsymbol{\theta}$ satisfying S3 implies that $\rho_{\boldsymbol{\theta}}$ is a plan for Π^+. Finally, it is immediate from the construction of $\rho_{\boldsymbol{\theta}}$ and Eq. 3 that $cost(\rho_{\boldsymbol{\theta}}) = \Sigma_{\varphi \in \Phi} \varphi(\boldsymbol{\theta})$.

We now show that any optimal plan $\rho = \langle a_1, \ldots, a_m \rangle$ for Π^+ induces an assignment $\boldsymbol{\theta}_\rho = \langle \theta_{v_1}, \ldots, \theta_{v_n} \rangle$ to $\mathcal{X}$ such that $\Sigma_{\varphi \in \Phi} \varphi(\boldsymbol{\theta}_\rho) = cost(\rho)$. By the definition of MFDR, for $1 \leq i \leq m$ and for each state variable $v \in V$, $I[\![a_1, \ldots, a_{i-1}]\!][v] \subseteq I[\![a_1, \ldots, a_i]\!][v]$. By optimality of ρ, for $1 \leq i \leq m$, there exists at least one variable $v \in V$ for which $I[\![a_1, \ldots, a_{i-1}]\!][v] \subset I[\![a_1, \ldots, a_i]\!][v]$. In particular, that implies that actions $\{a_1, \ldots, a_m\}$ are all different, and for each variable $v \in V$, if

$$\rho/v \triangleq \{a_i \mid I[\![a_1, \ldots, a_{i-1}]\!][v] \subset I[\![a_1, \ldots, a_i]\!][v]\},$$

then[2] $|\rho/v| \leq \Delta$.

Adopting an arbitrary ordering $\{v_1, \ldots, v_n\}$ of the state variables V, for $1 \leq j \leq n$, let $\rho/v_j = \{a_{j_1}, \ldots, a_{j_{m_j}}\}$. In turn, for $1 \leq l \leq m_j$, let $d_{j_l} \in \mathcal{D}(v_j)$ be the value achieved and accumulated for v_j by the action a_{j_l}, i.e., $I[\![a_1, \ldots, a_{j_l}]\!][v_j] \setminus I[\![a_1, \ldots, a_{j_l - 1}]\!][v_j] = \{d_{j_l}\}$. For $1 \leq l \leq m_j$, if $a_{j_l} \notin \bigcup_{k=1}^{j-1} \rho/v_k$, then θ_{v_j} is set to contain $(d_{j_l}, v_j, a_{j_l}, j_l)$, and otherwise, θ_{v_j} is set to contain $(d_{j_l}, v_k, \perp, j_l)$ for $k = \min\{k' \mid a_{j_l} \in \rho/v_{k'}\}$.

By the construction of $\boldsymbol{\theta}_\rho$, each action a from ρ is present in the value of exactly one variable x_v, and by Eq. 3, $\varphi_v(\theta_v)$ sums up the cost of a exactly once. Hence, if $\boldsymbol{\theta}_\rho$ satisfies the constraints S1-S3 that are enforced by the step functions $\varphi_{v,w}$, we have $\Sigma_{\varphi \in \Phi} \varphi(\boldsymbol{\theta}_\rho) = cost(\rho)$. The former also directly follows from the construction of $\boldsymbol{\theta}_\rho$. For $1 \leq j \leq n$, let $\theta_{v_j} = \{(d_{j_l}, \mathsf{id}_{j_l}, a_{j_l}, t_{j_l})\}_{l=1}^{m_j}$. By the definition of the values d_{j_l} as above, the set of values $\{d_{j_1}, \ldots, d_{j_{m_j}}\}$ is exactly the set of values of v_j that gets accumulated by the relaxed plan ρ from I, and thus satisfaction of S3 follows from ρ being a plan for Π^+. Again, by the construction of $\boldsymbol{\theta}_\rho$, the sequence of time points $\{t_{j_1}, \ldots, t_{j_{m_j}}\}$ corresponds to the time points of the first achievements of $\{d_{j_1}, \ldots, d_{j_{m_j}}\}$, respectively, along ρ, and for each such first achiever a_{j_l} along ρ, it is captured and properly scheduled either by θ_{v_i} or by a neighbor of v_i in the causal graph. Hence, the constraints S1 and S2 are all satisfied as well. This finalizes the proof for the compilation correctness, and thus of Theorem 3. $\square$

Note that Theorem 3 in particular answers the second macro-question on our agenda:

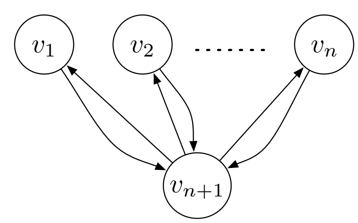
Corollary 3 *There exist fragments of* FDR *for which (even satisficing) planning is NP-equivalent, while optimal relaxed planning is polynomial-time.*

2. That corresponds to the well-known fact that, under our notation, optimal plans in MFDR cannot be longer than $\sum_{v \in V} |\mathcal{D}(v)| \leq n\Delta$.

This corollary is immediate from Theorem 3 and the discovery of Gimenez and Jonsson (2009b) that FDR planning over chain causal graphs is NP-hard even if restricted to variables with domains of size 5.

Following our "STRIPS vs. FDR" discussion in the introduction, Theorem 3 trivially implies that optimal relaxed planning for STRIPS tasks Π can be done in time polynomial in $||\Pi||$ and exponential only in the tree-width of the causal graph. This is actually an example of a tractability fragment for which one can only benefit from switching to a propositional representation: While the formulation of the result remains the same, the coverage of the result grows because, if the size of the FDR variable domains is bounded by $\Delta = O(1)$, then the tree-width of the causal graph under STRIPS representation is at most Δ times larger than this under FDR. However, it can also be smaller, down to identical. Later, however, we present some results that directly benefit from the finite-domain input representation of the planning tasks.

Our discussion in the remainder of this section addresses the readers familiar with the work of Brafman and Domshlak (2013) in detail. This discussion can be skipped without any loss of continuity.



At first view, the compilation in the proof of Theorem 3 brings to mind the compilation in the algorithm behind the proof of Theorem 6 of Brafman and Domshlak (2013). One might thus naturally ask whether that algorithm cannot be used directly for the proof of our Theorem 3. As it stands, however, the algorithm of Brafman and Domshlak does not yield polynomial-time complexity on those tasks with which Theorem 3 is concerned. To see why, consider an FDR task $\Pi = \langle V, A, I, G, cost\rangle$ with $V = \{v_1, \dots, v_{n+1}\}$, where, for all v_i, $\mathcal{D}(v_i) = \{0, 1\}$, $I[v_i] = 0$, and $G[v_i] = 1$, and $A = \{a_1, \dots, a_n\}$ with $\mathcal{V}(\mathsf{pre}(a_i)) = \emptyset$, $\mathcal{V}(\mathsf{eff}(a_i)) = \{v_i, v_{n+1}\}$, and $\mathsf{eff}(a_i)[v_i] = \mathsf{eff}(a_i)[v_{n+1}] = 1$. The causal graph of Π, depicted above, has tree-width of 1. However, for *any* plan ρ for Π^+, $\max_{v \in V} \{|\rho_{\downarrow v}|\} \geq n$, and the local depth δ_{Π^+} is n. This is because Π^+ cannot be solved without applying each of the n actions A at least once, and each of these actions affects the value of the variable v_{n+1}. Hence, finding even a non-optimal plan for Π^+ using the algorithm of Brafman and Domshlak (2013) will take time exponential in $||\Pi||$.

As a final note, it actually can be shown that the algorithm of Brafman and Domshlak (2013) is optimal and polynomial-time on a sub-class of MFDR tasks like those in Theorem 3 but also restricted to *single-effect* operators. It is *multiple-effect* actions that complicate matters, and require a different algorithmic approach to guarantee planning tractability.

5. Positive Results II: M-unfoldable State Variables

Despite the discouraging results in Section 3, we now return to consider FDR tasks with parametric-size domains. Recall that, while the variable values of monotonic relaxation Π^+ correspond to sets of values of the respective variables in Π, these large variable domains in Π^+ are given implicitly, via the variable domains $\mathcal{D}(v_1), \dots, \mathcal{D}(v_n)$ of Π. This conciseness of representation, however, hides many aspects of the problem structure in Π^+ that otherwise might have be exploited for planning efficiency. In particular, reasoning about the domain transition graphs induced by the FDR tasks has been successfully exploited in complexity analysis of FDR (Jonsson & Bäckström, 1998; Domshlak & Dinitz, 2001; Katz & Domshlak, 2008; Gimenez & Jonsson, 2009a). In contrast, in monotonic relaxations, the true domain transition graphs of Π^+, denoted henceforth as $DTG(v, \Pi^+)$, cannot always be represented explicitly because the number of nodes in these graphs is exponential in $||\Pi||$. However, this is not always, or does not always have to be, the case, and below we focus on planning with such "accessible" monotonic variables.

Definition 2 *Given an* FDR *planning task* $\Pi = \langle V, A, I, G, cost \rangle$*, the* **effective domain** $\mathcal{D}^{\oplus}(v)$ *of* $v \in V$ *in* Π^+ *consists of all value subsets* $\vartheta \in \mathcal{D}^+(v) = 2^{\mathcal{D}(v)} \setminus \emptyset$ *that are reachable from* $\{I[v]\}$ *in* $DTG(v, \Pi^+)$*. That is,* $\vartheta \in \mathcal{D}^{\oplus}(v)$ *iff*

(i) $I[v] \in \vartheta$*, and*

(ii) for each value $d \in \vartheta$*, there is a directed path from* $I[v]$ *to* d *in the unlabeled digraph induced by* $DTG(v, \Pi)$ *such that all the values along that path belong to* ϑ*.*

The elements of $\mathcal{D}^{\oplus}(v)$ *are called the* **effective values** *of* v *in* Π^+*.*

Definition 3 *Let* $\mathbf{\Pi}$ *be an infinite set of* FDR *tasks, and* Γ *be a property of state variables that, for each task* $\Pi \in \mathbf{\Pi}$*, partitions the state variables of* Π *into those that satisfy* Γ *(referred as* Γ*-variables), and those that do not satisfy* Γ*. We say that* Γ*-variables in* $\mathbf{\Pi}$ *are* **monotonically unfoldable (M-unfoldable)** *if there exists an integer* $k \in \mathbb{N}$ *such that, for every task* $\Pi \in \mathbf{\Pi}$ *and every* Γ*-variable* v *of* Π*,* $|\mathcal{D}^{\oplus}(v)| = O(||\Pi||^k)$*.*

Γ in Definition 3 can be any property of FDR state variables, and in particular, any property defined with respect to the tasks' causal graphs, such as "root," "sink," "nodes whose causal graph in-degrees are larger than their causal graph out-degrees," etc. Informally, Γ-variables of a set of FDR tasks $\mathbf{\Pi}$ are M-unfoldable if, for every task $\Pi \in \mathbf{\Pi}$ and every Γ-variable v of Π, the "relevant subgraph" of $DTG(v, \Pi^+)$ can be described *explicitly* in space polynomial in the representation size of Π. For instance, v is trivially M-unfoldable if the size of its domain in Π is bounded by a constant, or even by $O(\log(||\Pi||))$. More generally, let $DTG^{\dagger}(v, \Pi)$ be the digraph obtained from the (labels ignored) domain transition graph $DTG(v, \Pi)$ by unifying parallel edges. It is not hard to verify from Definitions 2 and 3 that v is M-unfoldable if and only if the number of arborescence subgraphs of $DTG^{\dagger}(v, \Pi)$ rooted in $I[v]$, and covering $G[v]$ if $v \in \mathcal{V}(G)$, is $O(\mathrm{poly}(||\Pi||))$.

We now return to consider fork-structured FDR tasks. While in Theorem 1 we considered fork-structured FDR tasks with only root variables being unrestricted, we now consider an "inverse" fragment, corresponding to fork-structured FDR tasks with only root variables being restricted. Optimal FDR planning for such tasks is polynomial-time for $|\mathcal{D}(r)| = 2$ (Katz & Domshlak, 2010), but it is NP-equivalent for $|\mathcal{D}(r)| > 2$ (Katz & Keyder, 2012). In contrast, Theorem 4 below shows that optimal relaxed planning for fork-structured FDR tasks is polynomial-time for a much wider class of root variables. Moreover, a simple observation behind the construction in the proof of Theorem 4 is later generalized to capture a much richer fragment of causal graphs.

Theorem 4 *Optimal relaxed planning is* **polynomial time** *for any set of* FDR *tasks* $\mathbf{\Pi}$ *with fork-structured causal graphs and* M*-unfoldable root variables.*

Proof: Let $\Pi = \langle V, A, I, G, cost \rangle$ be a fork-structured FDR task with root r and $V_{\text{leafs}} = V \setminus \{r\} = \{v_1, \dots, v_n\}$. We assume that goal values are specified for all variables in V_{leafs}; this is because none of the leaves in $V_{\text{leafs}} \setminus \mathcal{V}(G)$ need to change their value at all, and thus they can be schematically removed from the problem. Given such a FDR task Π, an optimal plan for its relaxation Π^+ can be constructed as follows.

- All the effective values $\vartheta \in \mathcal{D}^{\oplus}(r)$ consistent with the goal (that is, $G[r] \in \vartheta$ if $r \in \mathcal{V}(G)$) are processed one by one, independently. For each such effective value ϑ, we extract the following information.
  (1) For the root variable r, we determine a cheapest path from $I[r]$ to ϑ in $DTG(r, \Pi^+)$, with $\pi(\vartheta)$ denoting the action sequence inducing that path.
  (2) For each leaf variable $v \in V_{\text{leafs}}$, we
    (a) schematically remove from the domain transition graph $DTG(v, \Pi)$ all the arcs labeled with actions a that are "not supported by ϑ", that is, actions a with $r \in \mathcal{V}(\mathsf{pre}(a))$ and $\mathsf{pre}(a)[r] \notin \vartheta$, and then
    (b) determine a cheapest path from $I[v]$ to $G[v]$ in that arc-reduced domain transition graph, with $\pi_{\vartheta}(G[v])$ denoting the action sequence inducing that path.
- Return the concatenation of action sequences $\pi(\vartheta^*) \cdot \pi_{\vartheta^*}(G[v_1]) \cdot \dots \cdot \pi_{\vartheta^*}(G[v_n])$ where

$$\vartheta^* = \operatorname*{argmin}_{\vartheta \in \mathcal{D}^{\oplus}(r)} \left[cost(\pi(\vartheta)) + \sum_{i=1}^{n} cost(\pi_{\vartheta}(G[v_i])) \right],$$

  and $cost(\pi)$ for an action sequence π is the sum of the costs of actions along π.

The algorithm is polynomial-time given an explicit description of the effective part of $DTG(r, \Pi^+)$, and thus it is polynomial-time if r is M-unfoldable. Recall that, since the causal graph is acyclic, no action affects more than one variable. The correctness of the algorithm stems from a simple observation that, for any relaxed plan ρ for a fork-structured FDR task Π, $\rho' = \rho_{\downarrow r} \cdot \rho_{\downarrow v_1} \cdot \dots \cdot \rho_{\downarrow v_n}$ is also a relaxed plan for Π, and (trivially) of identical cost. Hence, while searching for an optimal relaxed plan for Π, we can restrict ourselves to plans of the latter form, and it is immediate from the description of the algorithm that it finds the cheapest such plan. □

As an aside, following our "STRIPS vs. FDR" discussion, note that Theorem 4 provides an example for exploiting "value grouping" induced by the FDR representation of the planning tasks. The simple algorithm in the proof exploits the fact that only the actions that change the value of a leaf variable depend on its value, and this prevents restrictions from being placed on either the size of the leaf domains, or on the structure of their domain transition graphs. Of course, Theorem 4 can also be reformulated in terms of STRIPS, yet this would require the respective partition of the propositions to be given/discovered, which is essentially equivalent to starting with something like FDR input in the first place.

While the scope of the tractability result in Theorem 4 is fairly limited in terms of the causal graph structure, the nice property of the sets of optimal relaxed plans for fork-structured FDR tasks, exploited in the proof of Theorem 4, can be generalized to a much wider fragment of causal graphs. In turn, this generalization allows us to provide our next tractability result for a wide fragment of optimal relaxed planning for FDR.

Lemma 1 *Let $\Pi = \langle V, A, I, G, cost\rangle$ be an* FDR *task with a directed acyclic causal graph, and let $\{v_1, \dots, v_n\}$ be an arbitrary topological ordering of V with respect to CG_Π. Then, for any plan ρ for Π^+, $\rho' = \rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_n}$ is also a plan for Π^+.*

Proof: Directed acyclicity of the causal graph in particular implies that no action affects more than one variable. The proof of the lemma stems from combining this property with (i) the fact that ρ' preserves the order of the actions $\rho_{\downarrow v_i}$ as in ρ, and (ii) the core property of monotonic relaxations Π^+ that, for any variable v, any value $d \in \mathcal{D}(v)$, and any state s of the relaxed task Π^+, if $d \in s[v]$, then $d \in s'[v]$ for any s' reachable from s in Π^+.

Since CG_Π forms a DAG and the state variables are ordered according to a topological ordering of CG_Π, $\mathcal{V}(\mathsf{pre}(a)[v_1]) \subseteq \{v_1\}$ for all actions $a \in \rho_{\downarrow v_1}$. Thus, order preservation of $\rho_{\downarrow v_1}$ along ρ' with respect to ρ implies that $\rho_{\downarrow v_1}$ is applicable in I, with $I[\![\rho_{\downarrow v_1}]\!][v_1] = I[\![\rho]\!][v_1]$, and $I[\![\rho_{\downarrow v_1}]\!][v_i] = I[v_i]$ for $i > 1$. Assume now that, for $i \geq 1$, $\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_i}$ is applicable in I, and

$$I[\![\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_i}]\!][v_j] = \begin{cases} I[\![\rho]\!][v_j], & j \leq i \\ I[v_j], & j > i \end{cases}. \tag{5}$$

Together with the topological ordering of V and order preservation of $\rho_{\downarrow v_{i+1}}$ along ρ' with respect to ρ, Eq. 5 implies that $\rho_{\downarrow v_{i+1}}$ is applicable in $I[\![\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_i}]\!]$, and

$$I[\![\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_i}]\!][\![\rho_{\downarrow v_{i+1}}]\!][v_j] = \begin{cases} I[\![\rho]\!][v_j], & j = i+1 \\ I[\![\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_i}]\!][v_j], & \text{otherwise} \end{cases}. \tag{6}$$

Putting Eqs. 5 and 6 together then proves the induction hypothesis, and for $i = n$, Eq. 5 boils down to

$$I[\![\rho']\!] = I[\![\rho_{\downarrow v_1} \cdot \ldots \cdot \rho_{\downarrow v_n}]\!] = I[\![\rho]\!].$$

□

Definition 4 *Let $\mathbf{\Pi}$ be an infinite set of* FDR *tasks. We say that the tasks in $\mathbf{\Pi}$ are* **M-unfoldable** *if all state variables in $\mathbf{\Pi}$ are* M*-unfoldable.*

Theorem 5 *Let $\mathbf{\Pi}$ be an infinite set of* M*-unfoldable* FDR *tasks with directed acyclic causal graphs. If both the tree-width and node in-degree of the causal graphs in $\mathbf{\Pi}$ are bounded by a constant, then optimal relaxed planning for $\mathbf{\Pi}$ is* **polynomial-time**.

Proof: Similarly to the proof of Theorem 3, the proof of Theorem 5 is based on a planning-to-COP compilation. Given an FDR task $\Pi = \langle V, A, I, G, cost\rangle$, we compile its monotonic relaxation Π^+ into a constraint optimization problem $\mathsf{COP}_{\Pi^+} = (\mathcal{X}, \Phi)$ over variables $\mathcal{X}$, functions Φ, and the global objective to minimize $\Sigma_{\varphi \in \Phi} \varphi(\mathcal{X})$ such that, if Π^+ is unsolvable, then all the assignments to $\mathcal{X}$ evaluate the objective function to ∞, and otherwise, the optimum of the objective is obtained on and only on the assignments to $\mathcal{X}$ that correspond to optimal plans for Π^+. The specific construction of COPs here relies on the property of monotonic relaxations of DAG-structured FDR tasks expressed by Lemma 1.

Given an FDR task $\Pi = \langle V, A, I, G, cost \rangle$ as in the theorem, $\mathsf{COP}_{\Pi^+} = (\mathcal{X}, \Phi)$ is specified as follows. For each state variable $v \in V$,

- $\mathcal{X}$ contains a variable x_v with domain $\mathscr{D}(x_v) = \mathcal{D}^{\oplus}(v)$, that is, the effective domain of v in Π^+, and
- Φ contains a non-negative, extended real-valued function φ_v over v and its immediate ancestors in the causal graph, that is, over $\{x_v\} \cup \{x_w \mid w \in \mathsf{pred}(v)\}$.

Assuming an arbitrary fixed ordering $\{w_1, \dots, w_k\}$ of v's immediate ancestors $\mathsf{pred}(v)$, for each $\boldsymbol{\vartheta}_{\mathsf{pred}} \in \mathcal{D}^{\oplus}(\mathsf{pred}(v)) = \mathcal{D}^{\oplus}(w_1) \times \cdots \times \mathcal{D}^{\oplus}(w_k)$, let $DTG(v, \Pi^+|\boldsymbol{\vartheta}_{\mathsf{pred}})$ denote the restriction of $DTG(v, \Pi^+)$ to only edges "supported by $\boldsymbol{\vartheta}_{\mathsf{pred}}$": an edge marked with an action a remains in $DTG(v, \Pi^+|\boldsymbol{\vartheta}_{\mathsf{pred}})$ if and only if, for each $w \in \mathcal{V}(\mathsf{pre}(a)) \setminus \{v\}$, $\mathsf{pre}(a)[w] \in \boldsymbol{\vartheta}_{\mathsf{pred}}[w]$. Given that, for each effective value $\vartheta \in \mathcal{D}^{\oplus}(v)$ and each $\boldsymbol{\vartheta}_{\mathsf{pred}} \in \mathcal{D}^{\oplus}(\mathsf{pred}(v))$, $\varphi_v(\vartheta, \boldsymbol{\vartheta}_{\mathsf{pred}}) = \infty$ if ϑ is not reachable from $I[v]$ in $DTG(v, \Pi^+|\boldsymbol{\vartheta}_{\mathsf{pred}})$, or if $G[v]$ is specified yet $G[v] \notin \vartheta$. Otherwise, $\varphi_v(\vartheta, \boldsymbol{\vartheta}_{\mathsf{pred}})$ equals the cost of a cheapest path from $I[v]$ to ϑ in $DTG(v, \Pi^+|\boldsymbol{\vartheta}_{\mathsf{pred}})$. In what follows, the action sequence inducing that cheapest path is denoted by $\pi(\vartheta|\boldsymbol{\vartheta}_{\mathsf{pred}})$.

The properties of COP_{Π^+} constructed as above for the problems in the scope of Theorem 5 are as follows.

(1) The constraint network of COP_{Π^+} corresponds to (the undirected graph induced by) the moral graph of the causal graph CG_{Π}. Since both in-degree and tree-width of CG_{Π} are bounded by a constant, then so is the tree-width ω_{COP} of the constraint network. As we mentioned before, given a graph $\mathcal{G}$, a tree decomposition of a graph $\mathcal{G}$ with width $c \cdot tw(\mathcal{G})$ for a low constant c can be found in time polynomial in the size of $\mathcal{G}$ and exponential only in $tw(\mathcal{G})$. Hence, since $\omega_{\mathsf{COP}} = O(1)$, COP_{Π^+} can be solved in time polynomial in the size of its explicit representation.

(2) Since Π is M-unfoldable, the domain size of each COP variable is $O(poly(||\Pi||))$. Together with (1), that implies that COP_{Π^+} can be solved in time $O(poly(||\Pi||))$.

(3) By the definitions of monotonic relaxation and of domain transition graphs, explicit description of all $DTG(v, \Pi^+)$ for an M-unfoldable FDR task Π is polynomial in $||\Pi||$. Hence, the construction of functional components Φ, and thus of the entire COP_{Π^+}, can be done in time $O(poly(||\Pi||))$.

(4) By the construction of COP_{Π^+} and Lemma 1, for any topological ordering $\{v_1, \dots, v_n\}$ of V, every complete assignment $\boldsymbol{\theta}$ to the COP variables $\mathcal{X}$ such that

$$\sum_{v \in V} \varphi_v(\boldsymbol{\theta}[v], \boldsymbol{\theta}[\mathsf{pred}(v)]) = \alpha \neq \infty$$

induces a relaxed plan

$$\pi(\boldsymbol{\theta}[v_1]|\boldsymbol{\theta}[\mathsf{pred}(v_1)]) \cdot \ldots \cdot \pi(\boldsymbol{\theta}[v_n]|\boldsymbol{\theta}[\mathsf{pred}(v_n)])$$

of cost α for Π, and vice versa. Thus, if Π^+ is solvable, then, given an assignment $\boldsymbol{\theta}^*$ to $\mathcal{X}$ on which the minimization objective of COP_{Π^+} is obtained, we can derive from it (in $O(poly(||\Pi||))$ time) an optimal relaxed plan for Π. Otherwise, if Π^+ is unsolvable, then all the assignments to $\mathcal{X}$ evaluate the objective function to ∞.

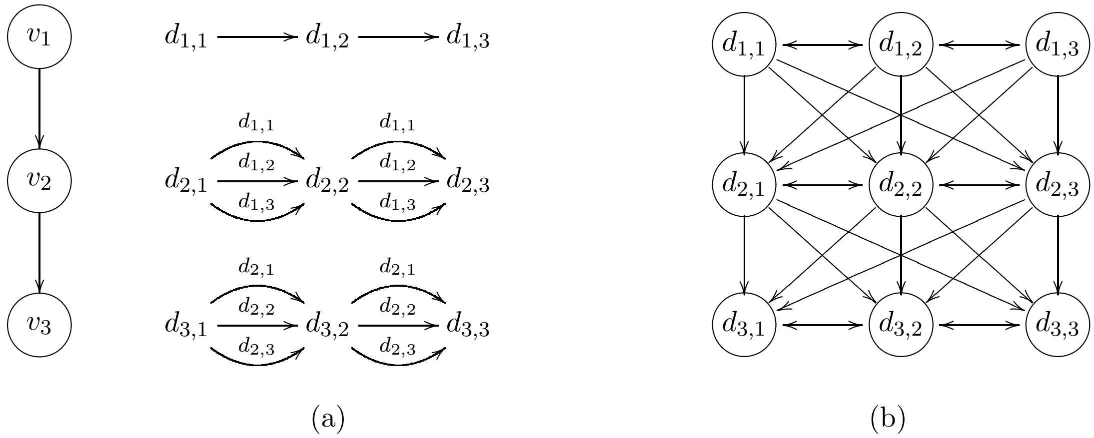


Figure 1: Illustration for the example used in the discussion of Theorem 5.

This finalizes the proof of Theorem 5, and Corollary 4 below generalizes it to digraphs that are "almost DAGs." □

Note that Theorem 5 provides yet another example of exploiting "value grouping" induced by the FDR representation of the planning tasks. Consider a planning task family $\mathbf{\Pi}_{(n)} = \langle V, A, I, G, cost \rangle$ in which $V = \{v_1, \ldots, v_n\}$; for $1 \le i \le n$, $\mathcal{D}(v_i) = \{d_{i,1}, \ldots, d_{i,n}\}$; $I[v_i] = d_{i,1}$; $G[v_i] = d_{i,n}$; and actions

$$A = \bigcup_{\substack{1 \le i,j \le n \\ 1 \le k \le n-1}} \{a_{i,k,j} = \langle \underbrace{\{v_i \leftarrow d_{i,k}, v_{i-1} \leftarrow d_{i-1,j}\}}_{\mathsf{pre}}, \underbrace{\{v_i \leftarrow d_{i,k+1}\}}_{\mathsf{eff}} \rangle \}.$$

Figure 1a illustrates the causal graph and the domain transition graphs for the task $\Pi_{(3)}$. The causal graphs in $\mathbf{\Pi}_{(n)}$ form directed chains, and thus both the tree-width and node in-degree of the causal graphs in $\mathbf{\Pi}_{(n)}$ equal 1. Likewise, the effective domain $\mathcal{D}^{\oplus}(v_i)$ of each v_i in $\Pi^{+}_{(n)}$ is of size n, and thus the tractability of optimal relaxed planning for $\mathbf{\Pi}_{(n)}$ is directly covered by Theorem 5. In contrast, if each variable value $d_{i,j}$ is represented by a separate propositional variable, inducing the causal graph as in Figure 1b, then *both* the tree-width and node in-degree in the family of the induced causal graphs are of the order of n, and in fact, the causal graph is not even acyclic. Therefore, Theorem 5 is no longer directly applicable.

Corollary 4 *Let* $\mathbf{\Pi}$ *be an infinite set of* M*-unfoldable* FDR *tasks. If the size of the strongly connected components, the tree-width, and the node in-degree of the causal graphs in* $\mathbf{\Pi}$ *are all bounded by a constant, then optimal relaxed planning for* $\mathbf{\Pi}$ *is* **polynomial-time**.

Proof: Any FDR task Π with a causal graph whose strongly connected components (SCCs) are of size at most k can be compiled into an equivalent FDR task Π_m with a directed acyclic causal graph by *merging* the variables of each SCC into a single variable (Seipp & Helmert, 2011). This compilation can be done in time polynomial in $||\Pi||$ and exponential only in k. The causal graph CG_{Π_m} is obtained from the causal graph CG_{Π} by contracting all nodes of each SCC. Since node contraction can only decrease the tree-width, we have $tw(CG_{\Pi_m}) \le$

$tw(CG_\Pi)$, and thus $tw(CG_{\Pi_m}) = O(1)$. Likewise, if the maximal node in-degree in CG_Π is c, then the maximal node in-degree in CG_{Π_m} is ck, and thus it is also $O(1)$. Finally, the domain of a variable u in Π_m that is obtained by merging some Π's SCC $\{v^u_1, \dots, v^u_{k'}\}$, $k' \leq k$, corresponds to the cross-product of the domains of these SCC's variables. It is easy to verify from Definition 2 that $\mathcal{D}^\oplus(u) \subseteq \mathcal{D}^\oplus(v^u_1) \times \cdots \times \mathcal{D}^\oplus(v^u_{k'})$, and thus, together with $k = O(1)$, M-unfoldability of all $v^u_1, \dots, v^u_{k'}$ implies $|\mathcal{D}^\oplus(u)| = O(\text{poly}(||\Pi||))$, fitting[3] Definition 3. □

Returning now to the statement of Theorem 5, a few comments on its extensions beyond Corollary 4 are in place. First, note that Theorem 5 as it is does not generalize Theorem 4 for fork-structured FDR tasks because the latter allows for general, and not only M-unfoldable, leaf variables. However, it is easy to see that Theorem 5 can be stratified to allow for such generalization. Since no other variable depends on a leaf v, all we care about in v is achieving $G[v]$. Thus, for any optimal relaxed plan ρ, $\rho_{\downarrow v}$ induces a simple path, and not a general arborescence, in $DTG(v, \Pi)$. Hence, using binary-valued ("$G[v]$ achieved: yes/no") COP variables x_v for the DAG leaf v, and specifying the respective functions φ_v using the procedure in the proof of Theorem 4, the scope of Theorem 5 is extended to generalize Theorem 4.

Second, in the case of DAG-structured causal graphs, Definition 2 of effective domains, on which the notion of M-unfoldability is based, is overly conservative. Instead of deriving the effective domains for the variables in isolation, we can derive them in a topological order of the causal graph, *given* the already derived effective domains of the immediate ancestors. In specific domains, this can substantially extend the scope of M-unfoldability for FDR tasks with DAG causal graphs.

Finally, Theorem 5 requires not only the tree-width, but also the in-degree of the causal graph to be bounded by a constant. As such an extra condition, the latter is sufficient, but not necessary. Below, under the notion of *prevail decomposability*, we list two local properties of state variables that guarantee polynomial-time optimal relaxed planning on arbitrary acyclic causal graphs with a fixed tree-width. It is very likely that other such helpful properties exist, and thus the boundaries of prevail decomposability can be further extended. Nicely, optimal relaxed planning will remain polynomial-time even if different state variables satisfy different such properties, and even if some state variables are not prevail decomposable, but have fixed in-degree.

Definition 5 *Let* $\mathbf{\Pi}$ *be an infinite set of* FDR *tasks, and* Γ *be a property of state variables that, for each task* $\Pi \in \mathbf{\Pi}$*, partitions the state variables of* Π *into those that satisfy* Γ *(referred as* Γ*-variables), and those that do not satisfy* Γ*. We say that* Γ*-variables in* $\mathbf{\Pi}$ *are* **prevail decomposable** *if, for every task* $\Pi \in \mathbf{\Pi}$ *and every* Γ*-variable* v *of* Π*, either*

(i) the set $\mathsf{PR}_v = \{\mathsf{pre}(a)[\mathsf{pred}(v)] \mid a \in A_v\}$ *of preconditions of actions* A_v *on variables other than* v *is of size* $O(\log(||\Pi||))$*, or*

3. Note that it is possible that $\mathcal{D}^\oplus(u) \subset \mathcal{D}^\oplus(v^u_1) \times \cdots \times \mathcal{D}^\oplus(v^u_{k'})$. For instance, if $V = \{x, y\}$, $\mathcal{D}(x) = \mathcal{D}(y) = \{0, 1\}$, $I = \{x \leftarrow 0, y \leftarrow 0\}$, $G = \{x \leftarrow 1, y \leftarrow 1\}$, and $A = \{\langle\{x \leftarrow 0, y \leftarrow 1\}, \{x \leftarrow 1\}\rangle, \langle\{y \leftarrow 0, x \leftarrow 1\}, \{y \leftarrow 1\}\rangle\}$, then $\mathcal{D}^\oplus(x) = \{\{0\}, \{0, 1\}\}$, $\mathcal{D}^\oplus(y) = \{\{0\}, \{0, 1\}\}$, but $\mathcal{D}^\oplus(xy) = \{\{x \leftarrow 0, y \leftarrow 0\}\}$. In fact, this example can be easily extended so that the "merged" variable xy is M-unfoldable, while both x and y are not.

(ii) *the set* ARB_v *of arborescence subgraphs of* $DTG(v, \Pi)$ *rooted in* $I[v]$*, and covering* $G[v]$ *if* $v \in \mathcal{V}(G)$*, is of size* $O(\log(||\Pi||))$*.*

We say that the tasks in $\mathbf{\Pi}$ *are* **prevail decomposable** *if all state variables in* $\mathbf{\Pi}$ *are prevail decomposable.*

Note that prevail decomposability of type (i) is tangential to the notion of M-unfoldability: neither does the former imply the latter, nor the other way around. In contrast, prevail decomposability of type (ii) is a direct stratification of M-unfoldability because the latter considers a compacted version $DTG^{\dagger}(v, \Pi)$ of $DTG(v, \Pi)$, and furthermore, allows for a polynomial (rather than logarithmic) bound on the number of arborescence subgraphs.

Theorem 6 *Let* $\mathbf{\Pi}$ *be an infinite set of* M*-unfoldable, prevail decomposable* FDR *tasks with directed acyclic causal graphs. If the tree-width of the causal graphs in* $\mathbf{\Pi}$ *is bounded by a constant, then optimal relaxed planning for* $\mathbf{\Pi}$ *is* **polynomial-time***.*

Proof: Here as well, our proof of Theorem 6 follows the planning-to-COP compilation methodology. However, the compilation under prevail decomposability must differ from the one in the proof of Theorem 5 since we can no longer rely on fixed in-degree of the causal graphs to derive the fixed tree-width of the constraint networks from the fixed tree-width of the causal graphs. For ease of presentation, we first specify the compilation assuming all the state variables satisfy the specific condition (i) of Definition 5. We then extend the specification to cover the alternative condition (ii) of Definition 5 as well.

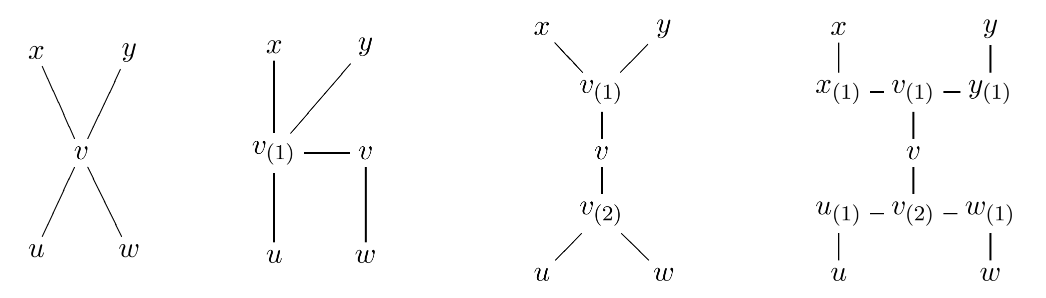
For our construction we need to establish a certain graph-theoretic formalism and a respective notation. Let $\mathcal{G} = (V, E)$ be a graph, and let $N : V \to 2^V$ be the node neighborhood function of $\mathcal{G}$, that is, $N(v) = \{u \mid \{v, u\} \in E\}$. The *splitting* of $v \in V$ with the support $S \subseteq N(v)$ transforms $\mathcal{G}$ by adding new vertex v' and edge $\{v, v'\}$, and, for all $u \in S$, removing edge $\{v, u\}$ and adding edge $\{v', u\}$. Informally, splitting can be seen as a (non-unique) reverse process to edge contraction, and the nodes added to $\mathcal{G}$ by splittings are called *stretch nodes*. For example, Figure 2b depicts the graph obtained from the graph in Figure 2a by splitting the node v with the support of $\{x, y, u\} \subset N(v) = \{x, y, u, w\}$, adding a stretch node $v_{(1)}$.

A graph $\mathcal{G}^*$ is an *expansion* of $\mathcal{G}$ if $\mathcal{G}$ can be transformed to $\mathcal{G}^*$ by a sequence of splittings. For example, Figure 2c depicts the graph obtained from the graph in Figure 2a by first splitting the node v with the support of $\{x, y\} \subset N(v) = \{x, y, u, w\}$, and then splitting v with the support of $\{u, w\} \subset N(v) = \{v_{(1)}, u, w\}$. More specifically, $\mathcal{G}^* = (V^*, E^*)$ is an expansion of $\mathcal{G} = (V, E)$ if and only iff there exist functions $f : V^* \to V$ and $g : E \to E^*$ such that

(a) For $v \in V$, the subgraph of $\mathcal{G}^*$ induced by $f^{-1}(v) = \{v' \in V^* \mid f(v') = v\}$ is a tree, and

(b) For $\{v, u\} \in E$, if $g(\{v, u\}) = \{v', u'\}$ then $f(v') = v$ and $f(u') = u$.

The tree subgraph $T^*(v)$ of $\mathcal{G}^*$ induced by $f^{-1}(v)$ is called the *stretch tree* of v. There is a bijective correspondence between the leaves of $T^*(v)$ and the neighbors $N(v)$ of v via the function g: for each $\{v, u\} \in E$, there is exactly one edge in E^*, $g(\{v, u\})$, that directly connects between $T^*(v)$ and $T^*(u)$. In other words, f induces a partition of V^*, with each part being the stretch tree $T^*(v)$ for some $v \in V$, and g maps the edges of $\mathcal{G}$ to those edges



(a) (b) (c) (d)

Figure 2: Node splitting and graph expansions.

of $\mathcal{G}^*$ that connect between the parts of this partition. Finally, if the node degree in $\mathcal{G}^*$ is bounded by 3, then $\mathcal{G}^*$ is called *sub-cubic*. For example, the expansion in Figure 2c is sub-cubic, while the expansion in Figure 2b is not.

The above terminology is mostly adopted from Markov and Shi (2011). In addition, we call an expansion $\mathcal{G}^*$ of $\mathcal{G}$ *fully separating* if the stretch trees in $\mathcal{G}^*$ are connected only at the stretch nodes, and not at the original nodes, of $\mathcal{G}$. That is, $\mathcal{G}^* = (V^*, E^*)$ is a fully separating expansion of $\mathcal{G} = (V, E)$ if there exists a function $\tau : V \to V^*$ such that, for each $v \in V$, it holds that $\tau(v) \in f^{-1}(v)$ and, for each edge $\{\tau(v), v'\} \in E^*$, $v' \in f^{-1}(v)$. For example, the expansion in Figure 2c is not fully separating, while the expansion in Figure 2d is.[4]

Given the above notion of graph expansion, if we now have a problem to be solved on a graph $\mathcal{G}$, and efficiency of solving this problem depends badly, possibly exponentially, on the node degree in $\mathcal{G}$, then we can try to reformulate this problem over a sub-cubic expansion $\mathcal{G}^*$ of $\mathcal{G}$. However, if the efficiency of the problem in question also depends badly on the graph's tree-width, then the tree-width of $\mathcal{G}^*$ should be as close as possible to that of $\mathcal{G}$. (The tree-width of $\mathcal{G}^*$ cannot be smaller than the tree-width of $\mathcal{G}$ because $\mathcal{G}^*$ has $\mathcal{G}$ as a minor.) While there are numerous efficient schemes for sub-cubic graph expansion, most of them can create expansions of arbitrarily larger tree-width than that of their expandees. Recently, however, Markov and Shi (2011) showed that this negative side-effect can always be eliminated, and sometimes even efficiently.

Theorem 3.1 of Markov and Shi (2011) states their main result: *there is a polynomial-time algorithm that, given a graph $\mathcal{G}$ and its tree decomposition of width w, computes a sub-cubic expansion $\mathcal{G}^*$ of $\mathcal{G}$ with $tw(\mathcal{G}^*) \leq w + 1$.* In particular, this result implies that any graph $\mathcal{G}$ admits a sub-cubic expansion whose tree-width is no more than $tw(\mathcal{G}) + 1$, and that this expansion can be constructed *efficiently* for arbitrary graph families with a fixed

4. Without any effective loss of generality, one can assume that $\tau(v) = v$, that is, the nodes in V are never mapped to stretch nodes, but only to their "mirrors" in V^*. However, here we decided to stick to the explicit use of the τ function to avoid confusion between the nodes V and the identically named nodes in V^*.

tree-width.[5] Moreover, it is straightforward to verify that any expansion can be transformed in linear time into a fully separating expansion, without increasing the tree-width and node degrees. Therefore, Theorem 3.1 of Markov and Shi (2011) holds even if we request fully separating sub-cubic expansions.[6]

Our COP compilation exploits such "tree-width friendly" expansions of causal graphs. Since our focus in Theorem 5 is on digraph families $\mathcal{C}$ with the tree-width in $\mathcal{C}$ being bounded by a constant, by Theorem 3.1 of Markov and Shi (2011), any digraph $\mathcal{G} \in \mathcal{C}$ can be efficiently associated with a fully separating sub-cubic expansion $\mathcal{G}^*$ with tree-width $\leq tw(\mathcal{G}) + 1$. Note, however, that the construction of $\mathcal{G}^*$ ignores the orientation of the arcs in $\mathcal{G}$: while $\mathcal{G}$ is a digraph, $\mathcal{G}^*$ is an *undirected* graph, and its construction is based on a tree decomposition of the *undirected* graph induced by $\mathcal{G}$. Since our COP compilation does depend on the direction of the arcs in the causal graph, we will have to restore in $\mathcal{G}^*$ the relevant bits of this information about $\mathcal{G}$.

But first we give some auxiliary notation.

- Given a fully separating expansion $\mathcal{G}^*$ of (the undirected graph induced by) a digraph $\mathcal{G} = (V, E)$, we consider stretch trees $T^*(v)$ as if rooted in the respective nodes $\tau(v)$, and by $T^*_{v'}(v)$ we denote the subtree of $T^*(v)$ rooted at $v' \in T^*(v)$.

- Recalling that the graphs $\mathcal{G}$ of our interest here are DAGs, and that the leaves of $T^*(v)$ are bijectively associated with the neighbors $N(v)$ of v in $\mathcal{G}$, let $N^{\text{in}}(v), N^{\text{out}}(v) \subseteq N(v)$ be the partition of v's neighbors in $\mathcal{G}$ into immediate ancestors and immediate descendants of v, respectively.

- By $N^{\text{in}}_{v'}(v) \subseteq N^{\text{in}}(v)$ and $N^{\text{out}}_{v'}(v) \subseteq N^{\text{out}}(v)$ we denote the respective neighbors of v that are associated with the leaves of the stretch subtree $T^*_{v'}(v)$. That is, we have $u \in N^{\text{in}}_{v'}(v) \cup N^{\text{out}}_{v'}(v)$ if and only if, for some leaf v'' of $T^*_{v'}(v)$ and some $u'' \in T^*(u)$, $\mathcal{G}^*$ contains edge $\{v'', u''\}$ (i.e., $g(\{v, u\}) = \{v'', u''\}$).

We now proceed with specifying our COP compilation for the FDR tasks as in Theorem 6. Given such a task $\Pi = \langle V, A, I, G, cost \rangle$, let $\mathcal{G}^* = (V^*, E^*)$ be a fully separating, sub-cubic expansion $\mathcal{G}^*$ of the causal graph CG_Π with tree-width $\leq tw(CG_\Pi) + 1$. The respective constraint optimization problem $\mathsf{COP}_{\Pi^+} = (\mathcal{X}, \Phi)$ is specified as follows.

For each $v \in V$, $\mathcal{X}$ contains a variable x_v that is schematically associated with the root $\tau(v)$ of $T^*(v)$, and a variable $x_{v'/v}$ for each stretch-tree node $v' \in T^*(v) \setminus \{\tau(v)\}$. The domain of the variable x_v is

$$\mathscr{D}(x_v) = \begin{cases} \{\vartheta \mid \vartheta \in \mathcal{D}^{\oplus}(v), G[v] \in \vartheta\}, & v \in \mathcal{V}(G) \\ \mathcal{D}^{\oplus}(v), & \text{otherwise} \end{cases}. \tag{7}$$

The domain of each variable $x_{v'/v}$ is

$$\mathscr{D}(x_{v'/v}) = \{0,1\}^{m_v} \times \mathscr{D}(x_v), \tag{8}$$

5. While determining optimal tree decomposition of a graph is NP-hard, it can be done in polynomial time for graph families having fixed tree-width (Bodlaender, 1996).
6. Requiring the expansions to be fully separating is more of a luxury than a need: relying on this property simplifies the compilation scheme described next, but that scheme can also be modified so to not require full separation.

where $m_v = |\mathsf{PR}_v|$. That is, $\mathscr{D}(x_{v'/v})$ is a set of some *pairs* $\langle \delta, \vartheta \rangle$, with $\delta \in \{0,1\}^{m_v}$ and $\vartheta \in \mathcal{D}^{\oplus}(v)$. For each $\delta \in \{0,1\}^{m_v}$, by $DTG(v, \Pi^+|\delta)$ we denote the restriction of $DTG(v, \Pi^+)$ to edges "supported by δ": Assuming an arbitrarily fixed numbering of the elements of $\mathsf{PR}_v = \{\mathsf{pr}_1, \ldots, \mathsf{pr}_{m_v}\}$, an edge marked with an action a such that $\mathsf{pre}(a)[\mathsf{pred}(v)] = \mathsf{pr}_i \in \mathsf{PR}_v$ remains in $DTG(v, \Pi^+|\delta)$ if and only if $\delta[i] = 1$. For $\vartheta \in \mathcal{D}^{\oplus}(v)$, by $c(\vartheta|\delta)$ we denote the cost of a cheapest path from $I[v]$ to ϑ in $DTG(v, \Pi^+|\delta)$; in case of unreachability, $c(\vartheta|\delta) = \infty$.

Similarly to the way each node of the causal graph's expansion $\mathcal{G}^*$ is associated with a COP variable, it is also associated with a non-negative, extended real-valued function. For each state variable $v \in V$:

(I) The stretch tree root $\tau(v)$ is associated with a function φ_v. The scope of φ_v is

$$Q(\varphi_v) = \{x_v\} \cup \{x_{v'/v} \mid v' \in N^*(\tau(v))\},$$

where $N^* : V^* \to 2^{V^*}$ is the node neighborhood function in $\mathcal{G}^*$. Note that $|N^*(\tau(v))| \leq 3$ because $\mathcal{G}^*$ is sub-cubic. For each $\vartheta \in \mathscr{D}(x_v)$ and each assignment $\boldsymbol{\theta} = \{\langle \delta_{v'}, \vartheta_{v'} \rangle\}_{v' \in N^*(\tau(v))}$ to $Q(\varphi_v) \setminus \{x_v\}$,

$$\varphi_v(\vartheta, \boldsymbol{\theta}) = \begin{cases} c(\vartheta|\delta_{\mathrm{H}(\boldsymbol{\theta})}), & \forall v' \in N^*(\tau(v)) : \vartheta_{v'} = \vartheta \\ \infty, & \text{otherwise} \end{cases}, \tag{9}$$

where $\delta_{\mathrm{H}(\boldsymbol{\theta})}$ is the Hadamard, or entrywise, product of all the indicator vectors $\{\delta_{v'}\}_{v' \in N^*(\tau(v))}$ in $\boldsymbol{\theta}$.

(II) Each *leaf* stretch node $v' \in T^*(v)$ is associated with a $0/\infty$ indicator function $\varphi_{v'/v}$, the scope of which is

$$Q(\varphi_{v'/v}) = \{x_{v'/v}, x_{u'/u}\},$$

where u' is the leaf node of $T^*(u)$ such that $g(\{v,u\}) = \{v', u'\}$. *Here the orientation of the arcs within the causal graph CG_Π matters.* Specifically, if $\{v, u\}$ represents a causal graph arc *from u to v* , then $\varphi_{v'/v}$ zeroes on the assignments $(\langle \delta_{v'}, \cdot \rangle, \langle \delta_{u'}, \vartheta_{u'} \rangle)$ such that

- the vector $\delta_{v'}$ enables all and only all the preconditions in PR_v that are, passively or actively, supported by the value $\vartheta_{u'}$ of u, and
- the vector $\delta_{u'}$ enables *all* the preconditions PR_u, since v does not condition the u-changing actions in our DAG-structured planning task Π.

That is, for each $\langle \delta_{v'}, \vartheta_{v'} \rangle \in \mathscr{D}(x_{v'/v})$ and each $\langle \delta_{u'}, \vartheta_{u'} \rangle \in \mathscr{D}(x_{u'/u})$,

$$\varphi_{v'/v}(\langle \delta_{v'}, \vartheta_{v'} \rangle, \langle \delta_{u'}, \vartheta_{u'} \rangle) = \begin{cases} 0, & \delta_{u'} = \mathbf{1} \ \wedge \\ & \forall 1 \leq i \leq m_v : (\delta_{v'}[i] = 0) \leftrightarrow (u \in \mathcal{V}(\mathsf{pr}_i) \wedge \mathsf{pr}_i[u] \notin \vartheta_{u'}) \\ \infty, & \text{otherwise} \end{cases}. \tag{10}$$

Note that the value of $\varphi_{v'/v}$ is independent of the $\vartheta_{v'}$ component of $\langle \delta_{v'}, \vartheta_{v'} \rangle, \langle \delta_{u'}, \vartheta_{u'} \rangle$.

Otherwise, if $\{v, u\}$ represents a causal graph arc[7] *from v to u*, then conversely,

$$\varphi_{v'/v}(\langle \delta_{v'}, \vartheta_{v'} \rangle, \langle \delta_{u'}, \vartheta_{u'} \rangle) = \begin{cases} 0, & \delta_{v'} = \mathbf{1} \;\wedge \\ & \forall 1 \le i \le m_u : (\delta_{u'}[i] = 0) \leftrightarrow (v \in \mathcal{V}(\mathsf{pr}_i) \wedge \mathsf{pr}_i[v] \notin \vartheta_{v'}) \\ \infty, & \text{otherwise} \end{cases} . \tag{11}$$

(III) Each *internal* stretch node $v' \in T^*(v)$ is also associated with a $0/\infty$ indicator function $\varphi_{v'/v}$, but its scope comprises the variable $x_{v'/v}$, together with all the variables $x_{v''/v}$ that correspond to the immediate descendants of v' in $T^*_{v'}(v)$. That is,

$$Q(\varphi_{v'/v}) = \{x_{v'/v}\} \cup \{x_{v''/v} \mid v'' \in N^*(v') \cap T^*_{v'}(v)\}.$$

For each $\langle \delta_{v'}, \vartheta_{v'} \rangle \in \mathscr{D}(x_{v'/v})$ and each assignment $\boldsymbol{\theta} = \{\langle \delta_{v''}, \vartheta_{v''} \rangle\}$ to $Q(\varphi_{v'/v}) \setminus \{x_{v'/v}\}$, $\varphi_{v'/v}$ zeroes on $(\langle \delta_{v'}, \vartheta_{v'} \rangle, \boldsymbol{\theta})$ if and only if the vector $\delta_{v'}$ enables all and only all the preconditions in PR_v that are (passively or actively) supported by *all* the immediate ancestors $u \in N^{\text{in}}_{v'}(v)$ via the values these ancestors "commit to" at the respective stretch-tree root COP variables x_u. That is,

$$\varphi_{v'/v}(\langle \delta_{v'}, \vartheta_{v'} \rangle, \boldsymbol{\theta}) = \begin{cases} 0, & \delta_{v'} = \delta_{\mathrm{H}(\boldsymbol{\theta})} \wedge \forall v'' \in N^*(v') \cap T^*_{v'}(v) : \vartheta_{v''} = \vartheta_{v'} \\ \infty, & \text{otherwise} \end{cases} . \tag{12}$$

In other words, starting with Eqs. 10 and 11, the support provided by $N^{\text{in}}_{v'}(v)$ to v is communicated to v by the indicator vectors $\delta_{v''}$ in $\boldsymbol{\theta}$, and it is aggregated/summarized in Eq. 12 by the Hadamard vector product $\delta_{\mathrm{H}(\boldsymbol{\theta})}$.

Complexity-wise, the properties of the $\mathsf{COP}_{\Pi+}$ constructed as above are as follows.

(1) The constraint network of $\mathsf{COP}_{\Pi+}$ is obtained from the expansion $\mathcal{G}^*$ by replacing each subgraph of $\mathcal{G}^*$ induced by nodes $Q(\varphi_\alpha)$ with a clique over $Q(\varphi_\alpha)$. Let $\mathcal{G}^*_Q$ be a graph with nodes $Q(\varphi_\alpha)$, $x_\alpha \in \mathcal{X}$, and edges $\{Q(\varphi_\alpha), Q(\varphi_\beta)\}$ for (only) all pairs α, β such that $Q(\varphi_\alpha) \cap Q(\varphi_\beta) \neq \emptyset$. By the construction of the COP functions Φ, $\mathcal{G}^*_Q$ is isomorphic to $\mathcal{G}^*$. Likewise, since $|Q(\varphi_\alpha) \cap Q(\varphi_\beta)| \le 1$ for all pairs of functions $\varphi_\alpha, \varphi_\beta$, the tree-width ω_{COP} of the constraint network is $\le tw(\mathcal{G}^*_Q) \cdot \max_\alpha |Q(\varphi_\alpha)|$. Together that implies

$$\omega_{\mathsf{COP}} \le tw(\mathcal{G}^*_Q) \cdot \max_\alpha |Q(\varphi_\alpha)| = tw(\mathcal{G}^*) \cdot \max_\alpha |Q(\varphi_\alpha)| \le 4 \cdot tw(\mathcal{G}^*) \le 4 \cdot (tw(CG_\Pi) + 1) = O(1).$$

Hence, since finding a constant-factor approximation to the graph's tree-width is polynomial in the size of the graph and exponential only in its tree-width, $\mathsf{COP}_{\Pi+}$ can be solved in time polynomial in the size of its representation.

(2) By M-unfoldability of Π and Eq. 7, for $v \in V$, the domain size of each COP variable x_v is $O(poly(||\Pi||))$. The domain of each stretch node variable $x_{v'/v}$ is a cross-product of two sets. The size of the second set in Eq. 8 is $O(poly(||\Pi||))$ because so is the domain size of the respective variable x_v. The size of the first set in Eq. 8 is $2^{|\mathsf{PR}_v|}$ and, by Definition 5, $2^{|\mathsf{PR}_v|} = 2^{O(\log(||\Pi||))} = O(poly(||\Pi||))$. Together with (1), this implies that $\mathsf{COP}_{\Pi+}$ can be solved in time $O(poly(||\Pi||))$.

7. Since Theorem 6 is devoted to directed acyclic causal graphs, we do not address here the case in which CG_Π contains both (v, u) and (u, v).

(3) By the definition of monotonic relaxation and the definition of domain transition graphs, explicit description of all $DTG(v, \Pi^+)$ for an M-unfoldable FDR task Π is polynomial in $||\Pi||$. Hence, the construction of functional components Φ as in Eqs. 9-12, and thus of the entire COP_{Π^+}, can be done in time $O(poly(||\Pi||))$.

We now proceed to prove the correctness of $\mathsf{COP}_{\Pi^+} = (\mathcal{X}, \Phi)$. That is, we will prove that if Π^+ is unsolvable, then all the assignments to $\mathcal{X}$ evaluate the objective function $\Sigma_{\varphi\in\Phi}\varphi(\mathcal{X})$ to ∞, and otherwise, the objective is minimized on and only on the assignments to $\mathcal{X}$ that correspond to optimal plans for Π^+.

First, given an assignment $\boldsymbol{\theta}$ to $\mathcal{X}$ such that $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta}) < \infty$, we show that $\boldsymbol{\theta}$ induces a valid plan $\rho_{\boldsymbol{\theta}}$ for Π^+ of cost $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta})$. By Eqs. 9 and 12, $\Sigma_{\varphi\in\Phi}\varphi(\boldsymbol{\theta}) < \infty$ implies that, for each $v \in V$ and each $x_{v'/v}$ in the stretch tree $T^*(v)$, $\boldsymbol{\theta}[x_{v'/v}] \in \{\langle \cdot, \boldsymbol{\theta}[x_v]\rangle\}$. That is, the relaxation value of v assigned by $\boldsymbol{\theta}$ to x_v is consistently propagated to all the nodes of $T^*(v)$, and in particular, to its leaves.

Let a leaf node $x_{v'/v}$ in $T^*(v)$ connect $T^*(v)$ with $T^*(u)$ for some causal graph neighbor $u \in N(v)$, and let $\boldsymbol{\theta}[x_{v'/v}] = \langle \delta_{v'}, \boldsymbol{\theta}[x_v]\rangle$. If $u \in N^{\text{in}}(v)$, then by Eq. 10, $\varphi_{v'/v}(\boldsymbol{\theta}) \neq \infty$ implies that $\delta_{v'}$ encodes all and only all preconditions in PR_v that are not disabled by the relaxation value $\boldsymbol{\theta}[x_u]$ of u. Otherwise, if $u \in N^{\text{out}}(v)$, then, by the DAG structure of CG_Π, u has nothing to do with preconditions of actions affecting v^+, and by Eq. 11, $\varphi_{v'/v}(\boldsymbol{\theta}) \neq \infty$ implies that $\delta_{v'} = \mathbf{1}$ trivially enables all the preconditions in PR_v.

Given that, for *each* (leaf or internal) stretch node in $T^*(v)$, let $\boldsymbol{\theta}[x_{v'/v}] = \langle \delta_{v'}, \boldsymbol{\theta}[x_v]\rangle$. By the conjunctive structure of the preconditions in FDR and the Hadamard vector product in Eq. 12, $\varphi_{v'/v}(\boldsymbol{\theta}) \neq \infty$ implies that $\delta_{v'}$ encodes all and only all the preconditions in PR_v that are not disabled by the values $\boldsymbol{\theta}[x_u]$ of all $u \in N^{\text{in}}_{v'}(v)$. Finally, by the definition of graph expansion, $\bigcup_{v'\in N^*(\tau(v))} N^{\text{in}}_{v'}(v) = N^{\text{in}}(v)$. Thus, by Eq. 11, $\varphi_v(\boldsymbol{\theta}) \neq \infty$ implies that $\boldsymbol{\theta}[x_v]$ is reachable from $I[v]$ in the properly restricted domain transition graph $DTG(v, \Pi^+|\boldsymbol{\theta}[\mathcal{X}_v])$ where $\mathcal{X}_v = \{x_u \mid u \in \mathsf{pred}(v)\}$, and that $\varphi_v(\boldsymbol{\theta})$ equals the cost of the cheapest such path. The rest follows from the DAG structure of CG_Π and Lemma 1. The proof of the opposite direction is straightforward from the construction of COP_{Π^+} and the "serialization" Lemma 1. □

In our final note we return to the definition of prevail decomposability, and specifically, to its second sufficient condition that, for all state variables $v \in V$, the set ARB_v of arborescence sugraphs of $DTG(v, \Pi)$ rooted in $I[v]$, and covering $G[v]$ if $v \in \mathcal{V}(G)$, is of size $O(\log(||\Pi||))$. While this condition was not addressed in our COP construction so far, switching from the first to the second sufficient condition of prevail decomposability requires only that the semantics of the indicator vectors δ be changed: Instead of encoding the support that $\mathsf{pred}(v)$ provide to individual preconditions of actions in A_v, they should encode the support that $\mathsf{pred}(v)$ provide to the entire arborescences ARB_v in $DTG(v, \Pi)$. Since the condition requires $|\mathsf{ARB}_v| = O(\log(||\Pi||))$, this support can be encoded and reasoned about efficiently. Note that the choice between the two conditions can be made on a variable-by-variable basis, and thus the two conditions are not mutually exclusive but complementary.

$\mathbf{\Pi}$		FDR	MFDR	
causal graph	extra condition	*in P?*	*in P?*	
fixed size		Yes	No	Th. 2
$\omega = O(1)$	$\lvert\mathcal{D}(v)\rvert = O(1)$	No	Yes	Th. 3
$\omega = O(1)$ & DAG	$\lvert\mathcal{D}(v)\rvert = O(1)$	No	Yes	Th. 3
$\omega = O(1)$ & DAG	in-degree $= O(1)$	No	Yes, if M-unfoldable	Th. 5
$\omega = O(1)$ & DAG		No	Yes, if M-unfoldable and prevail decomposable	Th. 6

Table 1: A summary of our main results for optimal MFDR planning, contrasted with the previously established complexity of the corresponding fragments of optimal FDR planning. In the table, $\mathbf{\Pi}$ is a fragment of FDR/MFDR planning, characterized in terms of the causal graph tree-width ω, causal graph in-degree, and upper bound $|\mathcal{D}(v)|$ on the size of the variable domains. M-unfoldability and prevail decomposability are two properties of MFDR tasks that have been introduced and exploited in this work.

6. Summary and Future Work

We took a step towards a fine-grained classification of worst-case time complexity of optimal monotonic planning, with a focus on "what gets harder" and "what gets easier" when switching from optimal planning to optimal relaxed planning, in the context of finite-domain planning task representations. Along the way, we established both negative and positive results on the complexity of some wide fragments of this problem, with the negative results emphasizing the role of the structure of state variable domains, and the positive results emphasizing the role of the causal graph topology. Table 1 lists our main results for optimal monotonic planning, contrasted with the complexity of the corresponding fragments of optimal FDR planning. The key conclusions are as follows.

1. Optimal planning for monotonic relaxations is hard even if restricted to very simple causal graph structures, but the complexity there stems from the size of the state variable domains.

2. Restricted to planning tasks with constant-bounded state variable domains, the problem becomes solvable in time exponential only in the tree-width of the causal graph, while it is known to be very much not so even for non-optimal regular planning.

3. While the tree-width of digraphs is independent of the edge directions, exploiting the directed structure of the causal graph *together* with its tree-width allows the computational tractability to be expanded beyond fixed-size state variable domains.

The latter conclusion opens an interesting venue for further investigation. While we addressed only directed acyclic causal graphs, the scope of tractability can perhaps be expanded by exploiting some existing "directed" notions of graph width (Johnson, Robertson, Seymour, & Thomas, 2001; Hunter & Kreutzer, 2008; Berwanger et al., 2012). This might be especially appealing because the tree-width of most standard planning benchmarks under their natural FDR encodings does not appear to be fixed across the respective families of the tasks.

Another interesting direction would be to examine the results and techniques introduced here in a wider context: that of the recently introduced framework of *red-black relaxations* (Katz et al., 2013b). In red-black (RB) planning, the variables are partitioned into two sets: the "black" set adopts the regular, value switching semantics of FDR, while the "red" set adopts the monotonic, value accumulating semantics of MFDR. In the context of satisficing planning, complexity analysis of RB planning complexity through the lens of causal graph topology has already led to some advances in the practice of heuristic-search planning (Katz et al., 2013a; Katz & Hoffmann, 2013). To take a similar step in optimal planning, admissible heuristics that are based on RB relaxations must be devised. That, in turn, calls for identifying tractable fragments of optimal RB planning. We are cautiously optimistic that some of the results and techniques presented in this paper will be found valuable in the context of RB planning as well. For instance, the positive result in Theorem 4 for fork-structured MFDR tasks can be straightforwardly extended to RB tasks with only root variables taking the monotonic semantics and all the leaves keeping their regular, FDR semantics. Similarly, the positive result in Theorem 5 for DAG-structured MFDR tasks can be straightforwardly extended to RB tasks with black-painted leaf variables. An interesting question in that respect is whether computational tractability of optimal RB planning can be extended to causal graphs in which some internal nodes get to keep their original FDR semantics.

Acknowledgments

The work was partly supported by the Israel Science Foundation (ISF) grant 1045/12, and the EOARD grant FA8655-12-1-2096.

References

Amir, E. (2010). Approximation algorithms for treewidth. *Algorithmica*, *56*(4), 448–479.

Arnborg, S., Cornell, D. G., & Proskurowski, A. (1987). Complexity of finding embeddings in a k-tree. *SIAM Journal of Algebraic Discrete Methods*, *8*, 277–284.

Bäckström, C., & Klein, I. (1991). Planning in polynomial time: The SAS-PUBS class. *Computational Intelligence*, *7*(3), 181–197.

Bäckström, C., & Nebel, B. (1995). Complexity results for SAS$^+$ planning. *Computational Intelligence*, *11*(4), 625–655.

Becker, A., & Geiger, D. (1996). A sufficiently fast algorithm for finding close to optimal junction trees. In *Proceedings of the 12th Conference on Uncertainty in Artificial Intelligence (UAI)*, pp. 81–89.

Berwanger, D., Dawar, A., Hunter, P., Kreutzer, S., & Obdzrálek, J. (2012). The DAG-width of directed graphs. *Journal of Combinatorial Theory, Series B*, *102*(4), 900–923.

Betz, C., & Helmert, M. (2009). Planning with h^+ in theory and practice. In *Proceedings of the 32nd Annual German Conference on Artificial Intelligence (KI)*, pp. 9–16.

Bodlaender, H. L. (1996). A linear-time algorithm for finding tree-decompositions of small treewidth. *SIAM Journal of Computing*, *25*(6), 1305–1317.

Bonet, B., & Geffner, H. (2001). Planning as heuristic search. *Artificial Intelligence*, *129*(1–2), 5–33.

Bonet, B., & Helmert, M. (2010). Strengthening landmark heuristics via hitting sets. In *Proceedings of the 19th European Conference on Artificial Intelligence*, pp. 329–334, Lisbon, Portugal.

Brafman, R. I., & Domshlak, C. (2006). Factored planning: How, when, and when not. In *Proceedings of the 18th National Conference on Artificial Intelligence (AAAI)*, pp. 809–814, Boston, MA.

Brafman, R. I., & Domshlak, C. (2013). On the complexity of planning for agent teams and its implications for single agent planning. *Artificial Intelligence*, *198*, 52–71.

Bylander, T. (1994). The computational complexity of propositional STRIPS planning. *Artificial Intelligence*, *69*(1-2), 165–204.

Cai, D., Hoffmann, J., & Helmert, M. (2013). Enhancing the context-enhanced additive heuristic with precedence constraints. In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 50–57.

Chen, H., & Gimenez, O. (2010). Causal graphs and structurally restricted planning. *Journal of Computer and System Sciences*, *76*(7), 579–592.

Dechter, R. (2003). *Constraint Processing*. Morgan Kaufmann.

Domshlak, C., & Dinitz, Y. (2001). Multi-agent off-line coordination: Structure and complexity. In *Proceedings of Sixth European Conference on Planning (ECP)*, pp. 277–288.

Downey, R. G., & Fellows, M. R. (1999). *Parameterized Complexity*. Springer-Verlag, New York.

Edelkamp, S. (2001). Planning with pattern databases. In *Proceedings of the European Conference on Planning (ECP)*, pp. 13–24.

Fabre, E., Jezequel, L., Haslum, P., & Thiébaux, S. (2010). Cost-optimal factored planning: Promises and pitfalls. In *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 65–72.

Fikes, R. E., & Nilsson, N. (1971). STRIPS: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, *2*, 189–208.

Flum, J., & Grohe, M. (2006). *Parameterized Complexity Theory*. Springer-Verlag.

Fox, M., & Long, D. (2001). Stan4: A hybrid planning strategy based on subproblem abstraction. *AI Magazine*, *22*(3), 81–84.

Fox, M., & Long, D. (2003). PDDL2.1: An extension to PDDL for expressing temporal planning problems. *Journal of Artificial Intelligence Research*, *20*, 61–124.

Gimenez, O., & Jonsson, A. (2009a). The influence of k-dependence on the complexity of planning. In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 138–145.

Gimenez, O., & Jonsson, A. (2009b). Planning over chain causal graphs for variables with domains of size 5 is NP-hard. *Journal of Artificial Intelligence Research*, *34*, 675–706.

Halin, R. (1976). s-functions for graphs. *Journal of Geometry*, *8*, 171–186.

Helmert, M. (2004). A planning heuristic based on causal graph analysis. In *Proceedings of the Fourteenth International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 161–170.

Helmert, M. (2006). The Fast Downward planning system. *Journal of Artificial Intelligence Research*, *26*, 191–246.

Helmert, M. (2009). Concise finite-domain representations for PDDL planning tasks. *Artificial Intelligence*, *173*, 503–535.

Helmert, M., & Domshlak, C. (2009). Landmarks, critical paths and abstractions: What's the difference anyway?. In *Proceedings of the 19th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 162–169.

Helmert, M., & Geffner, H. (2008). Unifying the causal graph and additive heuristics. In *Proceedings of the 18th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 140–147.

Helmert, M., Haslum, P., & Hoffmann, J. (2007). Flexible abstraction heuristics for optimal sequential planning. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 200–207.

Helmert, M., & Mattmüller, R. (2007). Accuracy of admissible heuristic functions in selected planning domains. In *Proceedings of the 23rd AAAI Conference on Artificial Intelligence*, pp. 938–943.

Hoffmann, J. (2005). When 'ignoring delete lists' works: Local search topology in planning benchmarks. *Journal of Artificial Intelligence Research*, *24*, 685–758.

Hoffmann, J. (2011). Analyzing search topology without running any search: On the connection between causal graphs and h^+. *Journal of Artificial Intelligence Research*, *41*, 155–229.

Hoffmann, J., & Nebel, B. (2001). The FF planning system: Fast plan generation through heuristic search. *Journal of Artificial Intelligence Research*, *14*, 253–302.

Hunter, P., & Kreutzer, S. (2008). Digraph measures: Kelly decompositions, games, and orderings. *Theoretical Computer Science*, *399*(3), 206–219.

Johnson, T., Robertson, N., Seymour, P. D., & Thomas, R. (2001). Directed tree-width. *Journal of Combinatorial Theory, Series B*, *82*(1), 138–154.

Jonsson, P., & Bäckström, C. (1998). State-variable planning under structural restrictions: Algorithms and complexity. *Artificial Intelligence*, *100*(1–2), 125–176.

Karp, R. (1972). Reducibility among combinatorial problems. In *Complexity of Computer Computations*, pp. 85–103. Plenum Press, New York.

Katz, M., & Domshlak, C. (2008). New islands of tractability of cost-optimal planning. *Journal of Artificial Intelligence Research*, *32*, 203–288.

Katz, M., & Domshlak, C. (2010). Implicit abstraction heuristics. *Journal of Artificial Intelligence Research*, *39*, 51–126.

Katz, M., & Hoffmann, J. (2013). Red-black relaxed plan heuristics reloaded. In *Proceedings of the 6th Annual Symposium on Combinatorial Search (SOCS)*, pp. 105–113.

Katz, M., Hoffmann, J., & Domshlak, C. (2013a). Red-black relaxed plan heuristics. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 489–495.

Katz, M., Hoffmann, J., & Domshlak, C. (2013b). Who said we need to relax all variables?. In *Proceedings of the 23rd International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 126–134.

Katz, M., & Keyder, E. (2012). Structural patterns beyond forks: Extending the complexity boundaries of classical planning. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence (AAAI)*, pp. 1779–1785.

Keyder, E., & Geffner, H. (2008a). Heuristics for planning with action costs revisited. In *Proceedings of the 18th European Conference on Artificial Intelligence (ECAI)*, pp. 588–592.

Keyder, E., & Geffner, H. (2008b). Heuristics for planning with action costs revisited. In *Proceedings of the 18th European Conference on Artificial Intelligence*, pp. 588–592.

Markov, I. L., & Shi, Y. (2011). Constant-degree graph expansions that preserve treewidth. *Algorithmica*, *59*, 461–470.

McDermott, D. V. (1999). Using regression-match graphs to control search in planning. *Artificial Intelligence*, *109*(1-2), 111–159.

Nissim, R., & Brafman, R. I. (2012). Multi-agent A^* for parallel and distributed systems. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1265–1266.

Nissim, R., Brafman, R. I., & Domshlak, C. (2010). A general, fully distributed multi-agent planning algorithm. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pp. 1323–1330.

Pednault, E. (1989). ADL: Exploring the middle ground between STRIPS and the situation calculus. In *Proceedings of the 1st International Conference on Principles of Knowledge Representation and Reasoning*, pp. 324–331.

Robertson, N., & Seymour, P. D. (1984). Graph minors III: Planar tree-width. *Journal of Combinatorial Theory*, *36*, 49–63.

Robertson, N., & Seymour, P. D. (1991). Graph minors X: Obstructions to tree decomposition. *Journal of Combinatorial Theory, Series B*, *52*(2), 153–190.

Seipp, J., & Helmert, M. (2011). Fluent merging for classical planning problems. In *Proceedings of the ICAPS-2011 Workshop on Knowledge Engineering for Planning and Scheduling (KEPS)*, pp. 47–53.