Tractable Optimal Planning via Causal Graph Analysis

Ph.D. Research Proposal

Michael Katz

Advisor: Dr. Carmel Domshlak

Faculty of Industrial Engineering and Management
Technion - Israel Institute of Technology

תכנון אופטימאלי יעיל דרך אנליזת גרף התלויות הסיבתיות

תיאור תמציתי של הצעת מחקר לקראת תואר דוקטור

מיכאל כץ

מנחה: דר' כרמל דומשלק

הפקולטה להנדסת תעשיה וניהול
הטכניון – מכון טכנולוגי לישראל

Contents

1. Introduction

The field of automated, domain-independent planning seeks to build algorithms enabling a system to *synthesize a course of action* that will achieve certain goals. As such, automated planning constitutes one of the most foundational areas of the Artificial Intelligence (AI) research (Russell & Norvig, 2004). In general, planning algorithms perform reachability analysis in large-scale state models that are implicitly described in a concise manner via some intuitive declarative language (Russell & Norvig, 2004; Ghallab, Nau, & Traverso, 2004). This way, the planning algorithms are not designed exclusively for problems coming from a certain problem domain (e.g., a domain of transportation tasks), but for a wide spectrum of problems describable in a certain formal language for action representation (such as, e.g., the classical Strips language (Fikes & Nilsson, 1971).) And though planning problems have been studied since the early days of artificial intelligence (Allen, Hendler, & Tate, 1990; Chapman, 1987; Dean & Wellman, 1991; Hendler, Tate, & Drummond, 1990; Wilkins, 1984), recent developments have dramatically advanced the field (Geffner, 2002; Ghallab et al., 2004; Weld, 1999).

There are two main issues to be addressed in planning, and in particular, in planning as search. The first issue is the size of the search space, that is, the number of search states examined before a goal state is found. The size of the search space determines the scalability of the search procedure. The second issue is the quality of the discovered goal-achieving action sequence, and ideally one would like to obtain as cost-efficient as possible goal-achieving action sequence. Both these issues can be addressed by controlling the order in which the search states are examined. The basic idea is to specify a *heuristic function* h from states to scalars, estimating the distance from states to their nearest goal states. The search algorithms then take these heuristic estimates as a search guidance. In particular, well-known search algorithms such as A* and IDA* explore the search nodes s in a non-decreasing order of $g(s) + h(s)$, where $g(s)$ is the true cost of reaching s from the initial search state (Pearl, 1984; Korf, 1985; Korf & Pearl, 1987; Korf, 1998, 1999). In particular, if h is *admissible*, that is, it never overestimates the true cost of reaching the nearest goal state, then such search algorithms are guaranteed to provide an optimal plan to the goal.

The difference between various algorithms for planning as heuristic search is mainly in the heuristic functions they define and use. Most typically, an (admissible) heuristic function for domain-independent planning is defined as the (optimal) cost of achieving the goals in an *over-approximating abstraction* of the planning problem in hand (Pearl, 1984; Bonet & Geffner, 2001). Such an abstraction is obtained by relaxing certain constraints that are present in the specification of the real problem, and the desire is to obtain a *tractable* (that is, solvable in polynomial time), and, at the same time, *informative* abstract problem. The main question is thus: What constraints should we relax to obtain such an effective over-approximating abstraction?

Conceptually, one can distinguish between homomorphism and embedding abstractions, and the former is of our focus in this research proposal. An *homomorphism abstraction* systematically contracts several states to create a single abstract state. Most typically, such a state-gluing is obtained by projecting the original problem onto a subset of its parameters, as if ignoring the constraints that fall outside the projection. Homomorphisms has been successfully explored in the scope of domain-independent pattern database (PDB) heuristics (Edelkamp, 2001; Haslum, Bonet, & Geffner, 2005; Haslum, Botea, Helmert, Bonet, & Koenig, 2007; Helmert, Haslum, & Hoffmann, 2007), inspired by the (similarly named) problem-specific heuristics for search problems such as $(k^2 - 1)$-puzzles, Rubik's Cube, etc. (Culberson & Schaeffer, 1998). A core property of the PDB heuristics is that the problem is projected onto a space of small (up to logarithmic) dimensionality so that reachability analysis in that space could be done by exhaustive search. Note that this constraint implies an inherent scalability limitation of the PDB heuristics—as the problems of interest grow, limiting patterns to logarithmic dimensionality will unavoidably make them less and less informative with respect to the original problems.

Since automated planning is known to be NP-hard even for extremely conservative problem formalisms (Bylander, 1994), extending the palette of good heuristic functions is one of the major keys to success in solving planning problems. Enriching the "toolbox" of foundations for effective heuristic estimates is essential because no heuristic can work well in all problem instances, and thus the best strategy is to extend the range of heuristic functions to work well on as large as possible range of problems. While pattern databases and approximated ignoring negative interactions have created an invaluable progress in the field of optimal heuristic-search planning, to our knowledge they remain the only conceptual foundations for the latter algorithms. In fact, this situation has already started puzzling the planning community, and some evidence for that can be found in the recent literature (e.g., see (Haslum et al., 2005; Haslum, 2006).)

In our work we suggest a generalization of the PDB abstractions to what we call *structural patterns*. In itself, the idea of structural patterns is simple, and it corresponds to *projecting the original problem to provably tractable fragments of optimal planning.* At least theoretically, moving to structural patterns alleviates the requirement for the projections to be of a low dimensionality. To materialize the idea of structural-patterns heuristics, we have started investigating the computational tractability of cost-optimal planning, and have already discovered numerous new problem classes for whose such optimization is tractable (Katz & Domshlak, 2007b, 2007a). The results are based on exploiting numerous structural and syntactic characteristics of planning problems such as the structure of their causal graphs. Moreover, we have already shown that the idea of structural-patterns heuristics is not of a theoretical interest only—in (Katz & Domshlak, 2007a) we suggest a concrete structural patterns abstraction based on decomposing the problem in hand along its causal graph, and show that the induced admissible heuristic can provide more informative estimates than its state-of-the-art alternatives.

2. Formalism and Notation

Probably the most foundational class of planning problems is this of *classical planning*, corresponding to state models with deterministic actions and complete information. Formally, such a **state model** is a tuple $\mathcal{S} = \langle S, s_0, S_G, A, f, \mathcal{C} \rangle$ where

- S is a finite set of states, $s_0 \in S$ is the initial state, and $S_G \subseteq S$ is a set of alternative goal states,
- A is a finite set of actions, and for each $s \in S$, $A(s) \subseteq A$ denotes the set of all actions applicable in s,
- $f : S \times A \rightarrow S$ is a transition function, such that if $a \in A(s)$, then $f(s, a)$ specifies the state obtained from applying a in s, otherwise (a convention is that) $f(s, a) = s$,
- $\mathcal{C} : S \times A^* \rightarrow \mathbb{R}^{0+}$ captures the cost of applying an action sequence in a state.

A solution (= plan) for such a state model $\mathcal{S}$ is a sequence of actions $\rho = \langle a_1, \dots, a_m \rangle$ that generates a sequence of states $s_0, \dots, s_m$ such that, for $0 \le i < m$, a_{i+1} is applicable in s_i, $f(s_i, a_{i+1}) = s_{i+1}$, and $s_m \in S_G$. A plan ρ is optimal if $\mathcal{C}(s_0, \rho)$ is minimal over all plans for $\mathcal{S}$.

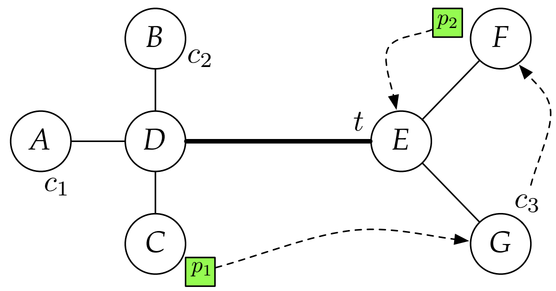


Figure 1: Logistics example adapted from Helmert(2006). Deliver p_1 from C to G, and p_2 from F to E using the cars c_1, c_2, c_3 and truck t, and making sure that c_3 ends up at F. The cars may only use city roads (thin edges), the truck may only use the highway (thick edge).

For describing such a model, we use the SAS$^+$ formalism (Bäckström & Nebel, 1995) that captures problems with multi-valued state variables. The notation here bares a close similarity to this suggested by Helmert ((2004)).

Definition 1 *A* **sas$^+$ problem instance** *is given by a quadruple* $\Pi = \langle V, A, I, G \rangle$*, where:*

- $V = \{v_1, \ldots, v_n\}$ *is a set of* state variables*, each associated with a finite domain* $dom(v_i)$*.*
- *the* initial state I *is a complete assignment, and the* goal G *is a partial assignment to* V*.*
- $A = \{a_1, \ldots, a_N\}$ *is a finite set of* actions*, where each action* a *is a pair* $\langle \mathsf{pre}(a), \mathsf{eff}(a) \rangle$ *of partial assignments to* V *called* preconditions *and* effects*, respectively. Each action* $a \in A$ *is associated with a non-negative real-valued cost* $\mathcal{C}(a)$*.*

A SAS$^+$ problem $\Pi = \langle V, A, I, G \rangle$ defines a state model $\mathcal{S} = \langle S, s_0, S_G, A, f, \mathcal{C} \rangle$ with

- the state space $S = \mathcal{D}(v_1) \times \ldots \times \mathcal{D}(v_n)$ being the space of all possible complete assignments to the state variables V, the initial state $s_0 = I$, and for each $s \in S$, we have $s \in S_G$ iff we have $s[v] = G[v]$ for all $v \in V$ such that $G[v]$ is specified.
- for each $s \in S$, we have $A(s) = \{a \in A \mid \mathsf{pre}(a) \subseteq s\}$, and, for every $s \in S$ and $a \in A(s)$, we have $f(s, a) = s'$ such that

$$s'[v] = \begin{cases} \mathsf{eff}(a)[v] & \mathsf{eff}(a)[v] \text{ is specified} \\ s[v] & \text{otherwise} \end{cases}$$

In this work we focus on *cost-optimal* (also known as *sequentially optimal*) planning in which the task is to find a plan $\rho \in A^*$ for Π minimizing $\sum_{a \in \rho} \mathcal{C}(a)$. The cost of all actions in A is assumed to be non-negative.

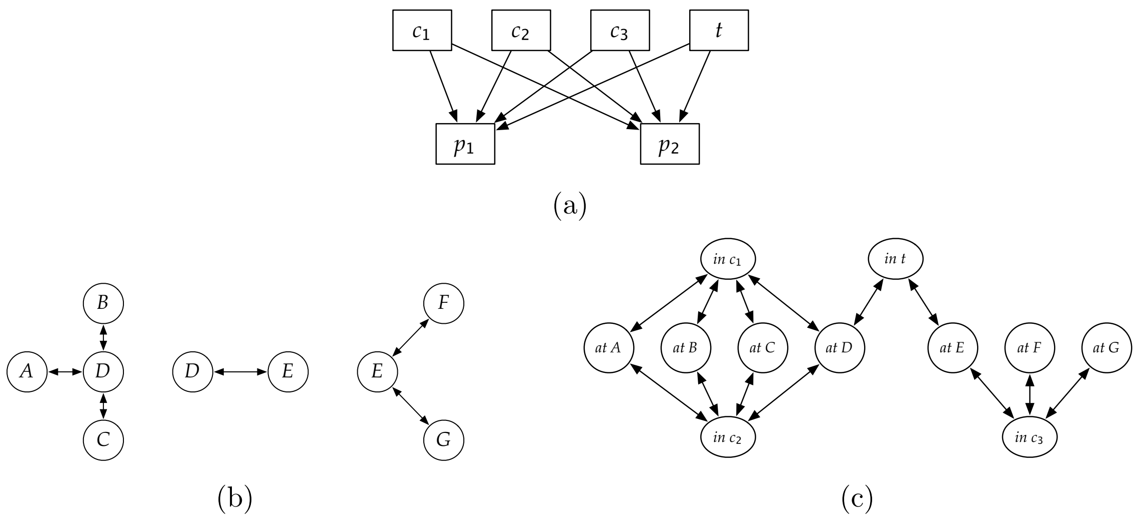


Figure 2: (a) Causal graph; (b) DTGs (labels omitted) of c_1 and c_2 (left), t (centre), and c_3 (right); (c) DTG of p_1 and p_2.

To illustrate various notions used and introduced in our work, we use a slight variation of a Logistics example of Helmert(2006). This example is depicted in Figure 1, and in SAS$^+$ it has

$$
\begin{aligned}
V &= \{p_1, p_2, c_1, c_2, c_3, t\} \\
dom(p_1) &= dom(p_2) = \{A, B, C, D, E, F, G, c_1, c_2, c_3, t\} \\
dom(c_1) &= dom(c_2) = \{A, B, C, D\} \\
dom(c_3) &= \{E, F, G\} \\
dom(t) &= \{D, E\} \\
I &= \{p_1 \mapsfrom C, p_2 \mapsfrom F, t \mapsfrom E, c_1 \mapsfrom A, c_2 \mapsfrom B, c_3 \mapsfrom G\} \\
G &= \{p_1 \mapsfrom G, p_2 \mapsfrom E, c_3 \mapsfrom F\}, \\
A &= \{a_1, \ldots, a_{70}\},
\end{aligned}
$$

where the actions correspond to all possible loads and unloads, plus single-segment movements of the vehicles. For instance, if action a captures loading p_1 into c_1 at C, then $\mathsf{pre}(a) = \{p_1 \mapsfrom C, c_1 \mapsfrom C\}$, and $\mathsf{eff}(a) = \{p_1 \mapsfrom c_1\}$.

Definition 2 *The* **causal graph** $CG(\Pi) = (V, E)$ *of a* SAS$^+$ *problem* $\Pi = \langle V, A, I, G\rangle$ *is a digraph over the nodes* V. *An arc* (v, v') *belongs to* $CG(\Pi)$ *iff* $v \neq v'$ *and there exists an action* $a \in A$ *such that* $\mathsf{eff}(a)[v']$, *and either* $\mathsf{pre}(a)[v]$ *or* $\mathsf{eff}(a)[v]$ *are specified.*

In what follows, for each $v \in V$, by $\mathsf{pred}(v)$ and $\mathsf{succ}(v)$ we refer to the sets of all immediate predecessors and successors of v in $CG(\Pi)$. Likewise, for any partial assignment x to V, and any $V' \subseteq V$, by $x^{[V']}$ we refer to the projection of x onto V'.

Definition 3 *Let* $\Pi = \langle V, A, I, G\rangle$ *be a* SAS$^+$ *problem, and let* $v \in V$. *The* **domain transition graph** $DTG(v, \Pi)$ *of* v *in* Π *is an arc-labeled digraph over the nodes* $dom(v)$ *such that an arc*

(ϑ, ϑ') *belongs to* $DTG(v, \Pi)$ *iff there is an action* $a \in A$ *with* $\mathsf{eff}(a)[v] = \vartheta'$*, and either* $\mathsf{pre}(a)[v]$ *is unspecified or* $\mathsf{pre}(a)[v] = \vartheta$*. In that case, the arc* (ϑ, ϑ') *is labeled with* $\mathsf{pre}(a)^{[V \setminus \{v\}]}$ *and* $\mathcal{C}(a)$*.*

Figure 2 depicts the causal and (labels omitted) domain transition graphs for our running example problem Π. Here we note (and later exploit) that this problem belongs to the class of *unary-effect*, *1-dependent* SAS^+ problems (Katz & Domshlak, 2007b), that is, for all $a \in A$, we have (i) $|\mathsf{eff}(a)| = 1$, and (ii) if $\mathsf{eff}(a)$ is specified for $V' \subseteq V$, then $|\mathsf{pre}(a)^{[V \setminus V']}| \leq 1$.

3. Homomorphism Abstractions in Planning as Heuristic Search

As we already mentioned, the difference between various algorithms for planning as heuristic search is mainly in the heuristic functions they define and use. The immediate question is thus: *What makes an arbitrary scalar-valued function from states to be a good heuristic function?* First, a good heuristic function h has to be *efficient*, that is computable in (as low as possible) polynomial time. The efficiency is important because the value of h is computed at every examined search state. Second, h should be *informative* on a wide class of problems, that is correlating as well as possible with the guidance that would be provided by an oracle. Needless to say that these two desiderata compete with each other, making devising effective heuristics very challenging. Finally, in contrast to general AI search where the heuristic is allowed to be specific to the problem in hand (Korf, 1995), a heuristic for domain-independent planning has to be *automatically extractable* from the declarative specification of the problem. One conceptual tool for achieving such automatically-derivable heuristics for planning is this of homomorphism abstractions.

3.1 Pattern Database Heuristics (PDBs)

An homomorphism abstraction systematically contracts several states to create a single abstract state. Most typically, such state gluing is obtained by projecting the original problem onto a subset of its parameters, as if ignoring the constraints that fall outside the projection. Homomorphisms has been successfully explored in the scope of domain-independent pattern database heuristics (Edelkamp, 2001, 2002), inspired by the (similarly named) problem-specific heuristics for search problems such as $(k^2 - 1)$-puzzles, Rubic's Cube, etc. (Culberson & Schaeffer, 1998).

Very roughly, given a SAS^+ problem $\Pi = \langle V, A, I, G \rangle$, each subset of variables $V' \subseteq V$ defines a *pattern abstraction* $\Pi^{[V']} = \langle V', A^{[V']}, I^{[V']}, G^{[V']} \rangle$ by intersecting the initial state, the goal, and all the actions' pre, prevail and post condition lists with V' (Edelkamp, 2001)[1]. The idea of PDHs is elegantly simple:

(1) Select a (relatively small) set of subsets $V_1, \ldots, V_k$ of V such that, for $1 \leq i \leq k$,

   (a) $\Pi^{[V_i]}$ is an over-approximating abstraction of Π, and

   (b) the size of V_i is sufficiently small to perform reachability analysis in $\Pi^{[V_i]}$ by an (either explicit or symbolic) exhaustive search.

(2) Let $h^{[V_i]}(s)$ be the optimal cost of achieving the abstract goal $G^{[V_i]}$ from the abstract state $s^{[V_i]}$. If the set of abstract problems $\Pi^{[V_1]}, \ldots, \Pi^{[V_k]}$ satisfy certain requirements of disjointness (Felner, Korf, & Hanan, 2004; Edelkamp, 2001), define the PDB heuristic function $h(s) =$

1. An additional way to define pattern abstractions has been recently suggested in (Hoffmann, Sabharwal, & Domshlak, 2006). However, the difference between the two approaches is not critical for our discussion here.

$\sum_{i=1}^{k} h^{[V_i]}(s)$. Otherwise, define $h(s) = \max_{i=1}^{k} h^{[V_i]}(s)$ (Holte, Felner, Newton, Meshulam, & Furcy, 2006a).

The PDB heuristic h as above is provably admissible. The word "databases" comes from the observation that the dimensionality of the abstract problems is anyway required to be suitable for exhaustive reachability analysis, and thus it is worth to pre-calculate, store, and then reuse the values $h^{[V_i]}(s)$ for all pattern abstractions $\Pi^{[V_i]}$ and all states s (that is, their images $s^{[V_i]}$).

3.2 From PDBs to Structural Patterns?

The Achilles heel of the PDB heuristics is that each pattern (that is, each selected subset of variables V_i) is required to be *small* so that reachability analysis in $\Pi^{[V_i]}$ could be done by exhaustive search. In short, computing $h^{[V_i]}(s)$ in polynomial time requires satisfying $|V_i| = O(\log |V|)$. Note that this constraint implies an inherent scalability limitation of the PDB heuristics. As the problems of interest grow, limiting patterns to logarithmic dimensionality will unavoidably make them less and less informative with respect to the original problems, and this unless the domain forces its problem instances to consist of small, loosely-coupled subproblems that can be captured well by individual patterns (Helmert & Mattmüller, 2007).

However, pattern databases are not necessarily the only way to proceed. In principle, given a SAS$^+$ problem $\Pi = \langle V, A, I, G \rangle$, one can select a (relatively small) set of subsets $V_1, \dots, V_m$ of V such that, for $1 \leq i \leq m$,

(a) $\Pi^{[V_i]}$ is an over-approximating abstraction of Π,

(b) *the reachability analysis in* $\Pi^{[V_i]}$ *is tractable (not necessarily due to the size of but) due to the specific structure of* $\Pi^{[V_i]}$.

What is important here is that the second requirement can be satisfied even if the size of each selected pattern V_i is $\Theta(|V|)$. On any event, having specified the abstract problems $\Pi^{[V_1]}, \dots, \Pi^{[V_k]}$ as above, the heuristic estimate h_{sp} (for *structural patterns*) is then formulated similarly to the PDB heuristics: If the set of abstract problems $\Pi^{[V_1]}, \dots, \Pi^{[V_k]}$ satisfy disjointness, then we set $h_{\text{sp}}(s) = \sum_{i=1}^{k} h^{[V_i]}(s)$, where $h^{[V_i]}(s)$ is the optimal cost of achieving the abstract goal $G^{[V_i]}$ from the abstract state $s^{[V_i]}$. Otherwise, we set $h_{\text{sp}}(s) = \max_{i=1}^{k} h^{[V_i]}(s)$.

A priori, this generalization of the PDB idea to structural patterns is appealing as it allows using patterns of unlimited dimensionality. The pitfall, however, is that such structural patterns correspond to tractable fragments of sequentially-optimal planning, and the palette of such known fragments is extremely limited (Bäckström & Nebel, 1995; Bylander, 1994; Jonsson & Bäckström, 1998; Jonsson, 2007). In our work we aim at showing that this palette can still be extended, and such extensions may allow us materializing the idea of structural patterns heuristics.

4. Causal Graph Structural Patterns

We begin with providing a syntactically slight, yet semantically important generalization of the mechanism for constructing *disjoint* decompositions of planning problems along subsets of their state variables.

Definition 4 *Let* $\Pi = \langle V, A, I, G \rangle$ *be a* SAS$^+$ *problem, and let* $\mathcal{V} = \{V_1, \dots, V_m\}$ *be a set of some subsets of* V. *A* **disjoint decomposition of** Π **over** $\mathcal{V}$ *is a set of* SAS$^+$ *problems* $\mathbf{\Pi} = \{\Pi_1, \dots, \Pi_m\}$, *such that*

(1) For each $\Pi_i = \langle V_i, A_i, I_i, G_i \rangle$, we have

(a) $I_i = I^{[V_i]}$, $G_i = G^{[V_i]}$, and

(b) if $a^{[V_i]} \stackrel{\text{def}}{=} \langle \mathsf{pre}(a)^{[V_i]}, \mathsf{eff}(a)^{[V_i]} \rangle$, then

$$A_i = \{a^{[V_i]} \mid a \in A \wedge \mathsf{eff}(a)^{[V_i]} \neq \emptyset\}.$$

(2) Each $a \in A$ satisfies

$$\mathcal{C}(a) \geq \sum_{i=1}^{m} \mathcal{C}_i(a^{[V_i]}). \tag{1}$$

Definition 4 generalizes the idea of "all-or-nothing" action cost partitioning from the literature on PDBs disjoining to arbitrary action cost partitioning. In short, the original cost of each action is distributed this or another way among the "representatives" of that action in the subproblems, with Eq. 1 being the only constraint posed on this cost distribution.

Proposition 1 *For any* SAS$^+$ *problem $\Pi = \langle V, A, I, G \rangle$, any set of V's subsets $\mathcal{V} = \{V_1, \dots, V_m\}$, and any disjoint decomposition of Π over $\mathcal{V}$, we have $h^*(I) \geq \sum_{i=1}^{m} h_i^*(I^{[V_i]})$.*

Proof: If $\rho = \langle a_1 \cdot a_2 \cdot \ldots \cdot a_s \rangle$ is a cost-optimal plan for Π, then $h^*(I) = \mathcal{C}(\rho) = \sum_{i=1}^{s} \mathcal{C}(a_i)$. For $1 \leq i \leq m$, let $\rho^{[V_i]} = \langle a_1^{[V_i]} \cdot a_2^{[V_i]} \cdot \ldots \cdot a_s^{[V_i]} \rangle$. From (1) in Definition 4, we have $\rho^{[V_i]}$ being a (not necessary optimal) plan for Π_i, and thus $h_i^*(I^{[V_i]}) \leq \mathcal{C}_i(\rho^{[V_i]}) = \sum_{j=1}^{s} \mathcal{C}_i(a_j^{[V_i]})$. From (2) in Definition 4 we then have

$$\sum_{i=1}^{m} h_i^*(I^{[V_i]}) \leq \sum_{i=1}^{m} \sum_{j=1}^{s} \mathcal{C}_i(a_j^{[V_i]}) = \sum_{j=1}^{s} \sum_{i=1}^{m} \mathcal{C}_i(a_j^{[V_i]}) \leq \sum_{j=1}^{s} \mathcal{C}(a_j) = h^*(I).$$

■

Although disjoint decomposition over subsets of variables is rather powerful, it is too general for our purposes because it does not account for any structural requirements one may have for the abstract problems. For instance, focusing on the causal graph, when we project the problem onto subsets of its variables, we leave all the causal-graph connections between the variables in each projected problem untouched. However, as here we aim at obtaining abstract problems with causal graphs of *specific structure*, we should somehow project the original problem onto a subgraph (or a set of subgraphs) of the causal graph respecting such structural requirements. This leads us to introducing what we call *causal graph structural patterns.*

Definition 5 *Let $\Pi = \langle V, A, I, G \rangle$ be a* SAS$^+$ *problem, and $\mathcal{G} = (V_{\mathcal{G}}, E_{\mathcal{G}})$ be a subgraph of the causal graph $CG(\Pi)$. A* **causal-graph structural pattern (CGSP)** *$\Pi_{\mathcal{G}} = \langle V_{\mathcal{G}}, A_{\mathcal{G}}, I_{\mathcal{G}}, G_{\mathcal{G}} \rangle$ is a* SAS$^+$ *problem defined as follows.*

1. *$I_{\mathcal{G}} = I^{[V_{\mathcal{G}}]}$, $G_{\mathcal{G}} = G^{[V_{\mathcal{G}}]}$,*

2. *$A_{\mathcal{G}} = \bigcup_{a \in A} A_{\mathcal{G}}(a)$, where each $A_{\mathcal{G}}(a) = \{a^1, \dots, a^{l(a)}\}$, $l(a) \leq |\mathsf{eff}(a)|$, is a set of actions over $V_{\mathcal{G}}$ such that*

(a) *for each* $a^i \in A_{\mathcal{G}}(a)$*, if* $\mathsf{eff}(a^i)[v']$*, and either* $\mathsf{eff}(a^i)[v]$ *or* $\mathsf{pre}(a^i)[v]$ *are specified, then* $(v, v') \in \mathcal{G}$.

(b) *for each* $(v, v') \in \mathcal{G}$*, and each* $a^i \in A_{\mathcal{G}}(a)$*, if* $\mathsf{eff}(a^i)[v']$ *is specified, then either* $\mathsf{eff}(a^i)[v]$ *or* $\mathsf{pre}(a^i)[v]$ *is specified as well.*

(c) *for each* $s \in dom(V_{\mathcal{G}})$*, if* $\mathsf{pre}(a)^{[V_{\mathcal{G}}]} \subseteq s$*, then the action sequence* $\rho = \langle a^1 \cdot a^2 \cdot \ldots \cdot a^{l(a)} \rangle$ *is applicable in* s*, and if applying* ρ *in* s *results in* $s' \in dom(V_{\mathcal{G}})$*, then* $s' \setminus s = \mathsf{eff}(a)$.

(d) $\mathcal{C}(a) \geq \sum_{i=1}^{l(a)} \mathcal{C}_{\mathcal{G}}(a^i)$.

Corollary 1 *For any* SAS$^+$ *problem* $\Pi = \langle V, A, I, G \rangle$*, and any subgraph* $\mathcal{G} = (V_{\mathcal{G}}, E_{\mathcal{G}})$ *of the causal graph* $CG(\Pi)$*, we have* $CG(\Pi_{\mathcal{G}}) = \mathcal{G}$.

Proposition 2 *For any* SAS$^+$ *problem* $\Pi = \langle V, A, I, G \rangle$*, and any subgraph* $\mathcal{G} = (V_{\mathcal{G}}, E_{\mathcal{G}})$ *of the causal graph* $CG(\Pi)$*, at least one CGSP* $\Pi_{\mathcal{G}}$ *can be efficiently constructed from* Π.

Proof: Let $\Pi = \langle V, A, I, G \rangle$ be a SAS$^+$ problem, and $\mathcal{G} = (V_{\mathcal{G}}, E_{\mathcal{G}})$ be a subgraph of the causal graph $CG(\Pi)$. We define a CGSP $\Pi_{\mathcal{G}} = \langle V_{\mathcal{G}}, A_{\mathcal{G}}, I_{\mathcal{G}}, G_{\mathcal{G}} \rangle$ as follows.

1. $I_{\mathcal{G}} = I^{[V_{\mathcal{G}}]}$, $G_{\mathcal{G}} = G^{[V_{\mathcal{G}}]}$,

2. $A_{\mathcal{G}} = \bigcup_{a \in A} A_{\mathcal{G}}(a)$, where if $\{v_1, \ldots, v_k\}$ is the subset of variables $V_{\mathcal{G}}$ affected by a, then each $A_{\mathcal{G}}(a) = \{a^1, \ldots, a^k\}$ is a set of actions over $V_{\mathcal{G}}$ such that

$$\begin{aligned}
\mathsf{eff}(a^i)[v] &= \begin{cases} \mathsf{eff}(a)[v], & v = v_i \\ \text{unspecified}, & \text{otherwise} \end{cases} \\
\mathsf{pre}(a^i)[v] &= \begin{cases} \mathsf{eff}(a)[v], & v = v_j \wedge j < i \wedge (v_j, v_i) \in E_{\mathcal{G}} \\ \mathsf{pre}(a)[v], & v = v_j \wedge j > i \wedge (v_j, v_i) \in E_{\mathcal{G}} \\ \mathsf{pre}(a)[v], & v = v_i \\ \text{unspecified}, & \text{otherwise} \end{cases} \\
\mathcal{C}_{\mathcal{G}}(a^i) &= \frac{\mathcal{C}(a)}{k}
\end{aligned} \tag{2}$$

It is not hard to verify that the requirements of part (2) in Definition 5 are satisfied by $A_{\mathcal{G}}$ as in Eq. 2. ■

Given a SAS$^+$ problem Π and a subgraph $\mathcal{G}$ of $CG(\Pi)$, if the structural pattern $\Pi_{\mathcal{G}}$ can be solved cost-optimally in polynomial time, we can use its solution as an admissible heuristic for Π. Moreover, given a set $\mathbf{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_m\}$ of subgraphs of the causal graph $CG(\Pi)$, these heuristic estimates for structural patterns $\{\Pi_{\mathcal{G}_1}, \ldots, \Pi_{\mathcal{G}_m}\}$ are additive if holds a certain property given by Definition 6.

Definition 6 *Let* $\Pi = \langle V, A, I, G \rangle$ *be a* SAS$^+$ *problem, and* $\mathbf{G} = \{\mathcal{G}_1, \ldots, \mathcal{G}_m\}$ *be a set of subgraphs of the causal graph* $CG(\Pi)$*. A* **disjoint CGSP decomposition of** Π **over** $\mathbf{G}$ *is a set of CGSPs* $\mathbf{\Pi} = \{\Pi_{\mathcal{G}_1}, \ldots, \Pi_{\mathcal{G}_m}\}$ *such that each action* $a \in A$ *satisfies*

$$\mathcal{C}(a) \;\; \geq \;\; \sum_{i=1}^{m} \sum_{a' \in A_{\mathcal{G}_i}(a)} \mathcal{C}_{\mathcal{G}_i}(a'), \tag{3}$$

Proposition 3 *For any* SAS$^+$ *problem* $\Pi = \langle V, A, I, G \rangle$*, any set of* $CG(\Pi)$*'s subgraphs* $\mathbf{G} = \{\mathcal{G}_1, \dots, \mathcal{G}_m\}$*, and any disjoint CGSP decomposition of* Π *over* $\mathbf{G}$*, we have* $h^*(I) \geq \sum_{i=1}^{m} h_i^*(I_{\mathcal{G}_i})$*.*

Proof: If $\rho = \langle a_1 \cdot a_2 \cdot \ldots \cdot a_s \rangle$ is a cost-optimal plan for Π, then $h^*(I) = \mathcal{C}(\rho) = \sum_{j=1}^{s} \mathcal{C}(a_j)$. By Definition 5, for $1 \leq i \leq m$, we have

$$\rho_{\mathcal{G}_i} = a_1^1 \cdot \ldots \cdot a_1^{l(a_1)} \cdot \ldots \cdot a_s^1 \cdot \ldots \cdot a_s^{l(a_s)}$$

being a (not necessary cost-optimal) plan for the CGSP $\Pi_{\mathcal{G}_i}$. Given that, we have

$$h_i^*(I_{\mathcal{G}_i}) \leq \mathcal{C}_{\mathcal{G}_i}(\rho_{\mathcal{G}_i}) = \sum_{j=1}^{s} \sum_{a' \in A_{\mathcal{G}_i}(a_j)} \mathcal{C}_{\mathcal{G}_i}(a').$$

From Eq. 3, for $1 \leq j \leq s$, we have $\sum_{i=1}^{m} \sum_{a' \in A_{\mathcal{G}_i}(a_j)} \mathcal{C}_{\mathcal{G}_i}(a') \leq \mathcal{C}(a_j)$, and thus

$$\begin{aligned} \sum_{i=1}^{m} h_i^*(I_{\mathcal{G}_i}) &\leq \sum_{i=1}^{m} \sum_{j=1}^{s} \sum_{a' \in A_{\mathcal{G}_i}(a_j)} \mathcal{C}_{\mathcal{G}_i}(a') \\ &= \sum_{j=1}^{s} \sum_{i=1}^{m} \sum_{a' \in A_{\mathcal{G}_i}(a_j)} \mathcal{C}_{\mathcal{G}_i}(a') \\ &\leq \sum_{j=1}^{s} \mathcal{C}(a_j) = h^*(I) \end{aligned}$$

■

Relying on Proposition 3, we can now decompose any given problem Π into a set of tractable CGSPs $\mathbf{\Pi} = \{\Pi_{\mathcal{G}_1}, \dots, \Pi_{\mathcal{G}_m}\}$, solve all these CGSPs in polynomial time, and derive an admissible heuristic for Π. Note that (similarly to Definition 4) Definition 6 leaves the decision about the actual partition of the action costs rather open. In our discussion henceforth, we consider the (kind of "least-committing") *uniform* action-cost partitioning in which the action cost is *equally split* among its *non-redundant* projections in $\mathbf{\Pi}$.

4.1 Disjoint Fork-Decompositions

We now introduce certain decomposition of SAS$^+$ planning problems along their causal graphs. In itself, this decomposition does *not* lead to structural patterns abstractions, yet it provides an important building block on our way towards them.

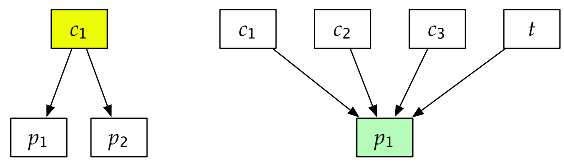
Definition 7 *Let* $\Pi = \langle V, A, I, G \rangle$ *be a* SAS$^+$ *problem. The* **fork-decomposition**

$$\mathbf{\Pi} = \{\Pi_{\mathcal{G}_v^{\mathsf{f}}}, \Pi_{\mathcal{G}_v^{\mathsf{if}}}\}_{v \in V}$$

is a disjoint CGSP decomposition of Π *over subgraphs* $\mathbf{G} = \{\mathcal{G}_v^{\mathsf{f}}, \mathcal{G}_v^{\mathsf{if}}\}_{v \in V}$ *where, for* $v \in V$*,*

$$V_{\mathcal{G}_v^{\mathsf{f}}} = \{v\} \cup \mathsf{succ}(v), \quad E_{\mathcal{G}_v^{\mathsf{f}}} = \bigcup_{u \in \mathsf{succ}(v)} \{(v, u)\}$$

$$V_{\mathcal{G}_v^{\mathsf{if}}} = \{v\} \cup \mathsf{pred}(v), \quad E_{\mathcal{G}_v^{\mathsf{if}}} = \bigcup_{u \in \mathsf{pred}(v)} \{(u, v)\}$$



$CG(\Pi^{\mathsf{f}}_{c_1})$ $\quad$ $CG(\Pi^{\mathsf{if}}_{p_1})$

Figure 3: Causal graphs of a fork and an inverted fork structural patterns of the running example.

Illustrating Definition 7, let us consider the (uniform) fork-decomposition of the problem Π from our running example, assuming all the actions in Π have the same unit cost. After eliminating from $\mathbf{G}$ all the singletons[2], we get $\mathbf{G} = \{\mathcal{G}^{\mathsf{f}}_{c_1}, \mathcal{G}^{\mathsf{f}}_{c_2}, \mathcal{G}^{\mathsf{f}}_{c_3}, \mathcal{G}^{\mathsf{f}}_{t}, \mathcal{G}^{\mathsf{if}}_{p_1}, \mathcal{G}^{\mathsf{if}}_{p_2}\}$. Considering the action sets of the problems in $\mathbf{\Pi}$, each original driving action is present (by its projections) in some three problems in $\mathbf{\Pi}$, while each load/unload action is present in some five such problems. For instance, the projections of the action "drive-c_1-from-A-to-D" are present in $\{\Pi^{\mathsf{f}}_{c_1}, \Pi^{\mathsf{if}}_{p_1}, \Pi^{\mathsf{if}}_{p_2}\}$, and the projections of the action "load-p_1-into-c_1-at-A" are present in $\{\Pi^{\mathsf{f}}_{c_1}, \Pi^{\mathsf{f}}_{c_2}, \Pi^{\mathsf{f}}_{c_3}, \Pi^{\mathsf{f}}_{t}, \Pi^{\mathsf{if}}_{p_1}\}$. Since our fork-decomposition is uniform, the cost of each driving (load/unload) action projection is set to 1/3 (respectively, to 1/5).

From Proposition 3 we have that the sum of costs of solving the problems $\mathbf{\Pi}$, that is,

$$h_{\Lambda\!\!\!\Lambda} = \sum_{v \in V} \left(h^*_{\Pi^{\mathsf{f}}_v} + h^*_{\Pi^{\mathsf{if}}_v} \right), \tag{4}$$

is an admissible estimate of h^*. The question now is how good this estimate is. The optimal cost of solving our problem is 19, and

$$\begin{aligned} h_{\Lambda\!\!\!\Lambda} &= h^*_{\Pi^{\mathsf{f}}_{c_1}} + h^*_{\Pi^{\mathsf{f}}_{c_2}} + h^*_{\Pi^{\mathsf{f}}_{c_3}} + h^*_{\Pi^{\mathsf{f}}_{t}} + h^*_{\Pi^{\mathsf{if}}_{p_1}} + h^*_{\Pi^{\mathsf{f}}_{p_2}} = \\ &= \tfrac{8}{5} + \tfrac{8}{5} + (\tfrac{8}{5} + \tfrac{6}{3}) + (\tfrac{8}{5} + \tfrac{2}{3}) + (\tfrac{6}{5} + \tfrac{9}{3}) + (\tfrac{2}{5} + \tfrac{4}{3}) = \\ &= 15 \end{aligned} \tag{5}$$

Taking as a basis for comparison the seminal $h_{\max}$ and h^2 heuristics (Bonet & Geffner, 2001; Haslum & Geffner, 2000), we have $h_{\max} = 8$ and $h^2 = 13$. Hence, it appears that using the additive CGSP heuristic $h_{\Lambda\!\!\!\Lambda}$ is at least promising.

Unfortunately, despite the seeming simplicity of the problems in $\mathbf{\Pi}$, turns out that fork-decompositions by themselves do not fit the requirements of the structural patterns framework. The causal graphs of $\{\Pi^{\mathsf{f}}_{c_1}, \Pi^{\mathsf{f}}_{c_2}, \Pi^{\mathsf{f}}_{c_3}, \Pi^{\mathsf{f}}_{t}\}$ and $\{\Pi^{\mathsf{if}}_{p_1}, \Pi^{\mathsf{if}}_{p_2}\}$ form directed forks and inverted forks, respectively (see Figure 3), and, in general, the number of variables in each such problem is $\Theta(n)$. Unfortunately for us, Domshlak and Dinitz ((2001)) show that even non-optimal planning for SAS$^+$ problems with fork and inverted fork causal graphs is NP-complete. Moreover, even if the domain-transition graphs of all the projections are strongly connected, optimal planning for forks and inverted forks remain NP-hard (see Helmert ((2003)) and ((2004)) for the respective results). However, in the next section we show that this is not the end of the story on fork-decompositions.

2. If the causal graph $CG(\Pi)$ is connected and $n > 1$, then this elimination is not lossy.

5. Meeting Structural and Domain Abstractions

While hardness of optimal planning for problems with fork and inverted fork causal graphs put a shadow on relevance of fork-decompositions, closer look at the proofs of these hardness results of Domshlak and Dinitz ((2001)) and Helmert ((2003, 2004)) reveals that these proofs in particular rely on root variables having large domains. It turns out that this dependence is not incidental, and Propositions 4 and 5 below present some significant islands of tractability within these structural fragments of SAS^+.

Proposition 4 *Given a* SAS^+ *problem* $\Pi = \langle V, A, I, G \rangle$ *inducing a fork causal graph with a root* $r \in V$*, if*

(i) $|dom(r)| = 2$*, or*

(ii) for all $v \in V$*, we have* $|dom(v)| = O(1)$*,*

then finding a cost-optimal plan for Π *is poly-time.*

Proof: First, if $|dom(r)| = 2$, let $dom(r) = \{0, 1\}$, where $I[r] = 0$. Let $\sigma(r)$ be a 0/1 sequence of length $1 + d$, where $d = \max_{u \in \mathsf{succ}(r)} |dom(u)|$, and, for $1 \leq i \leq |\sigma(r)|$,

$$\sigma(r)[i] = \begin{cases} 0, & \text{i is odd,} \\ 1, & \text{i is even} \end{cases}$$

Finally, let $\trianglerighteq^*[\sigma(r)]$ be the set of all non-empty prefixes of $\sigma(r)$ if $G[r]$ is unspecified, and the set of all non-empty prefixes of $\sigma(r)$ ending with $G[r]$ otherwise.

(1) For each $u \in \mathsf{succ}(r)$, let DTG_u^0 and DTG_u^1 be the subgraphs of $DTG(u, \Pi)$ obtained by removing from the latter all the arcs labeled with 1 and 0, respectively. For each $u \in \mathsf{succ}(r)$, and each $x, y \in dom(u)$, compute the shortest (that is, cost-minimal) paths from x to y in DTG_u^0 and DTG_u^1.

(2) For each $\sigma \in \trianglerighteq^*[\sigma(r)]$, and each $u \in \mathsf{succ}(r)$, build a layered digraph $\mathcal{L}_u(\sigma)$ with $|\sigma| + 1$ layers $L_0, \ldots, L_{|\sigma|}$, where L_0 consists of only $I[u]$, and for $1 \leq i \leq |\sigma|$, L_i consists of all nodes reachable from the nodes L_{i-1} in DTG_u^0 if i is odd, and in DTG_u^1 if i is even. For each $x \in L_{i-1}, y \in L_i$, $\mathcal{L}_u(\sigma)$ contains an arc (x, y) weighted with the cost of the cost-minimal path from x to y in DTG_u^0 if i is odd, and in DTG_u^1 if i is even.

(3) For each $\sigma \in \trianglerighteq^*[\sigma(r)]$, let $|\sigma| = s$. A candidate plan ρ_σ for Π is constructed as follows.

   (a) For each $u \in \mathsf{succ}(r)$, find a cost-minimal path from $I[u]$ to $G[u]$ in $\mathcal{L}_u(\sigma)$. Note that the i-th edge on this path (taking us from $x \in L_{i-1}$ to $y \in L_i$) corresponds to the cost-minimal path from x to y in either DTG_u^0 or DTG_u^1. Let us denote this path from x to y by S_u^i.

   (b) Set $\rho_\sigma = S^1 \cdot a_{\sigma[2]} \cdot S^2 \cdot \ldots \cdot a_{\sigma[s]} \cdot S^s$, where sequence S^i is obtained by an arbitrary merge of the sequences $\{S_u^i\}_{u \in \mathsf{succ}(r)}$, and a_α is the action that changes the value of r to α.

(4) Set and return $\rho = \text{argmin}_{\sigma \in \trianglerighteq^*[\sigma(r)]} \mathcal{C}(\rho_\sigma)$.

It is not hard to verify that the complexity of the above procedure is polynomial in the description size of Π, and that the constructed plan ρ is cost-optimal.

Now we proceed with the second case of the proposition. If $|dom(v)| = O(1)$ for all $v \in V$, let (again) $d = \max_{v \in V} |dom(v)|$. Assuming a unified labeling of the values in each $dom(v)$ as $\{1, 2, \ldots, |dom(v)|\}$, note that (i) there are at most d^d different sequences of domain values of size $\leq d$, (ii) any plan ρ for Π changes the value of each $u \in \mathsf{succ}(r)$ according to one of such sequences, and (iii) each value change of u along ρ is (possibly) prevailed by some value of r. Hence, the length of a sequence of r's values required to support the value changes of $\mathsf{succ}(r)$ along a cost-optimal plan for Π is $\Theta(d^{d+1})$. Since each value of r is reachable from another value of r in $< d$ steps, the total number of value changes of r along a cost-optimal plan for Π is $\Theta(d^{d+2})$.

Given that, we can now generate all possible paths of size $\leq d^{d+2}$ from $I[r]$ to $G[r]$ in $DTG(r, \Pi)$ in time $\Theta(d^{d^{d+2}})$, and, for each such path, and each $u \in \mathsf{succ}(r)$, find a minimal-cost path from $I[u]$ to $G[u]$ in $DTG(u, \Pi)$ that can be supported by the given path for r. This can be done by going over all cycle-free paths in $DTG(u, \Pi)$ from $I[u]$ to $G[u]$, and checking if a series of values of r that supports the corresponding actions is a subsequence of the path for r. There are $\Theta(d^d)$ such paths in $DTG(u, \Pi)$, and the test per can be done in time $\Theta(d^{d+2})$. Thus, for each considered path for r in $DTG(r, \Pi)$, the respective minimal-cost path for u in $DTG(u, \Pi)$ can be found in time $\Theta(d^{2d+2})$ and therefore cost-optimal plan for Π can be found in time[3] $\Theta(d^{2d+2} \cdot d^{d^{d+2}}) = \Theta(d^{d^{d+2}+2d+2}) = O(1)$. ■

Proposition 5 *Given a 1-dependent* SAS$^+$ *problem* $\Pi = \langle V, A, I, G \rangle$ *inducing an inverted fork causal graph with a root* $r \in V$*, if* $|dom(r)| = O(1)$*, then finding a cost-optimal plan for* Π *is poly-time.*

Proof: Let $|dom(r)| = d$. A naive algorithm that finds a cost-optimal plan for Π in time $\Theta(d^{d+1} + |\Pi|^3) = \Theta(|\Pi|^3)$ is as follows.

(1) Create all $\Theta(d^d)$ cycle-free paths from $I[r]$ to $G[r]$ in $DTG(r, \Pi)$.

(2) For each $u \in \mathsf{pred}(r)$, and each $x, y \in dom(u)$, compute the cost-minimal path from x to y in $DTG(u, \Pi)$.

(3) For each path in $DTG(r, \Pi)$ generated in step (1), construct a plan for Π based on that path for r, and the shortest paths computed in (2).

(4) Take minimal cost plan from (3).

The time complexity of this algorithm is $\Theta(|\Pi|^3)$, and it finds a optimal plan, if such exist. The latter can be shown as follows. For each cost-optimal plan ρ, it is easy to verify that $\rho\downarrow_r$ is one of the paths generated in step (1). For each $u \in \mathsf{pred}(r)$, let S_u denote the sequence of values from $dom(u)$ that is required by the prevail conditions of the actions along $\rho\downarrow_r$. If so, for each $u \in \mathsf{pred}(r)$, $\rho\downarrow_u$ corresponds to a path from $I[u]$ to $G[u]$ in $DTG(u, \Pi)$, traversing the values (= nodes) in S_u in the order required by S_u. And a plan for Π generated in (3) consists of minimal such paths for all $u \in \mathsf{pred}(r)$. Therefore, at least one of the plans generated in (3) will be cost-optimal for Π. ■

3. This constant bound is rather sarcastic, but our construction here is not intended to be complexity-optimal either. Finding more realistic bounds for this concrete problem is definitely of interest.

Propositions 4 and 5 allow us to meet between the fork-decompositions and tractable structural patterns at least for 1-dependent planning domains such as Logistics[4]. The basic idea is to further abstract each CGSP in fork-decomposition of Π by abstracting domains of its variables to meet the requirements of the tractable fragments.

Definition 8 *Let $\Pi = \langle V, A, I, G\rangle$ be a* SAS$^+$ *problem, $v \in V$, and let $\Phi = \{\phi_1, \dots, \phi_k\}$ be a set of mappings from $dom(v)$ to some sets $\Gamma_1, \dots, \Gamma_k$. A* **disjoint domain decomposition of Π over** *Φ is a set of* SAS$^+$ *problems $\mathbf{\Pi} = \{\Pi_1, \dots, \Pi_k\}$, such that*

(1) For each $\Pi_i = \langle V_i, A_i, I_i, G_i\rangle$, we have[5]

(a) $I_i = \phi_i(I)$, $G_i = \phi_i(G)$, and

(b) if $\phi_i(a) \stackrel{\text{def}}{=} \langle \phi_i(\mathsf{pre}(a)), \phi_i(\mathsf{eff}(a))\rangle$, then

$$A_i = \{\phi_i(a) \mid a \in A \ \wedge \ \phi_i(\mathsf{eff}(a)) \not\subseteq \phi_i(\mathsf{pre}(a))\}.$$

(2) Each $a \in A$ satisfies

$$\mathcal{C}(a) \geq \sum_{i=1}^{k} \mathcal{C}_i\left(\phi_i(a)\right). \tag{6}$$

Proposition 6 *For any* SAS$^+$ *problem $\Pi = \langle V, A, I, G\rangle$, any $v \in V$, any set of domain abstractions $\Phi = \{\phi_1, \dots, \phi_k\}$, and any disjoint domain decomposition of Π over Φ, we have $h^*(I) \geq \sum_{i=1}^{k} h_i^*(\phi_i(I))$.*

Proof: Let $\rho = \langle a_1 \cdot a_2 \cdot \ldots \cdot a_s\rangle$ be some optimal plan for Π. Then $h^*(I) = \mathcal{C}(\rho) = \sum_{j=1}^{s} \mathcal{C}(a_j)$. For $1 \leq i \leq k$, let $\phi_i(\rho) = \phi_i(a_1) \cdot \phi_i(a_2) \cdot \ldots \cdot \phi_i(a_s)$. From Definition 8 we have $\phi_i(\rho)$ being a (not necessary cost-optimal) plan for Π_i. Therefore, we have $h_i^*(I_i) \leq \mathcal{C}_i(\phi_i(\rho)) = \sum_{j=1}^{s} \mathcal{C}_i(\phi_i(a_j))$. Likewise, from Definition 8, part (2) we have $\sum_{i=1}^{k} h_i^*(I_i) \leq \sum_{i=1}^{k} \sum_{j=1}^{s} \mathcal{C}_i(\phi_i(a_j)) = \sum_{j=1}^{s} \sum_{i=1}^{k} \mathcal{C}_i(\phi_i(a_j)) \leq \sum_{j=1}^{s} \mathcal{C}(a_j) = h^*(I)$. ∎

Targeting tractability of the causal graph structural patterns, we connect between fork-decompositions and domain decompositions as in Definition 8. Given a fork-decomposition $\mathbf{\Pi} = \{\Pi_v^{\mathsf{f}}, \Pi_v^{\mathsf{if}}\}_{v \in V}$ of Π,

- For each $\Pi_v^{\mathsf{f}} \in \mathbf{\Pi}$,
  - (a) Associate the root r of $CG(\Pi_v^{\mathsf{f}})$ with mappings $\Phi_v = \{\phi_{v,1}, \dots, \phi_{v,k_v}\}$, $k_v = O(poly(|\Pi|))$, and all $\phi_{v,i} : dom(r) \to \{0, 1\}$.
  - (b) Disjointly decompose Π_v^{f} into $\mathbf{\Pi}_v^{\mathsf{f}} = \{\Pi_{v,i}^{\mathsf{f}}\}_{i=1}^{k_v}$ over Φ_v.
- For each $\Pi_v^{\mathsf{if}} \in \mathbf{\Pi}$,
  - (a) Associate the root r of $CG(\Pi_v^{\mathsf{if}})$ with mappings $\Phi'_v = \{\phi'_{v,1}, \dots, \phi'_{v,k'_v}\}$, $k'_v = O(poly(|\Pi|))$, all $\phi'_{v,i} : dom(r) \to \{0, 1, \dots, b_{v,i}\}$, $b_{v,i} = O(1)$.

4. If the problem of interest falls outside this problem class, then it should (and could) be first abstracted to a problem within that class. The palette of concrete choices for such an abstraction is rather wide, and currently we investigate their relative pros and cons.
5. For a partial assignment S on V, $\phi_i(S)$ denotes the abstracted partial assignment obtained from S by replacing $S[v]$ (if any) with $\phi_i(S[v])$.

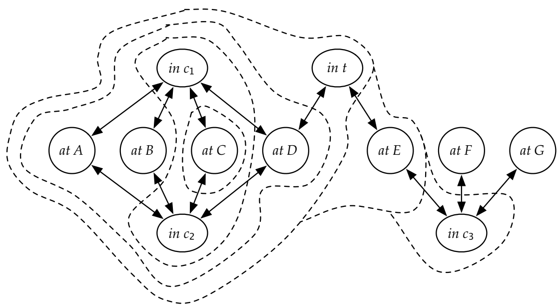


Figure 4: Binary-range domain abstractions for Π_{p_1}. The values within and outside each dashed curve are mapped to 0 and 1, respectively.

(b) Disjointly decompose Π^{if}_v into $\mathbf{\Pi}^{\mathsf{if}}_v = \{\Pi^{\mathsf{if}}_{v,i}\}_{i=1}^{k'_v}$ over Φ'_v.

For 1-dependent problems Π, from Proposition 3 and 6 we then have

$$h_{⩕} = \sum_{v \in V} \left(\sum_{i=1}^{k_v} h^*_{\Pi^{\mathsf{f}}_{v,i}} + \sum_{i=1}^{k'_v} h^*_{\Pi^{\mathsf{if}}_{v,i}} \right), \tag{7}$$

being an admissible estimate of h^* for Π, and from Propositions 4-5 we have that $h_{⩕}$ is also computable in polynomial time. The question is, however, how further abstracting our fork-projections affects the informativeness of the heuristic estimate. As we show later, the answer is somewhat surprising.

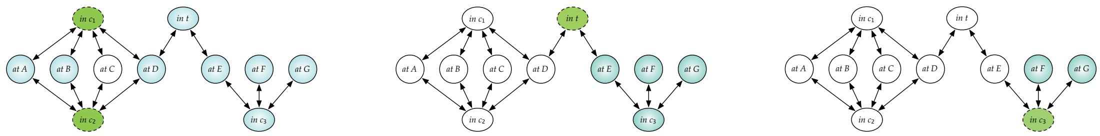
Let us again use our running example to illustrate the mixture of structural and domain projections as outlined above. To begin with an extreme setting of domain abstractions, first, let the domain abstractions for roots of both forks and inverted forks be to binary domains. Among multiple options for choosing the mapping sets $\{\Phi_v\}$ and $\{\Phi'_v\}$, here we use a simple choice of distinguishing between different values of each variable v on the basis of their distance from $I[v]$ in $DTG(v, \Pi)$. Specifically, for each $v \in V$, we set $\Phi_v = \Phi'_v$, and, for each value $\vartheta \in dom(v)$,

$$\phi_{v,i}(\vartheta) = \phi'_{v,i}(\vartheta) = \begin{cases} 0, & d(I[v], \vartheta) < i \\ 1, & \text{otherwise} \end{cases}$$

For instance, the problem $\Pi^{\mathsf{f}}_{c_1}$ is decomposed (see Figure 2b) into two problems, $\Pi^{\mathsf{f}}_{c_1,1}$ and $\Pi^{\mathsf{f}}_{c_1,2}$, with the 0/1 abstract domain of c_1 corresponding to the partitions $\{A\}/\{B, C, D\}$ and $\{A, D\}/\{B, C\}$ of $dom(c_1)$, respectively. The (interesting for certain reasons below) problem $\Pi^{\mathsf{if}}_{p_1}$ is decomposed (see Figure 2c) into six problems $\Pi^{\mathsf{f}}_{p_1,1}, \ldots, \Pi^{\mathsf{f}}_{p_1,6}$ along the abstractions of $dom(p_1)$ depicted in Figure 4.

Now, given the decomposition of Π over forks and $\{\Phi_v, \Phi'_v\}_{v \in V}$ as above, consider the problem $\Pi_{p_1,1}$, obtained from projecting Π onto the inverted fork of p_1 and then abstracting $dom(p_1)$ using

$$\phi_{p_1,1}(\vartheta) = \begin{cases} 0, & \vartheta \in \{C\} \\ 1, & \vartheta \in \{A, B, D, E, F, G, c_1, c_2, c_3, t\} \end{cases}$$



$dom(p_1)$ in $\Pi^{\mathsf{f}}_{p_1,1}$ $dom(p_1)$ in $\Pi^{\mathsf{f}}_{p_1,2}$ $dom(p_1)$ in $\Pi^{\mathsf{f}}_{p_1,3}$

Figure 5: Ternary-range domain abstractions for Π_{p_1}; values that mapped to the same abstract value are shown as nodes with the same color and borderline.

It is not hard to verify that, from the original actions affecting p_1, in $\Pi_{p_1,1}$ we are left only with actions conditioned by c_1 and c_2. If so, then no information is lost[6] if we

1. remove from $\Pi_{p_1,1}$ both the variables c_3 and t, and the actions changing (only) these variables, and

2. redistribute the (fractioned) cost of the removed actions between all other representatives of their originals in Π.

The latter revision of the action cost partitioning is obtained directly by replacing the cost-partitioning steps corresponding to Eqs. 3 and 6 by a single, joint action cost partitioning applied over the final projections $\bigcup_{v\in V}\left(\mathbf{\Pi}^{\mathsf{f}}_v \cup \mathbf{\Pi}^{\mathsf{if}}_v\right)$ and satisfying

$$\mathcal{C}(a) \geq \sum_{v\in V}\left(\sum_{i=1}^{k_v}\sum_{a'\in A_{\mathcal{G}^{\mathsf{f}}_v}(a)} \mathcal{C}^{\mathsf{f}}_{v,i}(\phi_{v,i}(a')) \quad + \sum_{i=1}^{k'_v}\sum_{a'\in A_{\mathcal{G}^{\mathsf{if}}_v}(a)} \mathcal{C}^{\mathsf{if}}_{v,i}(\phi'_{v,i}(a'))\right) \tag{8}$$

Overall, computing $h_{\wedge\!\!\wedge}$ as in Eq. 7 under these "all binary range domain abstractions" provides us with $h_{\wedge\!\!\wedge} = 12\frac{7}{15}$, and knowing that the original costs are all integers we can safely adjust it to $h_{\wedge\!\!\wedge} = 13$. Hence, even under most severe domain abstractions as above, $h_{\wedge\!\!\wedge}$ on our example problem does not fall from h^2.

Let us now slightly relax our domain abstractions for the roots of the inverted forks to be to the ternary range $\{0, 1, 2\}$. While mappings $\{\Phi_v\}$ stay as before, $\{\Phi'_v\}$ is set to

$$\forall\vartheta \in dom(v): \quad \phi'_{v,i} = \begin{cases} 0, d(I[v],\vartheta) < 2i-1 \\ 1, d(I[v],\vartheta) = 2i-1 \\ 2, d(I[v],\vartheta) > 2i-1 \end{cases}$$

For instance, the problem $\Pi^{\mathsf{if}}_{p_1}$ is decomposed now into three problems $\Pi^{\mathsf{f}}_{p_1,1}, \ldots, \Pi^{\mathsf{f}}_{p_1,3}$ along the abstractions of $dom(p_1)$ depicted in Figure 5.

Applying now the same computation of $h_{\wedge\!\!\wedge}$ as in Eq. 7 over our new set of domain abstractions gives $h_{\wedge\!\!\wedge} = 15\frac{1}{2}$, which, again, can be safely adjusted to $h_{\wedge\!\!\wedge} = 16$. Note that this value is *higher* than $h_{\wedge\!\!\wedge} = 15$ obtained using the fork-decomposition alone (as in Eq. 4). At first view, this outcome may seem counterintuitive as the domain abstractions are applied *over* the fork-decomposition. The

6. One of the reasons why no information is lost is the fact that we keep either fork or inverted fork for each variable of Π. In any event, here we omit further formal justifications of this optimization step.

explanation, however, is that (as shown above) the domain abstractions for the roots of inverted forks may create independence between the roots and their preconditioning variables. And exploiting such domain-abstraction specific independence relations leads to more targeted action cost partitioning as in Eq. 8.

6. Research Objectives

Our main objective is to extend and finalize the conceptual framework of structural-patterns admissible heuristics for domain-independent planning, to characterize and study possible instantiations of this framework, their effectiveness and computational efficiency, and to extend the developed heuristic estimates to richer formalisms of domain-independent planning.

Considering the essential ingredients of the structural-patterns framework, currently our primary goal is to *extend the pool of tractable subclasses of optimal classical planning.* Revealing the complexity hierarchy of subclasses of optimal classical planning is still very much an open problem, and our results so far show that progress in this direction is possible. In particular, currently we investigate various special topologies of the causal graphs, along with (practically interesting) local and global restrictions on the problem actions. Specifically, we have started investigating a few special cases of the causal graph structure, namely:

(1) Directed polytrees (poly-forests),

(2) Directed-path singly-connected DAGs, and

(3) Directed-path δ-connected DAGs.

Optimal planning for problems inducing such forms of causal graphs is considered both in general settings, as well as under additional limiting properties such as

(i) actions restricted to only unary effects,

(ii) (prevail) $O(1)$-dependence of the actions,

(iii) $O(1)$-bounded in-degree and/or out-degree of the causal graph, and

(iv) $O(1)$-bounded domains of the state variables.

For unary-effect problems with binary-valued variables, we have recently shown (Katz & Domshlak, 2007b) that optimal planning is tractable for the problem fragments characterized by

(1) Directed polytree with $O(1)$-bounded in-degree,

(2) Directed polytree when 1-dependence of the actions.

In addition, for some minimal extensions of these fragments we have shown that optimal planning is NP-hard. The latter results give us a better understanding of the boundaries of the optimal planning tractability. (The extended version of (Katz & Domshlak, 2007b) is in Appendix.)

The basic principles of the structural patterns framework motivate further research in numerous directions, and in particular, in

- discovering new islands of tractability of optimal planning, and

- translating and/or abstracting the general planning problems into such islands.

In our research we aim at pursuing both these directions by "mining" the tractable fragments of optimal planning, and by performing formal and empirical analysis of alternative schemes for abstracting general planning problems to meet the specification of such islands of tractability. In addition, we plan to start investigating numerous additional issues in using tractable subclasses of optimal planning in homomorphism abstractions for planning as heuristic search. In particular, we plan to devote our efforts to the following research questions.

1. *Optimization of structural patterns selection.* Having established a set of structural classes of planning problems for which optimal planning is tractable, the next step is to formalize the criteria for selecting concrete structural patterns for a given planning problem. The pitfall here is that the number of alternative structural patterns (e.g., the number of different sub-forests of a given causal graph) can be exponential in the size of the problem description. It is also apparent that some choices of structural patterns will be more informative than the other. First, we plan to provide a concrete formal model for optimizing the outcome of structural patterns selection. Second, we will aim at suggesting some tractable approximations for this optimization problem (as the latter is naturally expected to be NP-hard in itself.)

2. *Optimization of variable domains abstraction.* As we show in (Katz & Domshlak, 2007a), selecting a set of structural patterns based on the causal graph decomposition alone might be insufficient (or, at least, informativeness-wise sub-optimal.) In particular, while selecting sub-graphs of the problem's causal graph to form our structural patterns, we might need to further abstract the domains of the multi-valued variables underlying the nodes of the causal graph. For instance, such a domain abstraction will be essential if the problem is described over general multi-valued variables, while the structural patterns are required to be defined over $O(1)$-valued variables. For good and for bad, here as well we have a substantial degree of freedom, and thus ideally we should provide a concrete formal model for optimizing the process of variable domain abstraction. We believe that a substantial progress in this direction can be achieved by bridging between the relevant principles of structural patterns and PDHs, as well as by exploring the structure and interplay between the variables' domain-transition graphs.

3. *Structural patterns disjoining via action cost partitioning.* Similarly to PDHs, the issue of patterns disjoining in the context of structural patterns is appealing because it allows for improving the informativeness of structural patterns heuristics by summing up the local estimates provided by the patterns. In Section 4 we have shown that patterns disjoining can be done by an arbitrary distribution of the action costs between the patterns. It is clear, however, that various such action-cost distributions will lead to heuristic estimates of different quality. Hence, one of our goals is to provide both conceptual and algorithmic foundations for an optimized partitioning of the action costs between the patterns that share these actions.

4. *Structural Patterns beyond CGSPs.* In some cases, projecting the problem onto a subgraph of its causal graph is in itself insufficient to meet the requirements of tractability, and thus some additional transformations of the problem are possibly needed. For instance, if the problem does not exhibit $O(1)$-dependence, but the considered patterns of tractability are based on this property, then either the problem or CGSPs induced by this problem should be reformulated

or further abstracted to satisfy this property. We are interested in studying and formally classifying these (tangential to causal graph structural patterns) problem transformations, as well as their interplay with the core idea of CGSPs.

5. *Approximation-oriented structural patterns.* While heuristic-oriented abstractions for planning are typically based on tractable subclasses of planning problems, the only purpose of the abstract problem is to provide admissible and informative estimates of the distances to the goal in the problem of interest. Thus, in principle, nothing prevents from distance queries in such abstractions (e.g., in structural patterns) to be intractable, and yet both efficiently and informatively *approximatable*. It is clearly of our interest to look not only for islands of tractability of optimal planning, but also, e.g., for islands of "admitting FPTAS approximation schemes".

6. *Extensions to richer formalisms.* One of the challenging directions we would like to pursue at the later stages of our research is this of extending structural-pattern heuristics to some richer planning formalisms. Extending the classical planning, such formalisms allow problem specification to use numerical variables, temporal actions, delayed effects, etc., and for now it is not clear to what degree the homomorphism abstractions can be useful in dealing with such rich formalisms.

7. *Empirical evaluation.* The last but not least, we plan to implement a CGSP-based scheme for planning as heuristic search, and empirically evaluate it on problem domains from the recent planning competitions, as well as on other domains considered in the literature.

References

Allen, J., Hendler, J., & Tate, A. (Eds.). (1990). *Readings in Planning.* Morgan-Kaufmann.

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

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

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

Chapman, D. (1987). Planning for conjunctive goals. *Artificial Intelligence*, *32*(3), 333–377.

Culberson, J., & Schaeffer, J. (1998). Pattern databases. *Computational Intelligence*, *14*(4), 318–334.

Dean, T., & Wellman, M. (1991). *Planning and Control.* Morgan-Kaufmann.

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

Edelkamp, S. (2001). Planning with pattern databases. In *Proceedings of the European Conference on Planning (ECP)*, pp. 13–34.

Edelkamp, S. (2002). Symbolic pattern databases in heuristic search planning. In *Proceedings of the International Conference on AI Planning and Scheduling (AIPS)*, pp. 274–293.

Felner, A., Korf, R. E., & Hanan, S. (2004). Additive pattern database heuristics. *Journal of Artificial Intelligence Research*, *22*, 279–318.

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

Geffner, H. (2002). Perspectives on artificial intelligence planning. In *Proceedings of the Eighteenth National Conference on Artificial intelligence (AAAI)*, pp. 1013–1023.

Ghallab, M., Nau, D., & Traverso, P. (2004). *Automated Planning*. Morgan Kaufmann.

Haslum, P. (2006). Improving heuristics through relaxed search: An analysis of TP4 and HSP*a in the 2004 planning competition. *Journal of Artificial Intelligence Research*, *25*, 233–267.

Haslum, P., Bonet, B., & Geffner, H. (2005). New admissible heuristics for domain-independent planning. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI)*, pp. 1163–1168.

Haslum, P., Botea, A., Helmert, M., Bonet, B., & Koenig, S. (2007). Domain-independent construction of pattern database heuristics for cost-optimal planning. In *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI)*, pp. 1007–1012.

Haslum, P., & Geffner, H. (2000). Admissible heuristics for optimal planning. In *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems (ICAPS)*, pp. 140–149.

Helmert, M. (2003). Complexity results for standard benchmark domains in planning. *Artificial Intelligence*, *146*(2), 219–262.

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., 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)*.

Helmert, M., & Mattmüller, R. (2007). Accuracy of admissible heuristic functions in selected planning domains. In *ICAPS-07 Workshop on Heuristics for Domain-independent Planning: Progress, Ideas, Limitations, Challenges*.

Hendler, J., Tate, A., & Drummond, M. (1990). AI planning: Systems and techniques. *AI Magazine*.

Hoffmann, J., Sabharwal, A., & Domshlak, C. (2006). Friends or foes? an AI planning perspective on abstraction and search. In *Proceedings of the 16th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 294–303.

Holte, R. C., Felner, A., Newton, J., Meshulam, R., & Furcy, D. (2006a). Maximizing over multiple pattern databases speeds up heuristic search. *Artificial Intelligence*. to appear (extends (Holte, Newton, Felner, Meshulam, & Furcy, 2006b)).

Holte, R. C., Newton, J., Felner, A., Meshulam, R., & Furcy, D. (2006b). Multiple pattern databases. In *Proceedings of the 14th International Conference on Automated Planning and Scheduling (ICAPS)*, pp. 122–131.

Jonsson, A. (2007). The role of macros in tractable planning over causal graphs. In *IJCAI*, pp. 1936–1941.

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

Katz, M., & Domshlak, C. (2007a). Structural patterns heuristics. In *ICAPS-07 Workshop on Heuristics for Domain-independent Planning: Progress, Ideas, Limitations, Challenges*.

Katz, M., & Domshlak, C. (2007b). Structural patterns of tractable sequentially-optimal planning. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS)*.

Korf, R. E. (1985). Depth-first iterative-deepening: An optimal admissible tree search. *Artificial Intelligence*, *27*(1), 97–109.

Korf, R. E. (1995). Heuristic evaluation functions in artificial intelligence search algorithms. *Minds and Machines*, *5*(4), 489–498.

Korf, R. E. (1998). Artificial intelligence search algorithms. In Atallah, M. J. (Ed.), *CRC Handbook of Algorithms and Theory of Computation*, No. 36, pp. 1–20. CRC Press.

Korf, R. E. (1999). Heuristic search. In Wilson, R. (Ed.), *Encyclopedia of Cognitive Science*, pp. 372–373. MIT Press.

Korf, R. E., & Pearl, J. (1987). Search techniques. In *Annual Review of Computer Science*, Vol. 2, pp. 451–467. Annual Reviews Inc.

Pearl, J. (1984). *Heuristics - Intelligent Search Strategies for Computer Problem Solving*. Addison-Wesley.

Russell, S., & Norvig, P. (2004). *Artificial Intelligence: A Modern Approach* (2 edition). Pearson.

Weld, D. S. (1999). Recent advances in AI planning. *AI Magazine*, *20*(2), 93–123.

Wilkins, D. E. (1984). Domain-independent planning: Representation and plan generation. *Artificial Intelligence*, *22*, 269–301.

New Islands of Tractability of Optimal Classical Planning

Michael Katz, DUGI@TX.TECHNION.AC.IL,
Carmel Domshlak, DCARMEL@IE.TECHNION.AC.IL
Technion - Israel Institute of Technology, Haifa, Israel

Abstract

We study the complexity of cost-optimal classical planning over propositional state variables and unary-effect actions. We discover novel problem fragments for whose such optimization is tractable, and identify certain conditions that differentiate between tractable and intractable problems. The results are based on exploiting structural and syntactic characteristics of planning problems, as well as a constructive proof technique that connects between certain tools from planning and tractable constraint optimization.

Contents

1. Précis

AI problem solving is inherently facing a computational paradox. On the one hand, most general tasks of AI reasoning are known to be very hard, and this to a degree that membership in NP by itself is sometimes perceived as "good news". On the other hand, if the intelligence is somehow modeled by a computation, and the computation is delegated to the computers, then artificial intelligence has to escape the traps of intractability as much as possible. Planning is one of such reasoning tasks, corresponding to finding a sequence of state-transforming actions that achieve a goal from a given initial state. It is well known that planning is intractable in general (Chapman, 1987), and that even the "simple" classical planning with propositional state variables is PSPACE-complete (Bylander, 1994).

While there were ups and downs in the interest of the planning community in the formal complexity analysis of planning, it is of a growing understanding these days that computational tractability is a fundamental issue in all problem solving. The pragmatic reasons for that are twofold.

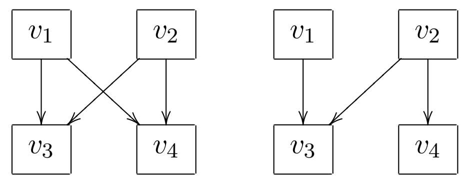
1. Many planning problems in the manufacturing and other process controlling systems are believed to be highly structured, thus have a potential to allow for efficient planning if exploiting this structure (Klein, Jonsson, & Bäckström, 1998). On the other hand, if this structure is not accounted for explicitly, a general-purpose planner is likely to go on tour in an exponential search space even for tractable problems. Moreover, since intractable theories provide no guarantees about the performance of engineering systems, in cases where such guarantees are required it is unavoidable to design the controlled system in a complexity-aware manner so that planning for it will be provably tractable (Williams & Nayak, 1996, 1997).

2. Computational tractability can be an invaluable tool even in dealing with problems that fall outside all the known tractable fragments of planning. For instance, tractable fragments of planning provide the foundations for most (if not all) rigorous heuristic estimates employed in planning as heuristic search (Bonet & Geffner, 2001; Hoffmann, 2003; Helmert, 2006; Hoffmann & Nebel, 2001; Edelkamp, 2001). This is in particular true for admissible heuristic functions for planning that are typically defined as the optimal cost of achieving the goals in an over-approximating abstraction of the planning problem in hand. Such an abstraction is obtained by relaxing certain constraints in the specification of the original problem, and the purpose of the abstraction is to provide us with a provably tractable abstract problem (Haslum, 2006; Haslum & Geffner, 2000; Haslum, Bonet, & Geffner, 2005).

Unfortunately, the palette of known tractable fragments of planning is still very limited, and the situation is even more severe for tractable optimal planning. To our knowledge, just less than a handful of non-trivial fragments of optimal planning are known to be tractable. While there is no difference in theoretical complexity of regular and optimal planning in the general case (Bylander, 1994), many of the classical planning domains are provably easy to solve, but hard to solve optimally (Helmert, 2003). Practice also provides a clear evidence for strikingly different scalability of satisficing and optimal general-purpose planners (Hoffmann & Edelkamp, 2005).

A	pre(a)				eff(a)			
	v_1	v_2	v_3	v_4	v_1	v_2	v_3	v_4
a_1	0				1			
a_2						1		
a_3	0	0					1	
a_4	0	0		0				1
a_5		1		1				0

(a)



(b) (c)

Figure 1: Example of two simple action sets that fit the characteristics of the UB fragment. (a) Unary-effect action set A over propositional variables $V = \{v_1, \ldots, v_4\}$, (b) Causal graph induced by A, (c) Causal graph induced by $A \setminus \{a_4\}$.

In this work we show that the search for new islands of tractability of optimal classical planning is far from being exhausted. Specifically, we study the complexity of optimal planning for problems specified in terms of propositional state variables, and actions that each changes the value of a single variable. In some sense, we continue the line of complexity analysis suggested in (Brafman & Domshlak, 2003), and extend it from non-optimal to optimal planning. Our results for the first time provide a dividing line between tractable and intractable such problems.

1.1 The UB (Optimal) Planning Problems

Problems of *classical planning* correspond to reachability analysis in state models with deterministic actions and complete information. In this work we focus on state models describable in a certain fragment of the SAS$^+$ formalism (Bäckström & Nebel, 1995) that allows only for propositional state variables and unary-effect actions. Following (Bäckström & Nebel, 1995), in what follows we refer to this subclass of SAS$^+$ as UB(short for "**u**nary-effect, **b**inary-valued"). Somewhat surprisingly, even non-optimal planning for UB is PSPACE-complete, that is, as hard as general propositional planning (Bylander, 1994).

Definition 1 *A* **SAS$^+$ problem instance** *is given by a quadruple* $\Pi = \langle V, A, I, G \rangle$, *where:*

- $V = \{v_1, \ldots, v_n\}$ *is a set of* state variables, *each associated with a finite domain* $Dom(v_i)$*; the* initial state I *is a complete assignment, and the* goal G *is a partial assignment to* V*, respectively.*

- $A = \{a_1, \ldots, a_N\}$ *is a finite set of* actions, *where each action* a *is a pair* $\langle \mathsf{pre}(a), \mathsf{eff}(a) \rangle$ *of partial assignments to* V *called* preconditions *and* effects, *respectively. Each action* $a \in A$ *is associated with a non-negative real-valued cost* $\mathcal{C}(a)$. *An action* a *is applicable in a state* $s \in Dom(V)$ *iff* $s[v] = \mathsf{pre}(a)[v]$ *whenever* $\mathsf{pre}(a)[v]$ *is specified. Applying an applicable action* a *changes the value of* v *to* $\mathsf{eff}(a)[v]$ *if* $\mathsf{eff}(a)[v]$ *is specified.*

A SAS$^+$ *problem instance belongs to the* **fragment UB** *of* SAS$^+$ *iff all the state variables in V are binary-valued, and each action changes the value of exactly one variable, that is, for all $a \in A$, we have $|\mathsf{eff}(a)| = 1$.*

Different sub-fragments of UB can be defined by placing syntactic and structural restrictions on the actions sets of the problems. For instance, Bylander shows that planning in UB domains where each action is restricted to have only positive preconditions is tractable, yet optimal planning for this UB fragment is hard. In general, the seminal works of Bylander and Erol *at al.* indicate that extremely severe syntactic restrictions on single actions are required to guarantee tractability, or even membership in NP (Bylander, 1994; Erol, Nau, & Subrahmanian, 1995). Bäckström and Klein (1991), consider syntactic restrictions of a more global nature, and show that UB planning is tractable if the preconditions of any two actions do not require different values for non-changing variables, and no two actions have the same effect. Interestingly, this fragment of UB, known as **PUBS**, remains tractable for optimal planning as well. While the characterizing properties of PUBS are very restrictive, this result of Bäckström and Klein provided an important milestone in the research on planning tractability.

Given the limitations of syntactic restrictions observed in (Bylander, 1994; Erol et al., 1995; Bäckström & Klein, 1991), more recent works have studied the impact of posing structural and mixed structural/syntactic restrictions on the action sets. In the scope of UB, most of these works relate between the complexity of planning and the topological properties of the problem's *causal graph* structure.

Definition 2 *The* **causal graph** $CG(\Pi)$ *of a* UB *problem* $\Pi = \langle V, A, I, G \rangle$ *is a digraph over the nodes V. An arc (v, v') belongs to $CG(\Pi)$ iff $v \neq v'$ and there exists an action $a \in A$ changing the value of v' while being preconditioned by some value of v, that is, both* $\mathsf{eff}(a)[v']$ *and* $\mathsf{pre}(a)[v]$ *are specified.*

Informally, the immediate predecessors of v in $CG(\Pi)$ are all those variables that directly affect our ability to change the value of v, and it is evident that constructing the causal graph $CG(\Pi)$ of any given UB planning problem Π is straightforward. For instance, consider the action set depicted in Figure 1a. It is easy to verify that all the actions in this set are unary effect. The causal graph induced by this action set is depicted in Figure 1a. The actions a_1 and a_2 are the only actions that change the values of v_1 and v_2, respectively, and these actions have no preconditions outside the affected variables. Hence, the causal graph contains no arcs incoming to the nodes v_1 and v_2. On the other hand, the actions changing v_3 and v_4 are preconditioned (in both cases) by the values of v_1 and v_2, and thus both v_3 and v_4 have incoming arcs from v_1 and v_2.

Way before being used for the planning complexity analysis, causal graphs have been (sometimes indirectly) considered in the scope of hierarchically decomposing planning tasks (Newell & Simon, 1961; Sacerdoti, 1974; Knoblock, 1994; Tenenberg, 1988; Bacchus & Yang, 1994). The first result relating between the complexity of UB planning and the structure of the causal graph is due to Bäckström and Jonsson (1995, 1998) that identify a fragment of UB, called **3S**, which has an interesting property of inducing tractable plan existence yet intractable plan generation. One of the key characteristics of **3S** is the acyclicity of the causal

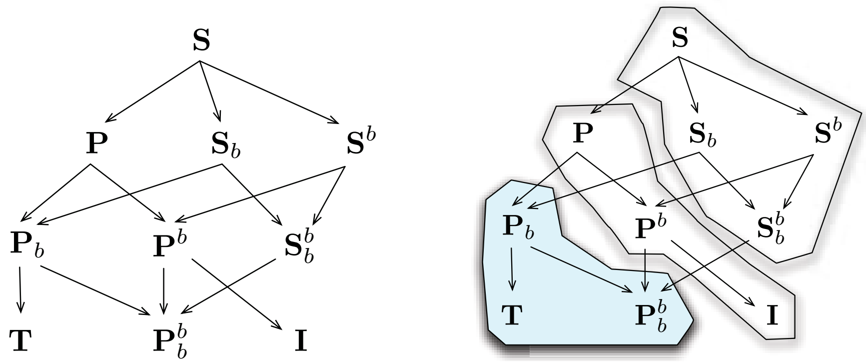


Figure 2: Inclusion-based hierarchy and complexity of plan generation for some UB problems with acyclic causal graphs. (a) The hierarchy of STRIPS fragments corresponding to tree, inverted tree, polytree, and directed-path singly connected topologies of the causal graph, and (possibly) $O(1)$ bounds on the causal graph in-degree and/or out-degree. (b) Plan generation is tractable for the fragments in the (bottom-most) shaded region, and NP-complete for all other depicted fragments. The intermediate and top-most (empty) regions corresponds to the results of Brafman and Domshlak (2003) and Jonsson and Gimenez (2007), respectively.

graphs. A special case of **3S** is also independently studied by Williams and Nayak (1997) in the scope of incremental planning for more general SAS$^+$ problems.

More recently, Brafman and Domshlak (2003) provide a detailed account of the complexity of finding plans for UB problems with acyclic causal graphs. These results are most closely related to the problems examined in our paper, and thus we survey them in more details. For ease of presentation, here we already introduce certain notation that is heavily used throughout the paper anyway.

– For each node $v \in CG(\Pi)$, by $\mathsf{In}(v)$ and $\mathsf{Out}(v)$ we denote the in- and out-degrees of v, respectively, and $\mathsf{In}(CG(\Pi))/\mathsf{Out}(CG(\Pi))$ stand for the maximal in-degree/out-degree of the $CG(\Pi)$ nodes.

– Assuming $CG(\Pi)$ is connected[1], we provide a special notation to the following topologies of acyclic causal graphs. A causal $CG(\Pi)$ is a

T *tree* if $\mathsf{In}(CG(\Pi)) \leq 1$, and there exists $v \in V$ such that $\mathsf{In}(v) = 0$.

I *inverted tree* if $\mathsf{Out}(CG(\Pi)) \leq 1$, and there exists $v \in V$ such that $\mathsf{Out}(v) = 0$.

P *polytree* if $CG(\Pi)$ contains no undirected cycles. (For an example of a polytree that is neither tree nor inverted tree see Figure 1c.)

1. If $CG(\Pi)$ consists of a few connected components, then these components identify independent subproblems of Π that can be easily identified and treated separately.

S *directed-path singly connected* if there is at most one directed path from each node $v \in CG(\Pi)$ to any other node $v' \in CG(\Pi)$. (For an example of a directed-path singly connected DAG see Figure 1b.)

In what follows, we use **T**, **I**, **P**, and **S** to refer to the corresponding fragments of UB, and we use subscript/superscript b to refer to a fragment induced by the additional constraint of in-degree/out-degree being bounded by a constant. It is not hard to verify that we have $\mathbf{T}, \mathbf{I} \subset \mathbf{P} \subset \mathbf{S}$, with $\mathbf{T} \subset \mathbf{P}_b$ and $\mathbf{I} \subset \mathbf{P}^b$; the complete inclusion hierarchy of these sub-fragments of UB is shown in Figure 2a.

The key tractability result in (Brafman & Domshlak, 2003) corresponds to a polynomial time plan generation procedure for $\mathbf{P}_b$, that is, for UB problems inducing polytree causal graphs with all nodes having $O(1)$-bounded indegree. In addition, Brafman and Domshlak show that plan generation is NP-complete for the fragment **S**, and we note that the proof of this claim can be easily modified to hold for $\mathbf{S}_b^b$. These results of tractability and hardness (as well as their immediate implications) are depicted in Figure 2b by the shaded bottom-most and the empty top-most free-shaped regions. The empty free-shaped region in between corresponds to the gap left in (Brafman & Domshlak, 2003). This gap has been recently closed by Jonsson and Gimenez (2007) that prove NP-completeness of plan generation for **P**. We note that the proof of Jonsson and Gimenez actually carries out to the **I** fragment as well, and so the gap left in (Brafman & Domshlak, 2003) is now entirely closed.

1.2 Summary of Results

The complexity results in both (Brafman & Domshlak, 2003) and (Jonsson & Gimenez, 2007) are devoted to plan generation, and do not distinguish between the plans on the basis of their quality. In contrast, here we study the complexity of *optimal plan generation* for UB, focusing on (probably the most canonical) *cost-optimal* (also known as sequentially-optimal) planning. Cost-optimal planning corresponds to the task of finding a plan $\rho \in A^*$ that minimizes $\mathcal{C}(\rho) = \sum_{a \in \rho} \mathcal{C}(a)$. We provide novel tractability results for cost-optimal planning for UB, and draw a dividing line between the tractable and intractable such problems. Almost all our tractability results are based on a proof technique that connects between certain tools from planning and tractable constraint optimization. We believe that this "technical" contribution of the paper is of interest on its own due to a clear evidence for its robustness—-different our algorithms exploit this proof technique in very much different manners.

1.2.1 Cost-optimal planning for $\mathbf{P}_b$

Following (Brafman & Domshlak, 2003), here we relate between the complexity of (cost-optimal) UB planning and the topology of the causal graph, and for that we consider the structural hierarchy depicted in Figure 2a. We begin with considering cost-optimal planning for $\mathbf{P}_b$—it is apparent from Figure 2b that this is the most expressive fragment of the hierarchy that is still a candidate for tractable cost-optimal planning. Our first positive result affirms this possibility, showing that the complexity map of the cost-optimal planning for the UB fragments in Figure 2a is *identical* to this for regular plan generation (that is, Figure 2b).

Our algorithm for $\mathbf{P}_b$ is based on compiling the given $\mathbf{P}_b$ problem Π into a *constraint optimization problem* $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ over variables $\mathcal{X}$, functional components $\mathcal{F}$, and the global objective $\min \sum_{\varphi \in \mathcal{F}} \varphi(\mathcal{X})$ such that

(I) COP_Π can be constructed in time polynomial in the description size of Π,

(II) the tree-width of the cost network of COP_Π is bounded by a constant, and the optimal tree-decomposition of the network is given by the compilation process,

(III) if Π is unsolvable then all the assignments to $\mathcal{X}$ evaluate the objective function to ∞, and otherwise, the optimum of the global objective is obtained on and only on the assignments to $\mathcal{X}$ that correspond to SO-plans for Π,

(IV) given an optimal solution to COP_Π, the corresponding SO-plan for Π can be reconstructed from the former in polynomial time.

Having such a compilation scheme, we then solve COP_Π using the standard, poly-time algorithm for constraint optimization over trees (Dechter, 2003), and find an optimal solution for Π. The compilation is based on a certain property of the cost-optimal plans for $\mathbf{P}_b$ that allows for conveniently bounding the number of times each state variable changes its value along such an optimal plan. Given this property of $\mathbf{P}_b$, each state variable v is compiled into a single COP variable x_v, and the domain of that COP variable corresponds to all possible sequences of value changes that v may undergo along a cost-optimal plan. The functional components $\mathcal{F}$ are then defined one for each COP variable x_v, and the scope of such a function captures the "family" of the original state variable v in the causal graph, that is, v itself and its immediate predecessors in $CG(\Pi)$. For an illustration, Figure 3a depicts a causal graph of a $\mathbf{P}$ problem Π, with the family of the state variable v_4 being depicted by the shaded region, and Figure 3b shows the cost network induced by compiling Π as a $\mathbf{P}_b$ problem, with the dashed line surrounding the scope of the functional component induced by the family of v_4. It is not hard to verify that such a cost network induces a tree over variable-families cliques, and for a $\mathbf{P}_b$ problem, the size of each such clique is bounded by a constant. Hence, the tree-width of the cost-network is bounded by a constant as well.

1.2.2 Causal graphs and k-dependence

While causal graphs provide important information about the structure of the planning problems, a closer look at their definition reveals that some information used for defining causal graphs actually gets hidden by this structure. To start with an example, let us consider the multi-valued encoding of the Logistics-style problems (Helmert, 2006). In these problems, each variable representing the location of a package has as its parents in the causal graph *all* the variables representing alternative transportation means (i.e., tracks, planes, etc.), and yet, each individual action affecting the location of a package is preconditioned by at most *one* such parent variable. (You cannot load/unload a package into/from more than one vehicle.) This exemplifies the fact that, even if the in-degree of the causal graph is proportional to some problem domain's parameters, the number of variables that determine applicability of each action may still be bounded by a constant.

In other words, while the causal graph provides an aggregative view on the independence relationships between the problem variables, the individual dependencies of the problem ac-

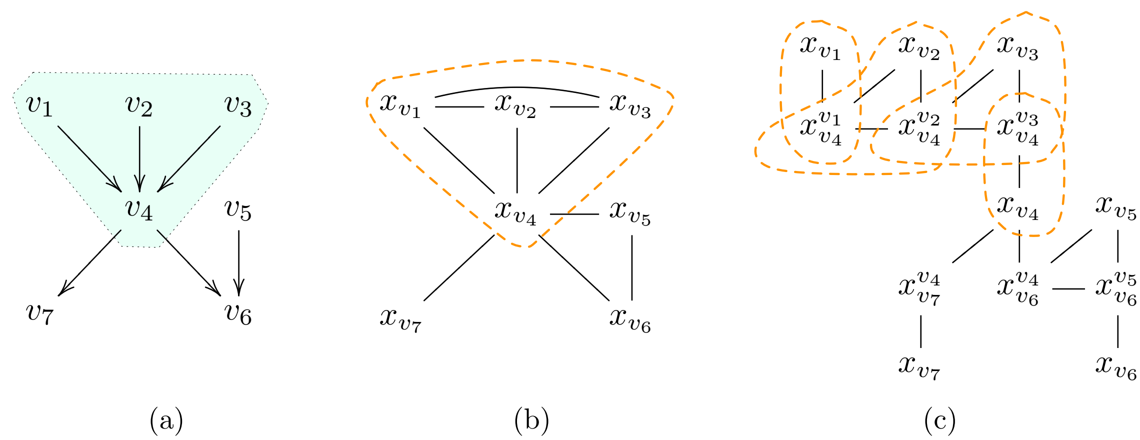


Figure 3: Cost networks induced by the planning-to-COP compilation schemes for $\mathbf{P}$. (a) Causal graph of a $\mathbf{P}$ problem Π, with the family of the state variable v_4 being depicted by the shaded region. (b) Cost network induced by compiling Π as a $\mathbf{P}_b$ problem, with the dashed line surrounding the scope of the functional component induced by the family of v_4. (c) Cost network induced by compiling Π as a $\mathbf{P}(1)$ problem, with the dashed lines surrounding the scopes of the four functional components induced by the family of v_4.

tions on the non-changing variables are getting suppressed by this view. Targeting these actual dependencies of the actions, here we define a (tangential to the causal graph's topology) classification of the UB problems, and study the connection between this classification and the computational tractability of both general and cost-optimal plan generation for UB.

Definition 3 *For any $k \in \mathbb{Z}^*$, and any* SAS$^+$ *problem instance $\Pi = (V, A, I, G)$, we say that Π is* **k-dependent** *if it satisfies*

$$\max_{a \in A} |\{v \in V \mid \mathsf{pre}(a)[v] \neq \mathsf{u} \wedge \mathsf{eff}(a)[v] = \mathsf{u}\}| \quad \leq \quad k,$$

with "$= \mathsf{u}$" being a short for "unspecified".

In other words, a SAS$^+$ problem is k-dependent if no action in its action set depends on more than k non-changing variables. Putting now our two classifications of the problems together, for any structural fragment $\mathbf{F}$ of UB (such as, e.g., these in Figure 2), and any $k \in \mathbb{Z}^*$, by $\mathbf{F}(k)$ we denote the set of all k-dependent problems within $\mathbf{F}$.

Recall that the fragment $\mathbf{P}$ of UB is NP-hard even for non-optimal planning (Jonsson & Gimenez, 2007). Our main result here is positive—at least for the most extreme (yet, says the example above, not entirely unrealistic) setting of $k = 1$, satisfying k-dependence does bring us to an island of tractability $\mathbf{P}(1)$.

Similarly to our treatment of $\mathbf{P}_b$, our algorithm for $\mathbf{P}(1)$ exploits the idea of compiling a planning problem Π into a tractable constraint optimization problem COP_Π. At the same time, the planning-to-COP compilation scheme here is very much different from this devised for $\mathbf{P}_b$. In fact, this difference is unavoidable since our construction for $\mathbf{P}_b$ heavily relies on the assumption that $\mathsf{In}(CG(\Pi)) = O(1)$, and we do not have this luxury in $\mathbf{P}(1)$. Instead, we identify certain properties of the cost-optimal plan sets of the $\mathbf{P}(1)$ problems, and exploit these properties in devising suitable planning-to-COP compilation schemes.

We begin with considering only $\mathbf{P}(1)$ problems with uniform-cost actions; the cost of a plan for such a problem is proportional to the length of the plan[2]. We show that any such solvable problem has a cost-optimal plan that makes all the changes of each variable to a certain value using exactly the same (type of) action. And while devising a correct and tractable planning-to-COP compilation scheme is more than a step away from identifying this property of $\mathbf{P}(1)$, the latter provides a critical brick that everything else lies upon it. Relying on this property of $\mathbf{P}(1)$, each state variable v and *each edge* (v, v') are uniquely compiled into COP variables x_v and $x^v_{v'}$ (see Figure 3c), and a certain *set* of functional components are then defined for each COP variable x_v. Here both the domains of the COP variables and the specification of the functional components are technically involved, and thus relegated to later in the paper. It is important, however, to note here already that the cost networks of such COPs are guaranteed to induce trees over cliques of size ≤ 3, and thus having tree-width bounded by a constant. The reader can get an intuition of where the "cliques of size ≤ 3" are coming from by looking on the example depicted in Figure 3c.

Unfortunately, the aforementioned helpful property of the $\mathbf{P}(1)$ problems with uniform-cost actions does not hold for more general action-cost schemes for $\mathbf{P}(1)$. On the other hand, turns out that *all* problems in $\mathbf{P}(1)$ satisfy another property that still allows for devising a general, correct, and tractable planning-to-COP scheme for $\mathbf{P}(1)$. Specifically, we show that any solvable problem in $\mathbf{P}(1)$ has a cost-optimal plan that makes all the changes of each variable using at most three types of action. The algorithm resulting from exploiting this property is poly-time, yet it is more complex and more costly than this devised for the $\mathbf{P}(1)$ problems with uniform-cost actions. Interestingly, the cost networks of such COPs are topologically identical to these for problems with uniform-cost actions, with the difference being in the domains of the COP variables, and in the specification of the actual functional components.

Having read this far, the reader may rightfully wonder whether $O(1)$-dependence is not a strong enough property to make the cost-optimal planning tractable even for some more complex than polytree forms of the causal graph. Turns out that that the dividing line between tractable and intractable problems is much more delicate. Figure 4 summaries our current understanding of time complexity of both cost-optimal and satisficing plan generation for the $\mathbf{P}$ and $\mathbf{S}$ fragments of UB. First, we show that even the satisficing planning with directed-path singly connected, bounded in- and out-degree causal graphs is hard under 2-dependence, and that cost-optimal planning for this structural fragment of UB is hard even for 1-dependent such problems. In addition, we note that the hardness of planning for $\mathbf{P}(3)$ can be easily derived from the proof of hardness of Jonsson and Gimenez for $\mathbf{P}$. Note that the complexity of (both cost-optimal and satisficing) plan generation for

2. This is probably the origin for the term "sequential optimality".

	$k=1$	$k=2$	$k=3$	$k=\Theta(n)$
$\mathbf{P}_b$	—	—	—	P
$\mathbf{P}(k)$	P		NPC	NPC
$\mathbf{S}_b^b$	NPC	—	—	NPC

(a)

	$k=1$	$k=2$	$k=3$	$k=\Theta(n)$
$\mathbf{P}_b$	—	—	—	P
$\mathbf{P}(k)$	P		NPC	NPC
$\mathbf{S}_b^b$		NPC	—	NPC

(b)

Figure 4: Complexity of (a) cost-optimal and (b) satisficing plan generation for fragments of UB. The "—" mark indicates that the complexity is implied by other results in the row. All other shaded and regular cells correspond to the results obtained in this work and in the past, respectively. Empty cells correspond to open questions. Note that the only difference in our understanding of cost-optimal and satisficing planning here is the complexity of planning for $\mathbf{S}(1)$.

$\mathbf{P}(2)$ remains an interesting open problem. An additional open problem left by this work is the complexity of satisficing plan generation for $\mathbf{S}(1)$.

1.3 Remarks

Our goal in this work has been identifying new islands of tractability for cost-optimal planning, by that improving our understanding of what makes the planning problems hard or easy to solve. Of our lesser interest here was to make the poly-time algorithms practically efficient by reducing their (quite prohibitive) polynomial time complexity. In fact, in some places we intentionally sacrificed some possible optimizations to keep the already involved constructions as apprehensible as possible. Therefore, it is more than likely that the time complexity of our planning-to-COP algorithms can be further improved, or some conceptually different algorithmic ideas will be found more appropriate for the problems in question. In addition, much more efficient algorithms may work for some special cases of the general tractable families. For instance, in the paper we illustrate such possibility by presenting a low poly-time algorithm for UB problems with tree causal graphs (that is, the $\mathbf{T}$ fragment) and uniform-cost actions.

Of course, the reader may ask whether striving for practical efficiency in solving various special fragments of planning is any motivated. As it appears from our discussion at the beginning of the paper, we believe that the answer to this question is "yes". While most of the research on AI planning is rightfully devoted to solving general planning problems, many tools developed and employed in that research rely on tractable fragments of planning. And if, for instance, to obtain a heuristic estimator for a problem, the problem is projected to (or embedded in) a tractable "relaxed" problem, then we will be happy to know that the latter can be solved in low poly-time. On the other hand, making a tractable fragment also practically efficient is probably worth the effort only in face of some concrete "customer" of that fragment in practice.

1.4 A Guide to the Rest of the Paper

So far in the paper our aim was to provide an adequate description of our results for readers who do not want to delve into formal details, or just prefer not to do it in the first reading of the paper[3]. What follows provide the formal definitions, constructions, and proofs underlying these results.

2. Definitions and Notation

Starting with Definitions 1-3 from the previous section, here we introduce some additional definitions and notation that are used throughout the paper.

In contrast to the well-known STRIPS formalism for propositional planning, here we assume that all our actions are "value changing", and this in contrast to "value setting". That is, we have $\mathsf{eff}(a)[v]$ being specified only if $\mathsf{pre}(a)[v]$ is also specified, in which case we have $\mathsf{eff}(a)[v] \neq \mathsf{pre}(a)[v]$. While in general this assumption requires an exponential time translation, in case of unary-effect actions the translation takes only linear time. Given a UB problem $\Pi = \langle V, A, I, G \rangle$, by $A_v \subseteq A$ we denote the **actions that change the value of** v. Note that the unary-effectness of Π implies that $A_{v_1}, \ldots, A_{v_n}$ is a partition of the problem actions A. Considering the applicability of actions, in SAS$^+$ it also helps to give a special attention and notation to the action preconditions over the non-changing variables. The custom name for such preconditions is **prevail conditions** (Bäckström & Klein, 1991). For example, having truck T and package P in location L are both preconditions of loading P into T in L, but only the former is a prevail condition of this action because the truck is still in L after loading P, while P is no longer there (but inside T).

One of the key properties of cost-optimal plans for the UB problems with directed-path singly connected causal graphs is immediately derivable from Lemma 1 of Brafman and Domshlak (2003), and it is given by Corollary 1 below. Given a UB problem $\Pi = \langle V, A, I, G \rangle$, a variable subset $V' \subseteq V$, and an arbitrary sequence of actions $\rho \in A^*$, by $\rho\!\downarrow_{V'}$ we denote the **order-preserving restriction** of ρ to the actions $\bigcup_{v \in V'} A_v$. If the restriction is with respect to a singleton set $V' = \{v\}$, then we allow writing $\rho\!\downarrow_{\{v\}}$ simply as $\rho\!\downarrow_v$.

Lemma 1 ((Brafman & Domshlak, 2003)) *For any solvable problem $\Pi \in \mathbf{S}$ over n state variables, any irreducible plan ρ for Π, and any state variable v in Π, the number of value changes of v along ρ is $\leq n$, that is, $|\rho\!\downarrow_v| \leq n$.*

Corollary 1 *For any solvable problem $\Pi \in \mathbf{S}$ over n state variables, any cost-optimal plan ρ for Π, and any state variable v in Π, we have $|\rho\!\downarrow_v| \leq n$.*

Given an $\mathbf{S}$ problem $\Pi = \langle V, A, I, G \rangle$, we denote the initial value $I[v]$ of for each variable $v \in V$ by b_v, and the opposite value by w_v (short for, **black/white**). Using this notation and exploiting Corollary 1, by $\sigma(v)$ we denote the **longest possible sequence of values obtainable by** v **along a cost-optimal plan** ρ, with $|\sigma(v)| = n+1$, b_v occupying all the

3. We adopted this format from the seminal paper of Bylander (1994) as we feel it has contributed something to making that paper an extremely enjoyable reading.

odd positions of $\sigma(v)$, and w_v occupying all the even positions of $\sigma(v)$. In addition, by $\tau(v)$ we denote a per-value **time-stamping of** $\sigma(v)$

$$\tau(v) = \begin{cases} \mathsf{b}_v^1 \cdot \mathsf{w}_v^1 \cdot \mathsf{b}_v^2 \cdot \mathsf{w}_v^2 \cdots \mathsf{b}_v^{j+1}, & n = 2j, \\ \mathsf{b}_v^1 \cdot \mathsf{w}_v^1 \cdot \mathsf{b}_v^2 \cdot \mathsf{w}_v^2 \cdots \mathsf{w}_v^j, & n = 2j-1, \end{cases}, j \in \mathbb{N}.$$

The sequences $\sigma(v)$ and $\tau(v)$ play an important role in our constructions both by themselves and via their prefixes and suffixes. In general, for any sequence S, by $\trianglerighteq[S]$ and $\trianglelefteq[S]$ we denote the set of all non-empty **prefixes** and **suffixes** of S, respectively. In our context, a prefix $\sigma' \in \trianglerighteq[\sigma(v)]$ is called **goal-valid** if either the goal value $G[v]$ is unspecified, or the last element of σ' equals $G[v]$. The set of all goal-valid prefixes of $\sigma(v)$ is denoted by $\trianglerighteq^*[\sigma(v)] \subseteq \trianglerighteq[\sigma(v)]$. The notion of goal-valid prefixes is also similarly specified for $\tau(v)$.

Finally, given a SAS$^+$ problem $\Pi = \langle V, A, I, G\rangle$, a subset of state variables $V' \subseteq V$, and an action sequence $\rho \in A^*$, we say that ρ is **applicable with respect to** V' if restricting the preconditions and effects of the actions in ρ to the variables V' makes ρ applicable in I.

3. Cost-Optimal Planning for $\mathbf{P}_b$

This section is devoted to the proof of tractability of cost-optimal planning for the problem fragment $\mathbf{P}_b$. We begin with describing our planning-to-COP scheme for $\mathbf{P}_b$, and then prove its correctness and complexity. Finally, we present a interesting subset of $\mathbf{P}_b$ for which cost-optimal planning is not only tractable, but also practically efficient.

3.1 Construction

Before we proceed with the details of the construction, here we make an assumption that the specification of our actions is fully specified in terms of the variable parents in the causal graph. If $\mathsf{pred}(v) \subset V$ denotes the set of all the immediate predecessors of v in the causal graph $CG(\Pi)$, then we assume that, for each action $a \in A_v$, $\mathsf{pre}(a)[w]$ is specified for each $w \in \mathsf{pred}(v)$. While in general such an assumption requires an exponential translation, this is not the case with $\mathbf{P}_b$. Let $A^{\bowtie}$ be such a translation of the original problem actions A. To obtain $A^{\bowtie}$, for every variable $v \in V$, every action in A_v is represented in $A^{\bowtie}$ by a set of actions that are preconditioned by complete assignments to $\mathsf{pred}(v)$. If $|\mathsf{pred}(v)| = k$, and the precondition of a is specified only in terms of some $0 \le k' \le k$ parents of v, then a is represented in $A^{\bowtie}$ by a set of actions, each extending the precondition $\mathsf{pre}(a)$ by a certain instantiation of the previously unspecified parents of v, and having the cost $\mathcal{C}(a') = \mathcal{C}(a)$. Note that the expansions of two or more original actions may overlap, and thus $A^{\bowtie}$ may contain syntactically identical yet differently priced actions. Without loss of generality, one can assume that only a minimally-priced such clone is kept in $A^{\bowtie}$. The key point is that that compiling A into $A^{\bowtie}$ for the $\mathbf{P}_b$ problems is poly-time, as the procedure is linear in $|A^{\bowtie}| = O(n2^{\mathsf{In}(\Pi)+1})$. Finally, the (straightforward to prove) Proposition 1 summarizes the correctness of our assumption with respect to the cost-optimal planning for UB.

Proposition 1 *For any* UB *problem* $\Pi = \langle V, A, I, G\rangle$*, the cost of the optimal plans for* Π *is equal to this for* $\Pi^{\bowtie} = \langle V, A^{\bowtie}, I, G\rangle$*, with optimal plans for* Π *being reconstructible in linear time from the optimal plans for* $\Pi^{\bowtie}$ *and vice versa.*

We now specify our compilation of a given $\mathbf{P}_b$ problem Π into a constraint optimization problem COP_Π. The variable set $\mathcal{X}$ contains a variable x_v for each planning variable $v \in V$, and the domain $Dom(x_v)$ consists of all valid prefixes of $\tau(v)$. That is,

$$\begin{aligned} \mathcal{X} &= \{x_v \mid v \in V\} \\ Dom(x_v) &= \trianglerighteq^*[\tau(v)] \end{aligned} \tag{1}$$

Informally, the domain of each variable x_v contains all possible sequences of values that the planning variable v may undergo along a cost-optimal plan. Now, for each planning variable v with parents $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, the set $\mathcal{F}$ contains a single non-negative, real-valued function φ_v with the scope

$$Q_v = \{x_v, x_{w_1}, \ldots, x_{w_k}\} \tag{2}$$

The purpose of these functions is to connect between the value-changing sequences of v and these of its parents $\mathsf{pred}(v)$. The specification of these functions is the more involved part of the compilation.

First, for each planning variable v with $\mathsf{pred}(v) = \emptyset$, and each of its goal-valid (time-stamped) value-changing sequences $\tau' \in \trianglerighteq^*[\tau(v)]$, we set

$$\varphi_v(\tau') = \left\lfloor \frac{|\tau'|}{2} \right\rfloor \cdot \mathfrak{C}(a_{\mathsf{w}_v}) + \left\lfloor \frac{|\tau'|-1}{2} \right\rfloor \cdot \mathfrak{C}(a_{\mathsf{b}_v}) \tag{3}$$

where $\mathsf{eff}(a_{\mathsf{w}_v})[v] = \{\mathsf{w}_v\}$, $\mathsf{eff}(a_{\mathsf{b}_v})[v] = \mathsf{b}_v$, and $\mathfrak{C}(a) = \mathcal{C}(a)$ if $a \in A$, and ∞, otherwise. It is not hard to verify that $\varphi_v(\tau')$ corresponds to the optimal cost of performing $|\sigma'| - 1$ value changes of v in Π.

Now, for each non-root variable v with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}, k \geq 1$, we specify the function φ_v as follows. For each goal-valid value-changing sequence $\tau' \in \trianglerighteq^*[\tau(v)]$ of v, and each set of such goal-valid value-changing sequences $\{\tau'_1 \in \trianglerighteq^*[\tau(w_1)], \ldots, \tau'_k \in \trianglerighteq^*[\tau(w_k)]\}$, we want to set $\varphi_v(\tau', \tau'_1, \ldots, \tau'_k)$ to the *optimal cost of performing* $|\tau'| - 1$ *value changes of* v*, given that* $w_1, \ldots, w_k$ *change their values* $|\tau'_1| - 1, \ldots, |\tau'_k| - 1$ *times*, respectively. In what follows, we reduce setting the value $\varphi_v(\tau', \tau'_1, \ldots, \tau'_k)$ to solving a single-source shortest path problem on an edge-weighted digraph $G'_e(v)$ that slightly enhances a similarly-named graphical structure suggested in (Brafman & Domshlak, 2003). Though the construction of $G'_e(v)$ is very similar to this in (Brafman & Domshlak, 2003), here we provide it in full details to save the reader patching the essential differences on the construction in (Brafman & Domshlak, 2003).

Given the value-changing sequences $\tau'_1, \ldots, \tau'_k$ as above, the digraph $G'_e(v)$ is created in three steps. First, we construct a labeled directed graph $G(v)$ capturing information about all sequences of assignments on $\mathsf{pred}(v)$ that can enable n or less value flips of v. The graph $G(v)$ is defined as follows:

1. $G(v)$ consist of $\eta = \max_{\tau' \in \trianglerighteq^*[\tau(v)]} |\tau'|$ nodes.

2. $G(v)$ forms a *2-colored multichain*, i.e., (i) the nodes of the graph are colored with black (b) and white (w), starting with black; (ii) there are no two subsequent nodes with the same color; (iii) for $1 \leq i \leq \eta - 1$, edges from the node i are only to the node $i + 1$.

Observe that such a construction of $G(v)$ promises that the color of the last node will be consistent with the goal value $G[v]$ if such is specified.

3. The nodes of $G(v)$ are denoted precisely by the elements of the longest goal-valid value-changing sequence $\tau' \in \trianglerighteq^*[\tau(v)]$, that is, b_v^i stands for the ith black node in $G(v)$.

4. Suppose that there are m operators in A_v that, under different preconditions, change the value of v from b_v to w_v. In this case, for each i, there are m edges from b_v^i to w_v^i, and $|A_v| - m$ edges from w_v^i to b_v^{i+1}. Each such edge e is labeled by the prevail conditions of the corresponding action, which is a k-tuple of the values of $w_1, \ldots, w_k$, as well as with the cost of the action. This compound label of e is denoted by $l(e)$, and the prevail condition and cost parts of $l(e)$ are denoted henceforth by $\mathsf{prv}(e)$ and $\mathcal{C}(e)$, respectively.

As the formal definition of $G(v)$ is somewhat complicated, here we provide an illustrating example. Suppose that we are given a $\mathbf{P}_b$ problem over 5 variables, and we consider a variable v with $\mathsf{pred}(v) = \{u, w\}$, $I[v] = \mathsf{b}_v$, and $G[v] = \mathsf{w}_v$. Let

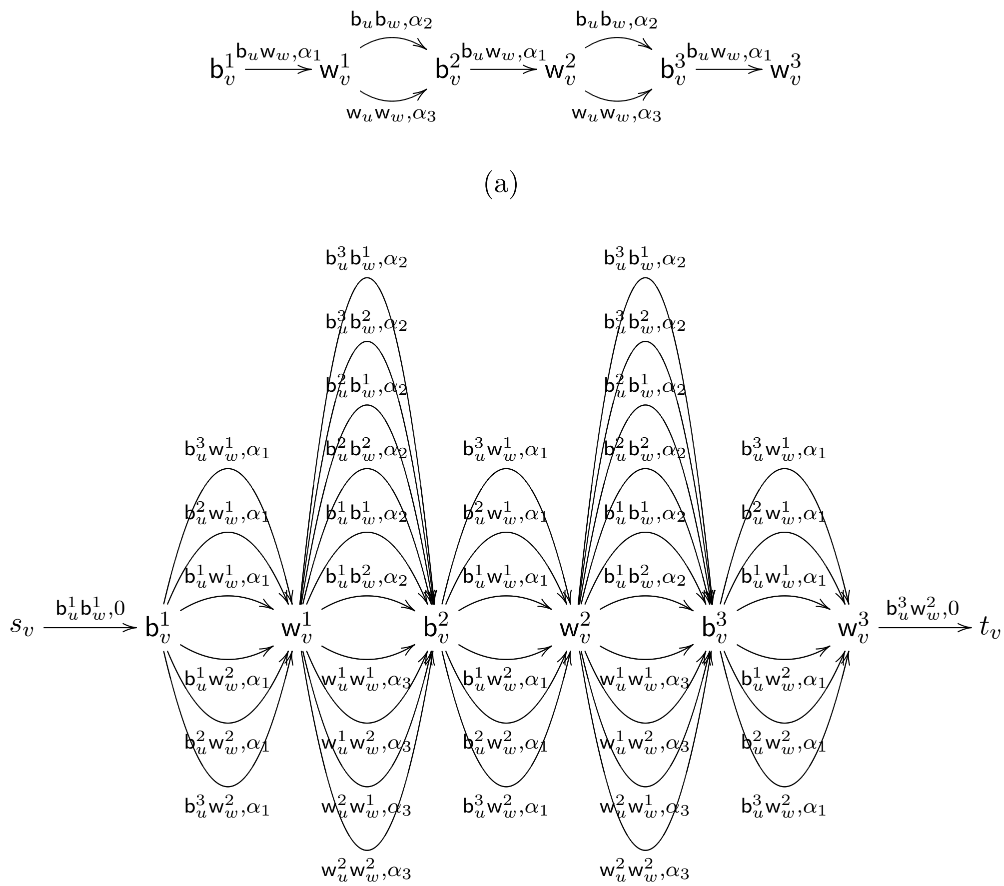
$$A_v = \begin{cases} a_1 \ : \ \mathsf{pre}(a_1) = \{\mathsf{b}_v, \mathsf{b}_u, \mathsf{w}_w\}, \ \mathsf{eff}(a_1) = \{\mathsf{w}_v\}, \ \mathcal{C}(a_1) = \alpha_1 \\ a_2 \ : \ \mathsf{pre}(a_2) = \{\mathsf{w}_v, \mathsf{b}_u, \mathsf{b}_w\}, \ \mathsf{eff}(a_2) = \{\mathsf{b}_v\}, \ \mathcal{C}(a_2) = \alpha_2 \\ a_3 \ : \ \mathsf{pre}(a_3) = \{\mathsf{w}_v, \mathsf{w}_u, \mathsf{w}_w\}, \ \mathsf{eff}(a_3) = \{\mathsf{b}_v\}, \ \mathcal{C}(a_3) = \alpha_3 \end{cases}$$

The corresponding graph $G(v)$ is depicted in Figure 5a. Informally, the graph $G(v)$ captures information about all *potentially possible* executions of the actions in A_v along a cost-optimal plan for Π. Each path started at the source node of $G(v)$ uniquely corresponds to one such an execution. Although the number of these alternative executions may be exponential in n, their graphical representation via $G(v)$ is compact—the number of edges in $G(v)$ is $O(n \cdot |A_v|)$. Note that the information about the number of times each action in A_v can be executed is not captured by $G(v)$. The following two steps add this essential information into the graphical structure.

At the second step, the digraph $G(v) = (V, E)$ is expanded into a digraph $G'(v) = (V', E')$ by substituting each edge $e \in E$ with a set of edges (between the same nodes), but with the labels corresponding to all possible assignments of the elements of $\tau'_1, \ldots, \tau'_k$ to $\mathsf{prv}(e)$. For example, an edge $e \in E$ labeled with $\|\mathsf{b}_{w_1}\mathsf{b}_{w_2}, 10\|$ might be substituted in E' with edges labeled with $\{\|\mathsf{b}_{w_1}^1\mathsf{b}_{w_2}^1, 10\|, \|\mathsf{b}_{w_1}^1\mathsf{b}_{w_2}^2, 10\|, \|\mathsf{b}_{w_1}^2\mathsf{b}_{w_2}^1, 10\|, \ldots\}$. Finally, we set $V' = V \cup \{s_v, t_v\}$, and add a single edge labeled with the first elements of $\tau'_1, \ldots, \tau'_k$ and zero cost (that is, $\|\mathsf{b}_{w_1}^1 \cdots \mathsf{b}_{w_k}^1, 0\|$) from s_v to the original source node b_v^1, plus a single edge labeled with the last elements of $\tau'_1, \ldots, \tau'_k$ and zero cost from the original sink node of $G(v)$ to t_v. Figure 5b illustrates $G'(v)$ for the example above, assuming $\tau'_u = \mathsf{b}_u^1 \cdot \mathsf{w}_u^1 \cdot \mathsf{b}_u^2 \cdot \mathsf{w}_u^2 \cdot \mathsf{b}_u^3$ and $\tau_w = \mathsf{b}_w^1 \cdot \mathsf{w}_w^1 \cdot \mathsf{b}_w^2 \cdot \mathsf{w}_w^2$.

Informally, the digraph $G'(v)$ can be viewed as a projection of the value-changing sequences $\tau'_1, \ldots, \tau'_k$ on the base digraph $G(v)$. The digraph $G'_e(v) = (V'_e, E'_e)$ is then constructed from $G'(v)$ as follows.

(i) The nodes V'_e correspond to the *edges* of $G'(v)$.



(a)

(b)

Figure 5: Example of the graphs (a) $G(v)$, and (b) $G'(v)$.

(ii) The edges $(v_e, v_{e'}) \in E'_e$ correspond to all pairs of immediately consecutive edges $e, e' \in E'$ such that, for $1 \leq i \leq k$, either $\mathsf{prv}(e)[w_i] = \mathsf{prv}(e')[w_i]$, or $\mathsf{prv}(e')[w_i]$ appears after $\mathsf{prv}(e)[w_i]$ along τ'_i.

(iii) Each edge $(v_e, v_{e'}) \in E'_e$ is weighted with $\mathcal{C}(e')$.

Figure 6 depicts the graph $G'_e(v)$ for our example.

Assuming $\alpha_3 \leq \alpha_2$, the dashed edges correspond to the *minimal-cost path of length 5* from the dummy source node $\mathsf{b}^1_u\mathsf{b}^1_w$. Note that, if the costs of actions A_v is all we care about, this path corresponds to the cost-optimal sequence of 5 value changes of v starting from its initial value b_v in Π. In fact, not only this path corresponds to such a cost-optimal sequence, but it also explicitly describes the underlying sequence of actions from A_v, as well as the total cost of these actions. Finally, for all $0 \leq i \leq n$, such minimal-cost paths of length i can be determined by running on $G'_e(v)$ a low-polynomial single-source shortest paths algorithm

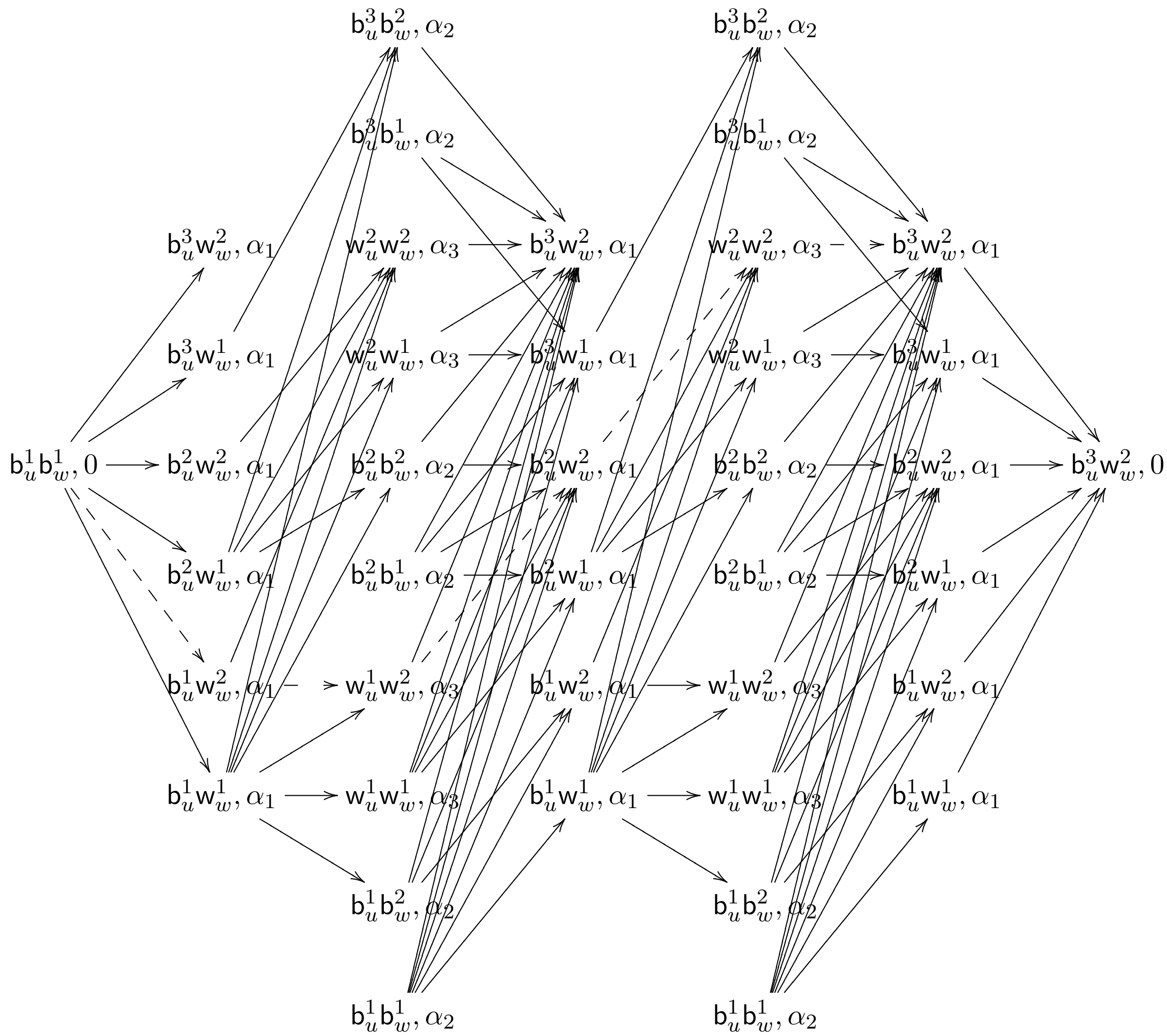


Figure 6: The graph $G'_e(v)$ constructed from the graph $G'(v)$ in Fugure 5b.

of Dijkstra (Cormen, Leiserson, & Rivest, 1990). This property of the graph $G'_e(v)$ provides us with the last building block for our algorithm for cost-optimal planning for $\mathbf{P}_b$.

The algorithm is depicted in Figure 7. Given a problem $\Pi \in \mathbf{P}_b$, the algorithm compiles it into the constraint optimization problem COP_Π, and solves it using a standard algorithm for constraint optimization over tree constraint networks (Dechter, 2003). The specification of COP_Π has already been explained inline. We believe it is already intuitive that this compilation takes time polynomial in the description size of Π, but in the next section we also prove it formally. Solving COP_Π using the algorithm for tree-structured constraint networks can be done in time linear in the description size of COP_Π because

(i) the tree-width of the cost network of COP_Π is bounded by a constant that bounds the in-degree of the causal graph, and

procedure polytree-k-indegree($\Pi = (V, A, I, G)$)
    takes a problem $\Pi \in \mathbf{P}_b$
    returns an optimal plan for Π if solvable, and fails otherwise
create a set of variables $\mathcal{X}$ and set their domains as in Eq. 1
create a set of functions $\mathcal{F} = \{\varphi_v \mid v \in V\}$ with scopes as in Eq. 2
for each $v \in V$ **do**
  if $\mathsf{pred}(v) = \emptyset$ **then**
    specify φ_v according to Eq. 3
  elseif $\mathsf{pred}(v) = \{w_1, \dots, w_k\}$ **then**
    construct graph $G(v)$
    for each k-tuple $\tau'_1 \in \trianglerighteq^*[\tau(w_1)], \dots, \tau'_k \in \trianglerighteq^*[\tau(w_k)]$ **do**
      construct graph $G'(v)$ from graph $G(v)$ and sequences $\tau'_1, \dots, \tau'_k$
      construct graph $G'_e(v)$ from graph $G'(v)$
      for each goal-valid sequence $\tau' \in \trianglerighteq^*[\tau(v)]$ **do**
        $\pi :=$ minimal-cost path of $|\tau'| - 1$ edges
          from the source node $\langle \mathsf{b}_{w_1} \cdots \mathsf{b}_{w_k} \rangle$ of $G'_e(v)$
        if returned π **then**
          $\varphi_v(\tau', \tau'_1, \dots, \tau'_k) := \mathcal{C}(\pi)$
        else
          $\varphi_v(\tau', \tau'_1, \dots, \tau'_k) := \infty$
        endif
      endfor
    endfor
  endif
endfor
set $\mathsf{COP}_\Pi := (\mathcal{X}, \mathcal{F})$ with global objective $\min \sum_{\varphi \in \mathcal{F}} \varphi(\mathcal{X})$
$\overline{x} := \mathsf{solve\text{-}tree\text{-}cop}(\mathsf{COP}_\Pi)$
if $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \infty$ **then return** failure
extract plan ρ from $\overline{x}$ with $\mathcal{C}(\rho) = \sum_{\varphi \in \mathcal{F}} \varphi(\overline{x})$
return ρ

Figure 7: Algorithm for cost-optimal planning for $\mathbf{P}_b$.

(ii) optimal tree-decomposition of the COP_Π's cost network is given by any topological ordering of the causal graph.

3.2 Correctness and Complexity

We now proceed with proving both the correctness and the polynomial time complexity of our algorithm for $\mathbf{P}$. We begin with proving a rather general property of polytrees that helps both here and in our constructs for the $\mathbf{P}(1)$ later in the paper; this property is formulated by the claim of Theorem 1. We note that a special case of this property has been already exploited in the past in the proof of Lemma 2 in (Brafman & Domshlak, 2003), but, to our knowledge, the property has never been formulated as a general claim. Throughout the paper we then demonstrate that it can be helpful in numerous situations.

Theorem 1 *Let G be a polytree over vertices $V = \{1, \dots, n\}$, and $\mathsf{pred}(i) \subset V$ denote the immediate predecessors of i in G. For each $i \in V$, let O_i be a finite set of objects associated with the vertex i, with the sets $O_1, \dots, O_n$ being pairwise disjoint. For each $i \in V$, let $>_i$ be a strict partial order over O_i, and, for each $j \in \mathsf{pred}(i)$, let $>_{i,j}$ be a strict partial order over $O_i \cup O_j$.*

If, for each $i \in V, j \in \mathsf{pred}(i)$, the transitively closed $>_i \cup >_{i,j}$ and $>_j \cup >_{i,j}$ induce (strict) partial orders over $O_i \cup O_j$, then so does the transitively closed

$$> \;=\; \bigcup_{i \in V} \left(>_i \cup \bigcup_{j \in \mathsf{pred}(i)} >_{i,j} \right)$$

over $O = \bigcup_{i \in V} O_i$.

Proof: In what follows, by o_i we denote an arbitrary object from O_i. Assume to the contrary that both $>_i \cup >_{i,j}$ and $>_j \cup >_{i,j}$ are (strict) partial orders, and yet $>$ is not so. That is, there exists a pair of objects $o_i, o_j \in O$ for which hold both $o_i > o_j$ and $o_j > o_i$. By the construction of $>$, we have that there is a, possibly empty, path between the vertices i and j in the undirected graph induced by G. Since G is a polytree, we know that such an undirected path between i and j is unique. Thus, we must have

$$\begin{array}{l} \alpha \;:\; o_i = o^1_{i_0} < \ldots < o^{x_0}_{i_0} < o^1_{i_1} < \ldots < o^{x_1}_{i_1} < \ldots\ldots < o^1_{i_m} < \ldots < o^{x_m}_{i_m} = o_j \\ \beta \;:\; o_i = \bar{o}^1_{i_0} > \ldots > \bar{o}^{y_0}_{i_0} > \bar{o}^1_{i_1} > \ldots > \bar{o}^{y_1}_{i_1} > \ldots\ldots > \bar{o}^1_{i_m} > \ldots > \bar{o}^{y_m}_{i_m} = o_j \end{array} \tag{4}$$

such that, for $0 \leq k \leq m$, both $x_k \geq 1$ and $y_k \geq 1$, and each step in both chains α and β is *directly implied* by some “local” relation $>_l$ or $>_{l,l'}$ constructing $>$. The corresponding unique undirected path between i and j is:

$$i = i_0 \cdot i_1 \cdot \ldots \cdot i_{m-1} \cdot i_m = j \tag{5}$$

Without loss of generality, we assume that the cycle in $>$ induced by α and β is *length-wise minimal* among all such cycles in $>$. In particular, this implies that

(i) for $0 \leq k \leq m$, we have $1 \leq x_k, y_k \leq 2$,

(ii) for each pair of objects $o \in \alpha$, $o' \in \beta$, we have $o \neq o'$, unless $o = o' = o_i$ or $o = o' = o_j$, and

(iii) for each pair of objects $o \in \alpha$, $o' \in \beta$, no $>_l$ (and no $>_{l,l'}$) implies $o' \succ_l o$ (respectively, $o' \succ_{l,l'} o$).

First, let us show that at least one of the chains α and β contains at least one internal element. Assume, to the contrary, that both α and β contain no internal elements. If $i = j$, then we have $o_i >_i o'_i$ (where $o'_i = o_j$) and $o'_i >_i o_i$, contradicting our assumption that $>_i$ is a partial order. (If $>_i$ is not a partial order, then so is each $>_i \cup >_{i,j}$.) Otherwise, if $i \neq j$, then either $i \in \mathsf{pred}(j)$ or $j \in \mathsf{pred}(i)$. Assuming the latter, $(o_i > o_j) \wedge (o_i > o_j)$ implies $(o_i >_{i,j} o_j) \wedge (o_i >_{i,j} o_j)$, contradicting our assumption that $>_{i,j}$ is a partial order.

Given that, let us now prove that $o_{i_m}^{x_m} \neq \bar{o}_{i_m}^{y_m}$, contradicting the assumption that the chains α and β as in Eq. 4 exist. We do it on a case-by-case basis of possible combinations of x_m, y_m, and length-minimality of the cycle $\alpha\beta$ implies that there are only four such cases to consider.

[$x_m = 2, y_m = 2$] In this case, Eq. 4 implies $\bar{o}_{i_m}^1 >_{i_m} \bar{o}_{i_m}^2 = o_{i_m}^2 >_{i_m} o_{i_m}^1$. The transitivity of $>_{i_m}$ then implies $\bar{o}_{i_m}^1 > o_{i_m}^1$, contradicting our assumption of minimality of the cycle $\alpha\beta$.

[$x_m = 1, y_m = 1$] From Eq. 5 we have either $i_{m-1} \in \mathsf{pred}(i_m)$ or $i_m \in \mathsf{pred}(i_{m-1})$. If $i_{m-1} \in \mathsf{pred}(i_m)$, then Eq. 4 implies $\bar{o}_{i_{m-1}}^{y_{m-1}} >_{i_m,i_{m-1}} \bar{o}_{i_m}^1 = o_{i_m}^1 >_{i_m,i_{m-1}} o_{i_{m-1}}^{x_{m-1}}$. The transitivity of $>_{i_m,i_{m-1}}$ then implies $\bar{o}_{i_{m-1}}^{y_{m-1}} > o_{i_{m-1}}^{x_{m-1}}$, contradicting our assumption of minimality of the cycle $\alpha\beta$. Otherwise, if $i_m \in \mathsf{pred}(i_{m-1})$, then Eq. 4 implies $\bar{o}_{i_{m-1}}^{y_{m-1}} >_{i_{m-1},i_m} \bar{o}_{i_m}^1 = o_{i_m}^1 >_{i_{m-1},i_m} o_{i_{m-1}}^{x_{m-1}}$. Again, the transitivity of $>_{i_{m-1},i_m}$ then implies $\bar{o}_{i_{m-1}}^{y_{m-1}} > o_{i_{m-1}}^{x_{m-1}}$, contradicting our assumption of minimality of the cycle $\alpha\beta$.

[$x_m = 2, y_m = 1$] Here as well, Eq. 5 implies that we have either $i_{m-1} \in \mathsf{pred}(i_m)$ or $i_m \in \mathsf{pred}(i_{m-1})$. If $i_{m-1} \in \mathsf{pred}(i_m)$, then Eq. 4 implies $\bar{o}_{i_{m-1}}^{y_{m-1}} >_{i_m,i_{m-1}} \bar{o}_{i_m}^1 = o_{i_m}^2 >_{i_m} o_{i_m}^1$. Then, the transitivity of $>_{i_m} \cup >_{i_m,i_{m-1}}$ implies $\bar{o}_{i_{m-1}}^{y_{m-1}} > o_{i_m}^1$, contradicting our assumption of minimality of the cycle $\alpha\beta$. Otherwise, if $i_m \in \mathsf{pred}(i_{m-1})$, then Eq. 4 implies $\bar{o}_{i_{m-1}}^{y_{m-1}} >_{i_{m-1},i_m} \bar{o}_{i_m}^1 = o_{i_m}^2 >_{i_m} o_{i_m}^1$. Then, the transitivity of $>_{i_m} \cup >_{i_{m-1},i_m}$ implies $\bar{o}_{i_{m-1}}^{y_{m-1}} > o_{i_m}^1$, contradicting our assumption of minimality of the cycle $\alpha\beta$.

[$x_m = 1, y_m = 2$] This case is similar to the previous case of $x_m = 2, y_m = 1$, *mutatis mutandis.*

■

Lemma 2 *Let* Π *be a planning problem in* $\mathbf{P}_b$, $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ *be the corresponding constraint optimization problem, and* $\overline{x} \in Dom(\mathcal{X})$ *be an optimal solution for* COP_Π *with* $\sum_{\varphi\in\mathcal{F}} \varphi(\overline{x}) = \alpha$.

(I) If $\alpha < \infty$*, then a plan of cost* α *for* Π *can be reconstructed from* $\overline{x}$ *in time polynomial in the description size of* Π.

(II) If Π *has a plan, then* $\alpha < \infty$.

Proof:
(I) Given a COP solution $\overline{x} = \{\tau_{v_1}, \dots, \tau_{v_n}\}$ with $\sum_{\varphi\in\mathcal{F}} \varphi(\overline{x}) = \alpha < \infty$, we construct a plan ρ for Π with $\mathcal{C}(\rho) = \alpha$.

First, for each variable $v \in V$ with $\mathsf{pred}(v) = \emptyset$, let a sequence ρ_v of actions from A_v be defined as

$$\rho_v = \begin{cases} \emptyset & |\tau_v| = 1 \\ a_v^1 \cdot \ldots \cdot a_v^{|\tau_v|-1} & \text{otherwise} \end{cases}, \tag{6}$$

where, for $1 \leq j \leq |\tau_v| - 1$,

$$a_v^j = \begin{cases} a_{\mathsf{b}_v}, & j \text{ is even} \\ a_{\mathsf{w}_v}, & j \text{ is odd} \end{cases}, \tag{7}$$

with $\mathsf{eff}(a_{\mathsf{b}_v}) = \{\mathsf{b}_v\}$, and $\mathsf{eff}(a_{\mathsf{w}_v}) = \{\mathsf{w}_v\}$. From Eq. 3 and $\varphi_v(\tau_v) < \infty$, we immediately have (i) $\{a_{\mathsf{w}_v}\} \subseteq A_v$ if $|\tau_v| \geq 2$, and $\{a_{\mathsf{b}_v}, a_{\mathsf{w}_v}\} \subseteq A_v$ if $|\tau_v| > 2$, and (ii) $\mathcal{C}(\rho_v) = \varphi_v(\tau_v)$. Now, for a purpose that gets clear below, let a binary relation $>_v$ over the action elements of ρ_v be defined as the transitive closure of $\{a_v^{j-1} < a_v^j \mid 1 < j \leq |\tau_v| - 1\}$. Clearly, $>_v$ constitutes a strict total ordering over the elements of ρ_v.

Next, for each non-root variable $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, we construct the graph $G'_e(v)$ with respect to $\tau_{w_1}, \ldots, \tau_{w_k}$, and determine in $G'_e(v)$ a minimal-cost path of $|\tau_v| - 1$ edges from the source node $\langle \mathsf{b}_{w_1}^1 \cdots \mathsf{b}_{w_k}^1 \rangle$. The existence of such a path is implied by $\varphi_v(\tau_v, \tau_{w_1}, \ldots, \tau_{w_k}) < \infty$. By the construction of $G'_e(v)$ we also know that, for $1 \leq j \leq |\tau_v| - 1$, the j-th edge on this path is from a node labeled with $\langle \tau_{w_1}[l_1^{j-1}] \cdots \tau_{w_k}[l_k^{j-1}] \rangle$ to a node labeled with $\langle \tau_{w_1}[l_1^j] \cdots \tau_{w_k}[l_k^j] \rangle$, where for $1 \leq l \leq k$, we have $l_i^0 = 1$ and $l_i^{j-1} \leq l_i^j$. Having that, let a sequence ρ_v of actions from A_v be defined as in Eq. 6, with, for $1 \leq j \leq |\tau_v| - 1$,

$$\begin{aligned} \mathsf{eff}(a_v^j) &= \{\tau_v[j+1]\} \\ \mathsf{pre}(a_v^j) &= \left\{\tau_v[j], \tau_{w_1}[l_1^j], \tau_{w_2}[l_2^j], \ldots, \tau_{w_k}[l_k^j]\right\} \end{aligned} \tag{8}$$

Note that $\{a_v^1, \ldots, a_v^{|\tau_v|-1}\} \subseteq A_v$ is implied by the construction of $G'_e(v)$ and the presence of the considered minimal-cost path in it.

Now, similarly to the case of the root variables, let a binary relation $>_v$ over the action elements of ρ_v be defined as the transitive closure of $\{a_v^{j-1} < a_v^j \mid 1 < j \leq |\tau_v| - 1\}$. Here as well, $>_v$ constitutes a strict total ordering over the elements of ρ_v. In addition, for each parent w_i of v, let a binary relation $>_{v,w_i}$ over the union of the action elements of ρ_v and ρ_{w_i} be defined as the transitive closure of $>_{v,w_l}$, which in turn is defined as

$$\begin{aligned} >_{v,w_i} &= >_{v,w_i}^- \cup >_{v,w_i}^+ \\ >_{v,w_i}^- &= \left\{a_{w_i}^{l_i^j - 1} < a_v^j \mid 1 \leq j \leq |\tau_v| - 1, l_i^j > 1\right\} \\ >_{v,w_i}^+ &= \left\{a_v^j < a_{w_i}^{l_i^j} \mid 1 \leq j \leq |\tau_v| - 1, l_i^j < |\tau_{w_i}|\right\}. \end{aligned} \tag{9}$$

It is not hard to verify from Eq. 9 that, for each $v \in V$ and each $w \in \mathsf{pred}(v)$, not only $>_{v,w}$ constitutes a strict partial ordering, but so are the transitively closed $>_v \cup >_{v,w}$ and $>_w \cup >_{v,w}$. Given that,

- By the definition of $\rho_w = \langle a_w^1 \cdot \ldots \cdot a_w^l \rangle$, and the polytree structure of the causal graph $CG(\Pi)$, restricting the preconditions and effects of each a_w^i to the variables $\{v\} \cup \mathsf{pred}(v)$, we have $\mathsf{pre}(a_w^i) = \{\mathsf{b}_w\}$, $\mathsf{eff}(a_w^i) = \{\mathsf{w}_w\}$ for i being odd, and $\mathsf{pre}(a_w^i) = \{\mathsf{w}_w\}$, $\mathsf{eff}(a_w^i) = \{\mathsf{b}_w\}$ for i being even. For each $1 \leq i \leq k$, from Eq. 8 we have $\mathsf{eff}(a_{w_i}^{l_i^j - 1}) \in \mathsf{pre}(a_v^j)$. From Eq. 9 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, the construction of the graph $G'_e(v)$ implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

- The polytree structure of the causal graph $CG(\Pi)$ and Theorem 1 together imply that the transitively closed relation

$$> = \bigcup_{v \in V} (>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w})$$

is a strict partial order over the union of the action elements of $\rho_{v_1}, \ldots, \rho_{v_n}$.

Putting thing together, the above implies that any linearization of $>$ constitutes a valid plan ρ for Π with cost

$$\mathcal{C}(\rho) = \sum_{v \in V} \mathcal{C}(\rho_v) = \sum_{v \in V} \varphi_v(\overline{x}),$$

which is exactly what we had to prove. Here we also note that the plan extraction step of the algorithm polytree-k-indegree corresponds exactly to the above construction along Eqs. 6-9, providing us in poly-time with concrete cost-optimal plan corresponding to the optimal solution for COP_Π.

(II) We now prove that if Π is solvable, then we must have $\alpha < \infty$. Assume to the contrary that this is not the case. Let Π be a solvable planning problem, and let ρ be an irreducible plan for Π. Given such ρ, let $\overline{x}_\rho = \{\tau_{v_1}, \ldots, \tau_{v_n}\}$ be an COP assignment with each $|\tau_{v_i}| = |\rho{\downarrow_{v_i}}| - 1$. Note that $\overline{x}_\rho$ is well-defined (that is, for $1 \leq i \leq n$, we have $\tau_{v_i} \in \trianglerighteq^*[\tau(v_i)]$) by the definition of $\tau(v_i)$, Corollary 1, and ρ being irreducible. Let us now show that $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}_\rho) \leq \mathcal{C}(\rho)$, contradicting our assumption that $\alpha = \infty$ due to $\alpha \leq \sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}_\rho)$ and $\mathcal{C}(\rho) < \infty$.

First, for each variable v with $\mathsf{pred}(v) = \emptyset$, Eq. 3 immediately implies $\varphi_v(\overline{x}_\rho) \leq \mathcal{C}(\rho{\downarrow_v})$. Next, for each non-root variable $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, consider the graph $G'_e(v)$ constructed with respect to $\tau_{w_1}, \ldots, \tau_{w_k}$. Let $\{a_1, \ldots, a_{|\rho\downarrow_v|}\}$ be the actions of $\rho{\downarrow_v}$ numbered in the order of their appearance along it. Let $\{y_{w_1}(1), \ldots, y_{w_k}(1)\}$ denote the prevail condition of a_1 with each $y_{w_i}(1)$ being time-stamped with its earliest appearance along τ_{w_i}, that is $y_{w_i}(1) \in \{\mathsf{b}^1_{w_i}, \mathsf{w}^1_{w_i}\}$. Now, for $2 \leq j \leq |\rho{\downarrow_v}|$, we set $\{y_{w_1}(j), \ldots, y_{w_k}(j)\}$ to the prevail condition of a_i with each $y_{w_i}(j)$ being time-stamped with the lowest possible time index along τ_{w_i} satisfying "$y_{w_i}(j-1)$ does not come after $y_{w_i}(j)$ along τ_{w_i}". Given that

(i) $\rho{\downarrow_v}$ is a complete order-preserving restriction of ρ to the v-changing actions A_v,

(ii) the sequence of time-stamped prevail conditions $\{\{y_{w_1}(j), \ldots, y_{w_k}(j)\}\}_{j=1}^{|\rho\downarrow_v|}$ is constructed as above, and

(iii) $|\rho{\downarrow_v}| = |\tau_v| - 1$ by the construction of $\overline{x}_\rho$,

we have that $G'_e(v)$ contains a path

$$\langle \mathsf{b}^1_{w_1} \cdots \mathsf{b}^1_{w_k} \rangle \rightarrow \langle y_{w_1}(1) \cdots y_{w_k}(1) \rangle \rightarrow \cdots \rightarrow \langle y_{w_1}(|\rho{\downarrow_v}|) \cdots y_{w_k}(|\rho{\downarrow_v}|) \rangle$$

and the cost of this path is $\mathcal{C}(\rho{\downarrow_v}) < \infty$. However, from the constructive definition of φ_v in the algorithm polytree-k-indegree, we have $\varphi_v(\overline{x}_\rho)$ being the cost of the minimal-cost path of $|\tau_v| - 1$ edges in $G'_e(v)$ originated in $\langle \mathsf{b}^1_{w_1} \cdots \mathsf{b}^1_{w_k} \rangle$, and thus $\varphi_v(\overline{x}_\rho) \leq \mathcal{C}(\rho{\downarrow_v})$. Overall, the latter argument is valid for all planning variables $v \in V$, we have

$$\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}_\rho) \quad \leq \quad \sum_{v \in V} \mathcal{C}(\rho{\downarrow_v}) \quad = \quad \mathcal{C}(\rho),$$

which is what we had to prove. ■

Theorem 2 *Cost-optimal planning for* $\mathbf{P}_b$ *is tractable.*

Proof: The correctness of the $\mathsf{polytree\text{-}k\text{-}indegree}$ algorithm is given by Lemma 2. We now show that, given a planning problem $\Pi \in \mathbf{P}_b$, the corresponding constraint optimization problem COP_Π can be constructed and solved in time polynomial in the description size of Π.

Let n be the number of state variables in Π, and κ be the maximal node in-degree in the causal graph $CG(\Pi)$. In $\mathsf{polytree\text{-}k\text{-}indegree}$, for each planning variable $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, and each k-tuple $\tau'_1 \in \trianglerighteq^*[\tau(w_1)], \ldots, \tau'_k \in \trianglerighteq^*[\tau(w_k)]$, we

(i) construct the graph $G'_e(v)$, and

(ii) use the Dijkstra algorithm to compute shortest paths from the source node of $G'_e(v)$ to all other nodes in that graph.

For each w_i, we have $\tau(w_i) = n$, and thus the number of k-tuples as above for each $v \in V$ is $O(n^k)$. For each such k-tuple, the corresponding graph $G'_e(v)$ can be constructed in time linear in the number of its edges $= O(n^{2k+2} \cdot |A_v|^2) = O(n^{2k+2} \cdot 2^{2k+2})$ (Brafman & Domshlak, 2003). The time complexity of the Dijkstra algorithm on a digraph $G = (N, E)$ is $O(E \log(N))$, and on $G'_e(v)$ it gives us $O\left(n^{2k+2} \cdot 2^{2k+2} \cdot \log\left(n^{k+1} \cdot 2^{k+1}\right)\right)$. Putting things together, the complexity of constructing COP_Π is

$$O\left(n^{3\kappa+3} \cdot 2^{2\kappa+2} \cdot \log\left(n^{\kappa+1} \cdot 2^{\kappa+1}\right)\right). \tag{10}$$

Applying a tree-decomposition of COP_Π that clusters the scopes of its functional components we arrive into an equivalent, tree-structured constraint optimization problem over n variables with domains of size $O(n^{\kappa+1})$. This COP is defined by the hard binary "compatibility" constraints between the variables, and costs associated with the variables' values. Such a tree-structured COP can be solved in time $O(xy^2)$ where x is the number of variables and y is an upper bound on the size of a variable's domain (Dechter, 2003). Therefore, solving our COP_Π can be done in time $O(n^{2\kappa+3})$. As the expression in Eq. 10 dominates both $O(n^{2\kappa+3})$, and the time complexity of extracting a plan from the optimal solution to COP_Π (see the proof of (I) in Lemma 2), the overall complexity of the algorithm $\mathsf{polytree\text{-}k\text{-}indegree}$ is given by Eq. 10. And since in $\mathbf{P}_b$ we have $\kappa = O(1)$, we conclude that the complexity of $\mathsf{polytree\text{-}k\text{-}indegree}$ is polynomial in the description size of Π.

■

procedure $\underline{\textsf{tree-uniform-cost}}(\Pi = (V, A, I, G))$
takes a problem $\Pi \in \mathbf{T}$ with uniform-cost actions A
returns a cost-optimal plan for Π if Π is solvable, and fails otherwise
$\rho = \langle\rangle$, $s := I$, $V' := V$
loop
$V' := \textsf{remove-solved-leafs}(s, V')$
if $V' = \emptyset$ **return** ρ
else
find $v \in V', a \in A_v$ such that $a \in A(s)$ and
$\forall u \in \textsf{Desc}(v, V') :\ A_u \cap A(s) = \emptyset$
if not found **return** failure
$\rho := \rho \cdot \langle a \rangle$, $s := (s \setminus \textsf{pre}(a)) \cup \textsf{eff}(a)$

Figure 8: A simple algorithm for cost-optimal planning for $\mathbf{T}$ problems with uniform-cost actions. The notation $\textsf{Desc}(v, V')$ stands for the subset of V' containing the descendants of v in $CG(\Pi)$, and $A(s)$ stands for the set of all actions applicable in the state s.

3.3 Towards Practically Efficient Special Cases

The polytree-k-indegree algorithm for $\mathbf{P}_b$ is polynomial, but is rather involved and its complexity is exponential in $poly(\textsf{ln}(CG(\Pi)))$. It is quite possible that more efficient algorithms for $\mathbf{P}_b$, or, definitely, for some of its fragments can be devised. Here we show that, for instance, a simple algorithm for $\mathbf{T} \subset \mathbf{P}_b$ problems has already appeared in the literature in a different context, but it was never checked to when (if at all) it provides cost-optimal solutions. This is the TreeDT algorithm for preferential reasoning with tree-structured CP-nets (Boutilier, Brafman, Domshlak, Hoos, & Poole, 2004), and it turns out that its straightforward adaptation for $\mathbf{T}$ planning always provides cost-optimal solutions for $\mathbf{T}$ problems with *uniform-cost actions*. The algorithm is depicted in Figure 8, and it is not hard to verify that its time complexity is linear in the length of the generated plan ρ—all it does is iteratively "removing" the parts of the problem that can be safely ignored in the later steps, and applying a value-changing action on a lowest (in the causal graph) variable for which such an action exists.

Theorem 3 *Given a* $\mathbf{T}$ *problem* Π *with uniform-cost actions over* n *state variables,*

(I) if the algorithm tree-uniform-cost *finds a plan for* Π*, then this plan is cost-optimal,*

(II) the algorithm tree-uniform-cost *finds a plan if and only if* Π *is solvable, and*

(III) the time complexity of tree-uniform-cost *is* $\Theta(n^2)$.

Proof:

Without loss of generality, in what follows we assume the actions of Π are all unit-cost, that is, for each plan ρ for Π, $\mathcal{C}(\rho) = |\rho|$.

(I) Assume to the contrary that the plan ρ provided by tree-uniform-cost is not optimal, that is, there exists a plan ρ' such that $|\rho'| < |\rho|$. In particular, this implies existence of a variable v such that $|\rho'{\downarrow}_v| < |\rho{\downarrow}_v|$. The semantics of planning implies that

$$|\rho'{\downarrow}_v| \leq |\rho{\downarrow}_v| - (\epsilon_v + 1) \tag{11}$$

where $\epsilon_v = 1$ if $G[v]$ is specified, and 0 otherwise. Likewise, since the causal graph $CG(\Pi)$ forms a directed tree, there exists a variable v satisfying Eq. 11 such that, for all the descendants u of v in $CG(\Pi)$ holds:

$$|\rho'{\downarrow}_u| \geq |\rho{\downarrow}_u| \tag{12}$$

Let $Ch(v)$ be the set of all the immediate descendants of v in $CG(\Pi)$. By the construction of tree-uniform-cost, we have that:

1. If $Ch(v) = \emptyset$, then $|\rho{\downarrow}_v| \leq \epsilon_v$, and this contradicts Eq. 11 as $|\rho'{\downarrow}_v|$ is a non-negative quantity by definition.

2. Otherwise, if $Ch(v) \neq \emptyset$, then, by the construction of tree-uniform-cost, there exists $u \in Ch(v)$ such that changing its value $|\rho{\downarrow}_u|$ times *requires* changing the value of v at least $|\rho{\downarrow}_v| - \epsilon_v$ times. In other words, there is no action sequence ϱ applicable in I such that $|\varrho{\downarrow}_u| \geq |\rho{\downarrow}_u|$ while $|\varrho{\downarrow}_v| < |\rho{\downarrow}_v| - \epsilon_v$. However, from Eq. 12 we have $|\rho'{\downarrow}_u| \geq |\rho{\downarrow}_u|$, and thus $|\rho'{\downarrow}_v|$ has to be at least $|\rho{\downarrow}_v| - \epsilon_v$. This, however, contradicts Eq. 11.

Hence, we proved that $|\rho'{\downarrow}_v| \geq |\rho{\downarrow}_v|$, contradicting our assumption that $|\rho'| < |\rho|$.

(II) Straightforward by reusing *as is* the proof of Theorem 11 in (Boutilier et al., 2004).

(III) Implied by Theorems 12 and 13 in (Boutilier et al., 2004). ∎

The requirement in Theorem 3 for all actions to have the same cost is essential. The example below shows that, in general case, the algorithm tree-uniform-cost is no longer cost-optimal. Consider $\Pi = (V, A, I, G) \in \mathbf{T}$ with $V = \{v, u\}$, $I = \{\mathsf{b}_v, \mathsf{b}_u\}$, $G = \{\mathsf{b}_v, \mathsf{w}_u\}$, and $A = \{a_1, a_2, a_3, a_4\}$ with

$$\begin{aligned}
\mathsf{eff}(a_1) &= \{\mathsf{w}_v\}, \mathsf{pre}(a_1) = \{\mathsf{b}_v\} \\
\mathsf{eff}(a_2) &= \{\mathsf{b}_v\},\ \mathsf{pre}(a_2) = \{\mathsf{w}_v\} \\
\mathsf{eff}(a_3) &= \{\mathsf{w}_u\}, \mathsf{pre}(a_3) = \{\mathsf{b}_u, \mathsf{w}_v\} \\
\mathsf{eff}(a_4) &= \{\mathsf{w}_u\}, \mathsf{pre}(a_4) = \{\mathsf{b}_u, \mathsf{b}_v\} \\
\mathcal{C}(a_1) &= \mathcal{C}(a_2) = \mathcal{C}(a_3) = 1 \\
\mathcal{C}(a_4) &= 4
\end{aligned}$$

On this problem, the tree-uniform-cost algorithm returns $\rho = \langle a_4 \rangle$ with $\mathcal{C}(\rho) = 4$, while the optimal plan is $\rho' = \langle a_1, a_3, a_2 \rangle$ with $\mathcal{C}(\rho') = 3$.

4. Cost-Optimal Planning for $\mathbf{P}(1)$ with Uniform-Cost Actions

In this section we provide a polynomial time algorithm for cost-optimal planning for $\mathbf{P}(1)$ problems with uniform-cost actions. We begin with showing that such problems exhibit an interesting property, then we exploit this property for devising a planning-to-COP scheme for these problems, and then prove the correctness and complexity of the algorithm.

Here as well, we begin with providing some notation. Given a $\mathbf{P}(1)$ problem $\Pi = (V, A, I, G)$, for each $v \in V$, each $w \in \mathsf{pred}(v)$, and each $\alpha \in \{\mathsf{b}_v, \mathsf{w}_v\}$, $\beta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, by $a_{\alpha|\beta}$ we denote the action a with $\mathsf{eff}(a)[v] = \alpha$ and $\mathsf{pre}(a)[w] = \beta$. Since Π is 1-dependent, this implies that applicability of a is prevailed only by the value of w. Note that $a_{\alpha|\beta}$ is only a notation, and thus the action $a_{\alpha|\beta}$ may *not* belong to the action set A of Π.

4.1 Post-Unique Plans and $\mathbf{P}(1)$ Problems

We now proceed with introducing the notion of post-unique action sequences that plays a key role in our planning-to-COP compilation here.

Definition 4 *Let $\Pi = (V, A, I, G)$ be a* UB *problem instance. An action sequence $\varrho \in A^*$ is called* **post-unique** *if, for each pair of actions $a, a' \in \varrho$, we have $\mathsf{eff}(a) = \mathsf{eff}(a')$ only if $a = a'$. That is, all the changes of each variable to a certain value along ϱ are performed by the same (type of) action. The (possibly empty) set of all* **post-unique plans** *for Π is denoted by $\mathcal{P}^{pu}(\Pi)$ (or simply $\mathcal{P}^{pu}$, if the identity of Π is clear from the context).*

The notion of post-unique action sequences is closely related to the notion of post-unique planning problems (Bäckström & Klein, 1991; Bäckström & Nebel, 1995), but is considerably weaker than the latter. While action sets of post-unique planning problems are not allowed to contain two actions with the same effect, Definition 4 poses a similar restriction only on action sequences, and not on the underlying planning problems. Still, the property of post-uniqueness for plans is strong. In general, solvable problems in UB may not exhibit post-unique plans at all. Turns out, however, that for the problems in $\mathbf{P}(1)$ this is very much not the case.

Theorem 4 *For every solvable $\mathbf{P}(1)$ problem $\Pi = (V, A, I, G)$, we have $\mathcal{P}^{pu}(\Pi) \neq \emptyset$. Moreover, if the actions A are uniform-cost, then $\mathcal{P}^{pu}(\Pi)$ contains at least one cost-optimal plan.*

Proof: As the correctness of second claim immediately implies the correctness of the first one, here we focus on the proof the second claim. Given a $\mathbf{P}(1)$ problem $\Pi = (V, A, I, G)$ with uniform-cost actions, and plan $\rho = \langle a_1, \ldots, a_m \rangle$ for Π, we construct a sequence of actions ρ^* such that:

- ρ^* is a post-unique plan for Π,
- $\mathcal{C}(\rho^*) = \mathcal{C}(\rho)$.

This construction is two-step. First, for each $v \in V$, the map the subsequence $\rho\!\downarrow_v = \langle a_{i_1}, \ldots, a_{i_k} \rangle$ into a post-unique sequence of actions $\rho^*_v = \langle a^*_{i_1}, \ldots, a^*_{i_k} \rangle$. Note that the indexes $i_1, \ldots, i_k$ of the action elements of each $\rho\!\downarrow_v$ are the *global* indexes of these actions along ρ, and the *exactly the same* indexes are used for marking the elements of the constructed sequences ρ^*_v. Having constructed the sequences $\rho^*_{v_1}, \ldots, \rho^*_{v_n}$, we merge them into a single actions sequence ρ^*, and show that ρ^* is a valid plan for Π. Note that the two properties of ρ^* as required above will then hold immediately because $|\rho^*| = |\rho|$, and post-uniqueness of ρ^* is implied by individual post-uniqueness of all its per-variable components ρ^*_v.

The mapping of subsequences $\rho{\downarrow_v}$ of ρ to the desired sequences ρ^*_v for all variables v is performed top-down, consistently with a topological ordering of the causal graph $CG(\Pi)$. This top-down processing allows us to assume that, when constructing ρ^*_v, the subsequences ρ^*_w for all $w \in \mathsf{pred}(v)$ are already constructed. Given that, while mapping each $\rho{\downarrow_v} = \langle a_{i_1}, \ldots, a_{i_k}\rangle$ to the corresponding ρ^*_v, we distinguish between the following three cases.

(1) *The subsequence $\rho{\downarrow_v}$ is already post-unique.*

In this case, we simply set ρ^*_v to $\rho{\downarrow_v}$. In addition, we construct the following sets of ordering constraints. First, we set a binary relation $>_v$ over the action elements of $\rho^*_v = \langle a^*_{i_1}, \ldots, a^*_{i_k}\rangle$ to

$$>_v = \{a^*_i < a^*_j \mid a^*_i, a^*_j \in \rho^*_v, i < j\}. \tag{13}$$

It is immediate from Eq. 13 that $>_v$ is a strict total order over the elements of ρ^*_v as $>_v$ simply follows the action indexing inherited by ρ^*_v from plan ρ via $\rho{\downarrow_v}$.

Now, for each $w \in \mathsf{pred}(v)$, we set a binary relation $>_{v,w}$ over the elements of ρ^*_v and ρ^*_w to

$$>_{v,w} = \begin{cases} \bigcup_{a^*_i \in \rho^*_v, a^*_j \in \rho^*_w} \{a^*_i < a^*_j \mid i < j\} \cup \{a^*_j < a^*_i \mid j < i\}, & \mathsf{pre}(a)[w] \text{ is specified for some } a \in \rho^*_v \\ \emptyset, & \text{otherwise} \end{cases}. \tag{14}$$

For each $w \in \mathsf{pred}(v)$, the relation $>_{v,w}$ defined by Eq. 14 is a strict total order over its domain because the ordering constraints between the elements of ρ^*_v and ρ^*_w are a *subset* of the constraints induced by the total-order plan ρ over the (corresponding) actions from $\rho{\downarrow_v}$ and $\rho{\downarrow_w}$. For the same reason, from Eqs. 13 and 14, we have that, for each $w \in \mathsf{pred}(v)$, $>_v \cup >_{v,w}$ is a strict total order over the union of the elements of ρ^*_v and ρ^*_w.

From Eqs. 13-14 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(2) *The subsequence $\rho{\downarrow_v}$ is not post-unique, but the actions in $\rho{\downarrow_v}$ are all prevailed by the value of a single parent $w \in \mathsf{pred}(v)$.*

Since $\rho{\downarrow_v}$ is *not* post-unique, $\rho{\downarrow_v}$ in this case has to contain instances of at least three action types from $\{a_{\mathsf{b}_v|\mathsf{b}_w}, a_{\mathsf{b}_v|\mathsf{w}_w}, a_{\mathsf{w}_v|\mathsf{b}_w}, a_{\mathsf{w}_v|\mathsf{w}_w}\}$. Thus, in particular, it must be that

(a) $|\rho{\downarrow_w}| \geq 1$, and

(b) for some $\beta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, we have $a_{\mathsf{w}_v|\beta}, a_{\mathsf{b}_v|\beta} \in \rho{\downarrow_v}$.

Given that, we set $\rho^*_v = \langle a^*_{i_1}, \ldots, a^*_{i_k}\rangle$ to

$$\forall 1 \leq j \leq k: \quad a^*_{i_j} = \begin{cases} a_{\mathsf{w}_v|\beta}, & j \text{ is odd} \\ a_{\mathsf{b}_v|\beta}, & j \text{ is even} \end{cases}.$$

Both post-uniqueness of such ρ_v^*, as well as its applicability with respect to v are straightforward. The ordering constraints $>_v$ are then set according to Eq. 13. Likewise, if $\rho_w^* = \langle a_{j_1}, \ldots, a_{j_l} \rangle$, we set

$$>_{v,w}= \begin{cases} \bigcup_{a_i^* \in \rho_v^*} \{a_i^* < a_{j_1}\}, & \beta = \mathsf{b}_w \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_{j_1}\}, & \beta = \mathsf{w}_w, l = 1 \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_{j_1}\} \cup \{a_i^* < a_{j_2}\}, & \beta = \mathsf{w}_w, l > 1 \end{cases} \tag{15}$$

Finally, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{w\}$ are set to empty sets.

The relation $>_v$ here is identical to this in case (1), and thus it is a strict total order over the elements of ρ_v^*. From Eq. 15, it is easy to verify that $>_{v,w}$ is also a strict partial order over the union of the elements of ρ_v^* and ρ_w^*. Finally, as all the elements of ρ_v^* are all identically constrained with respect to the elements of ρ_w^*, we have $>_v \cup >_{v,w}$ forming a strict partial order over the union of the elements of ρ_v^* and ρ_w^*. (For all other parents $w' \in \mathsf{pred}(v)$, we simply have $>_v \cup >_{v,w} = >_v$.)

From Eqs. 13, 15 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho\downarrow_v|$ implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(3) *The subsequence $\rho\downarrow_v$ is not post-unique, and the actions of $\rho\downarrow_v$ are prevailed by more than one parent of v.*

The setting of this case in particular implies that there is a pair of v's parents $\{u, w\} \subseteq \mathsf{pred}(v)$ such that $a_{\mathsf{w}_v|\alpha}, a_{\mathsf{b}_v|\beta} \in \rho_v$ for some $\alpha \in \{\mathsf{b}_u, \mathsf{w}_u\}$, $\beta \in \{\mathsf{b}_w, \mathsf{w}_w\}$. Given that, we set ρ_v^* to

$$\forall 1 \leq j \leq k : \quad a_{i_j}^* = \begin{cases} a_{\mathsf{w}_v|\alpha}, & j \text{ is odd} \\ a_{\mathsf{b}_v|\beta}, & j \text{ is even} \end{cases},$$

and, similarly to case (2), both post-uniqueness of ρ_v^*, and its applicability with respect to v are straightforward.

Here as well, the ordering constraints $>_v$ are set according to Eq. 13. Likewise, if $\rho_w^* = \langle a_{j_1}, \ldots, a_{j_l} \rangle$, and $\rho_u^* = \langle a_{j_1'}, \ldots, a_{j_{l'}'} \rangle$, we set $>_{v,w}$ according to Eq. 15 above, and $>_{v,u}$ according to Eq. 16 below.

$$>_{v,u} = \begin{cases} \bigcup_{a_i^* \in \rho_v^*} \{a_i^* < a_{j_1'}\}, & \alpha = \mathsf{b}_u \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_{j_1'}\}, & \alpha = \mathsf{w}_u, l' = 1 \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_{j_1'}\} \cup \{a_i^* < a_{j_2'}\}, & \alpha = \mathsf{w}_u, l' > 1 \end{cases} \tag{16}$$

Finally, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets.

The relation $>_v$ here is identical to this in cases (1-2), and relations $>_{v,u}$ and $>_{v,w}$ are effectively identical to the relation $>_{v,w}$ in case (2). Thus, we have $>_v \cup >_{v,u}$ and

$>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ^*_v and ρ^*_u, and ρ^*_v and ρ^*_w, respectively.

From Eqs. 13, 15, 16 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow}_v|$ implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

As the last step, we now prove that, for each $v \in V$ and each $w \in \mathsf{pred}(v)$, we have $>_w \cup >_{v,w}$ being a strict partial order over the union of the elements of ρ^*_w and ρ^*_v.

- If $>_{v,w}$ is constructed according to Eq. 14, then $>_w \cup >_{v,w}$ is a subset of the constraints induced by plan ρ over the (corresponding to ρ^*_v and ρ^*_w) actions from ρ_v and ρ_w.

- Otherwise, if $>_{v,w}$ is constructed according to Eq. 15 or (for us here identical) Eq. 16, then $>_{v,w}$ (i) addresses at most two elements of ρ_w, (ii) orders these elements consistently with $>_w$.

In both cases, the argued properties of $>_w \cup >_{v,w}$ implies that it forms a strict partial order over the union of the elements of ρ^*_v and ρ^*_w.

Until now, we have specified the sequences ρ^*_v, the orders $>_v$ induced by these sequences, the orders $>_{v,w}$, and proved that all $>_v \cup >_{v,w}$ and $>_w \cup >_{v,w}$ form strict partial orders over their domains. This construction allows us to apply now Theorem 1 to the (considered as sets) sequences ρ^*_v and orders $>_v$ and $>_{v,w}$, proving that

$$> = \bigcup_{v \in V} (>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w})$$

forms a strict partial order over the union of $\rho^*_{v_1}, \ldots, \rho^*_{v_n}$. Putting thing together, the above implies that any linearization ρ^* of $>$ is a plan for Π, and post-uniqueness of all its subsequences $\rho^*_{v_1}, \ldots, \rho^*_{v_n}$ then implies $\rho^* \in \mathcal{P}^{\mathrm{pu}}(\Pi)$. Moreover, if ρ is an optimal plan for Π, then $|\rho^*| = |\rho|$ implies the optimality of ρ^*. ∎

4.2 Construction

The main impact of Theorem 4 on our planning-to-COP scheme for uniform-cost $\mathbf{P}(1)$ is that we can now restrict our attention to post-unique plans only. Given that, the constraint optimization problem $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ for a uniform-cost problem $\Pi = (V, A, I, G) \in \mathbf{P}(1)$ is specified as follows.

The variable set $\mathcal{X}$ contains a variable x_v for each planning variable $v \in V$, and a variable x^w_v for each edge $(w, v) \in CG(\Pi)$. That is,

$$\begin{aligned} \mathcal{X} &= \mathcal{X}^V \cup \mathcal{X}^{\mathcal{E}}, \\ \mathcal{X}^V &= \{x_v \mid v \in V\} \\ \mathcal{X}^{\mathcal{E}} &= \{x^w_v \mid (w, v) \in CG(\Pi)\} \end{aligned} \tag{17}$$

For each variable $x_v \in \mathcal{X}^V$, the domain $Dom(x_v)$ consists of all goal-valid prefixes of $\sigma(v)$. For each variable $x^w_v \in \mathcal{X}^{\mathcal{E}}$, the domain $Dom(x^w_v)$ consists of all triples of integers

$[\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!]$ satisfying Eq. 18.

$$\begin{aligned} Dom(x_v) &= \trianglerighteq^*[\sigma(v)] \\ Dom(x_v^w) &= \{[\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!] \mid \delta_{\mathsf{w}}, \delta_{\mathsf{b}} \in \{0,1\}, 0 \leq \eta \leq n\} \end{aligned} \tag{18}$$

The semantics of Eq. 18 is as follows. Let $\{w_1, \ldots, w_k\}$ be an *arbitrary* fixed ordering of $\mathsf{pred}(v)$. If x_v takes the value $\sigma_v \in Dom(x_v)$, then v is forced to provide σ_v sequence of values. In turn, if $x_v^{w_i}$ takes the value $[\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!]$, then η corresponds to the number of value changes of v, $\delta_{\mathsf{w}} = 1$ ($\delta_{\mathsf{b}} = 1$) forces the *subset* of parents $\{w_1, \ldots, w_i\} \subseteq \mathsf{pred}(v)$ to support (that is, prevail) all the changes of v to w_v (respectively, to b_v), and $\delta_{\mathsf{w}} = 0$ ($\delta_{\mathsf{b}} = 0$) relieves this subset of parents $\{w_1, \ldots, w_i\}$ from this responsibility.

For each variable $x \in \mathcal{X}$, the set $\mathcal{F}$ contains a non-negative, real-valued function φ_x with the scope

$$Q_x = \begin{cases} \{x_v\}, & x = x_v, k = 0 \\ \{x_v, x_v^{w_k}\}, & x = x_v, k > 0 \\ \{x_v^{w_1}, x_{w_1}\}, & x = x_v^{w_1}, k > 0 \\ \{x_v^{w_j}, x_v^{w_{j-1}}, x_{w_j}\}, & x = x_v^{w_j}, 1 < j \leq k \end{cases} \tag{19}$$

where $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$ (and $k = 0$ means $\mathsf{pred}(v) = \emptyset$). Proceeding now with specifying these functional components $\mathcal{F}$ of COP_Π, first, for each x_v with $\mathsf{pred}(v) = \emptyset$, and for each $\sigma_v \in \trianglerighteq^*[\sigma(v)]$, we set $\varphi_{x_v}(\sigma_v)$ to

$$\varphi_{x_v}(\sigma_v) = \begin{cases} 0, & |\sigma_v| = 1, \\ 1, & (|\sigma_v| = 2) \wedge (a_{\mathsf{w}_v} \in A_v), \\ |\sigma_v| - 1, & (|\sigma_v| > 2) \wedge (a_{\mathsf{w}_v}, a_{\mathsf{b}_v} \in A_v), \\ \infty, & \text{otherwise} \end{cases} \tag{20}$$

In turn, for each planning variable $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, $k > 0$, the function φ_{x_v} is set to

$$\varphi_{x_v}(\sigma_v, [\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!]) = \begin{cases} 0, & (|\sigma_v| = 1) \wedge ([\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!] = [\![0,0,0]\!]), \\ 1, & (|\sigma_v| = 2) \wedge ([\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!] = [\![1,0,1]\!]), \\ |\sigma_v| - 1, & (|\sigma_v| > 2) \wedge ([\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!] = [\![1,1,|\sigma_v|-1]\!]), \\ \infty, & \text{otherwise} \end{cases} \tag{21}$$

The functions φ_{x_v} capture the, *marginal over the actions* A_v, cost of providing a sequence σ_v of value changes of v in Π, given that (in case of Eq. 21) the parents of v are "ready to support these value changes". In specifying the remaining functional components we use an "indicator" function φ specified in Eq. 22.

$$\varphi([\![\delta_{\mathsf{w}}, \delta_{\mathsf{b}}, \eta]\!], \sigma_w) = \begin{cases} 0, & \delta_{\mathsf{w}} = 0, \delta_{\mathsf{b}} = 0, \\ 0, & \delta_{\mathsf{w}} = 1, \delta_{\mathsf{b}} = 0, (a_{\mathsf{w}_v|\mathsf{b}_w} \in A_v) \vee ((|\sigma_w| > 1) \wedge (a_{\mathsf{w}_v|\mathsf{w}_w} \in A_v)), \\ 0, & \delta_{\mathsf{w}} = 0, \delta_{\mathsf{b}} = 1, (a_{\mathsf{b}_v|\mathsf{b}_w} \in A_v) \vee ((|\sigma_w| > 1) \wedge (a_{\mathsf{b}_v|\mathsf{w}_w} \in A_v)), \\ 0, & \delta_{\mathsf{w}} = 1, \delta_{\mathsf{b}} = 1, (a_{\mathsf{w}_v|\mathsf{b}_w}, a_{\mathsf{b}_v|\mathsf{b}_w} \in A_v) \vee ((|\sigma_w| > 1) \wedge (a_{\mathsf{w}_v|\mathsf{w}_w}, a_{\mathsf{b}_v|\mathsf{w}_w} \in A_v)), \\ 0, & \delta_{\mathsf{w}} = 1, \delta_{\mathsf{b}} = 1, |\sigma_w| \geq \eta, a_{\mathsf{w}_v|\mathsf{b}_w}, a_{\mathsf{b}_v|\mathsf{w}_w} \in A_v, \\ 0, & \delta_{\mathsf{w}} = 1, \delta_{\mathsf{b}} = 1, |\sigma_w| > \eta, a_{\mathsf{w}_v|\mathsf{w}_w}, a_{\mathsf{b}_v|\mathsf{b}_w} \in A_v, \\ \infty, & \text{otherwise} \end{cases} \tag{22}$$

The semantics of φ is that, for each planning variable $v \in V$, each $w \in \mathsf{pred}(v)$, and each $([\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!], \sigma_w) \in Dom(x_v^w) \times Dom(x_w)$, we have $\varphi([\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!], \sigma_w) = 0$ if the value sequence σ_w of w can support *all* the changes of v to w_v (if $\delta_\mathsf{w} = 1$) and *all* the changes of v to b_v (if $\delta_\mathsf{b} = 1$), out of η value changes of v in Π. Given this indicator function φ, for each $v \in V$, the functional component $\varphi_{x_v^{w_1}}$ is specified as

$$\varphi_{x_v^{w_1}}\left([\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!], \sigma_{w_1}\right) = \varphi\left([\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!], \sigma_{w_1}\right), \tag{23}$$

and the rest of the functions $\varphi_{x_v^{w_2}}, \ldots, \varphi_{x_v^{w_k}}$ are specified as follows. For each $2 \le i \le k$, the value of the function $\varphi_{x_v^{w_j}}$ at the combination of $[\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!] \in Dom(x_v^{w_i})$, $[\![\delta'_\mathsf{w}, \delta'_\mathsf{b}, \eta']\!] \in Dom(x_v^{w_{i-1}})$, and $\sigma_{w_i} \in Dom(x_{w_i}) = \trianglerighteq^*[\sigma(w_i)]$ is specified as

$$\varphi_{x_v^{w_i}}\left([\![\delta_\mathsf{w}, \delta_\mathsf{b}, \eta]\!], [\![\delta'_\mathsf{w}, \delta'_\mathsf{b}, \eta']\!], \sigma_{w_j}\right) = \begin{cases} \varphi\left([\![\delta_\mathsf{w} - \delta'_\mathsf{w}, \delta_\mathsf{b} - \delta'_\mathsf{b}, \eta]\!], \sigma_{w_j}\right), & \eta = \eta' \wedge \delta_\mathsf{w} \ge \delta'_\mathsf{w} \wedge \delta_\mathsf{b} \ge \delta'_\mathsf{b} \\ \infty & \text{otherwise} \end{cases} \tag{24}$$

This finalized the construction of COP_Π, and this construction constitutes the first three steps of the algorithm polytree-1-dep-uniform in Figure 9(a). The subsequent steps of this algorithm are conceptually similar to these of the polytree-k-indegree algorithm in Section 3, with the major difference being in the plan reconstruction routines. It is not hard to verify from Eqs. 17-19, and the fact that the causal graph of $\Pi \in \mathbf{P}(1)$ forms a polytree that

(i) for each variable $x \in \mathcal{X}$, $|Dom(x)| = poly(n)$,

(ii) the tree-width of the cost network of $\mathcal{F}$ is ≤ 3, and

(iii) the optimal tree-decomposition of the COP_Π's cost network is given by any topological ordering of the causal graph that is consistent with the (arbitrary yet fixed at the time of the COP_Π's construction) orderings of each planning variable's parents in the causal graph.

For an illustration, we refer the reader to Figure 3 (pp. 29) where Figure 3(a) depicts the causal graph of a problem $\Pi \in \mathbf{P}(1)$, and Figure 3(c) depicts the cost network of the corresponding COP_Π. The top-most variables and the cliques in the cost network correspond to the functional components of COP_Π.

4.3 Correctness and Complexity

Lemma 3 *Let Π be a $\mathbf{P}(1)$ problem with uniform-costs actions, $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ be the corresponding constraint optimization problem, and $\overline{x}$ be an optimal assignment to $\mathcal{X}$ with $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \alpha$.*

(I) If $\alpha < \infty$, then a plan of cost α for Π can be reconstructed from $\overline{x}$ in time polynomial in the description size of Π.

(II) If Π has a plan, then $\alpha < \infty$.

Proof:
(I) Given a COP solution $\overline{x}$ with $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \alpha < \infty$, we construct a plan ρ for Π with $\mathcal{C}(\rho) = \alpha$. We construct this plan by

procedure polytree-1-dep-uniform$(\Pi = (V, A, I, G))$
takes a problem $\Pi \in \mathbf{P}(1)$ with uniform-cost actions A
returns a cost-optimal plan for Π if Π is solvable, and fails otherwise
create a set of variables $\mathcal{X}$ as in Eqs. 17-18
create a set of functions $\mathcal{F} = \{\varphi_x \mid x \in \mathcal{X}\}$ with scopes as in Eq. 19
for each $x \in \mathcal{X}$ **do**
specify φ_x according to Eqs. 20-24
endfor
set $\mathsf{COP}_\Pi := (\mathcal{X}, \mathcal{F})$ with global objective $\min \sum_{\varphi\in\mathcal{F}} \varphi(\mathcal{X})$
$\overline{x} :=$ solve-tree-cop(COP_Π)
if $\sum_{\varphi\in\mathcal{F}} \varphi(\overline{x}) = \infty$ **then return** failure
extract plan ρ from $\overline{x}$ with $\mathcal{C}(\rho) = \sum_{\varphi\in\mathcal{F}} \varphi(\overline{x})$
return ρ

Figure 9: Algorithm for cost-optimal planning for $\mathbf{P}(1)$ problems with uniform-cost actions.

1. Traversing the planning variables in a topological ordering of the causal graph $CG(\Pi)$, and associating each variable v with a sequence $\rho_v \in A_v^*$.

2. Merging the constructed sequences $\rho_{v_1}, \dots, \rho_{v_n}$ into the desired plan ρ.

For each variable $x_v \in \mathcal{X}$, let σ_v denote the value provided by $\overline{x}$ to x_v. First, for each variable $v \in V$ with $\mathsf{pred}(v) = \emptyset$, let a sequence ρ_v of actions from A_v be defined as

$$\rho_v = \begin{cases} \emptyset & |\sigma_v| = 1 \\ a_v^1 \cdot \ldots \cdot a_v^{|\sigma_v|-1} & \text{otherwise} \end{cases}, \tag{25}$$

where, for $1 \le j \le |\sigma_v| - 1$,

$$a_v^j = \begin{cases} a_{\mathsf{b}_v}, & j \text{ is even} \\ a_{\mathsf{w}_v}, & j \text{ is odd} \end{cases}, \tag{26}$$

with $\mathsf{eff}(a_{\mathsf{b}_v}) = \{\mathsf{b}_v\}$, and $\mathsf{eff}(a_{\mathsf{w}_v}) = \{\mathsf{w}_v\}$. From Eq. 20 and $\varphi_v(\sigma_v) \le \alpha < \infty$, we immediately have (i) $\{a_{\mathsf{w}_v}\} \subseteq A_v$ if $|\sigma_v| \ge 2$, and $\{a_{\mathsf{b}_v}, a_{\mathsf{w}_v}\} \subseteq A_v$ if $|\sigma_v| > 2$, and (ii) $\mathcal{C}(\rho_v) = \varphi_v(\sigma_v)$. Let a binary relation $>_v$ over the action elements of ρ_v be defined as the transitive closure of $\{a_v^{j-1} < a_v^j \mid 1 < j \le |\sigma_v| - 1\}$, that is

$$>_v = \{a_v^{j'} < a_v^j \mid 1 \le j' < j \le |\sigma_v| - 1\} \tag{27}$$

Clearly, $>_v$ constitutes a strict total ordering over the elements of ρ_v, making ρ_v an applicable sequence of actions that provides to v the value $G[v]$ if the latter is specified.

Next, for each variable $v \in V$ with $\mathsf{pred}(v) \neq \emptyset$, let $\mathsf{pred}(v) = \{w_1, \dots, w_k\}$ be numbered according to their ordering used for constructing COP_Π. Likewise, for each $w_i \in \mathsf{pred}(v)$, let $[\![\delta_{\mathsf{w}}(i), \delta_{\mathsf{b}}(i), \eta]\!]$ be the value provided by $\overline{x}$ to $x_v^{w_i}$. Given that, let a pair of indexes

$0 \leq \langle \mathsf{w} \rangle, \langle \mathsf{b} \rangle \leq k$ be defined as

$$\langle \mathsf{w} \rangle = \begin{cases} 0, & \delta_{\mathsf{w}}(k) = 0, \\ 1, & \delta_{\mathsf{w}}(1) = 1, \\ j, & \delta_{\mathsf{w}}(j-1) < \delta_{\mathsf{w}}(j),\ 2 \leq j \leq k \end{cases} \tag{28}$$

$$\langle \mathsf{b} \rangle = \begin{cases} 0, & \delta_{\mathsf{b}}(k) = 0, \\ 1, & \delta_{\mathsf{b}}(1) = 1, \\ j, & \delta_{\mathsf{b}}(j-1) < \delta_{\mathsf{b}}(j),\ 2 \leq j \leq k \end{cases} \tag{29}$$

Informally, in our next-coming construction of the action sequence ρ_v for the state variable v, $\langle \mathsf{w} \rangle$ and $\langle \mathsf{b} \rangle$ will indicate the parents prevailing the value changes of v to w_v and to b_v, respectively, along ρ_v. Note that Eqs. 28-29 are well-defined because, for $2 \leq j \leq k$, Eq. 24 implies

$$\delta_{\mathsf{w}}(j-1) \leq \delta_{\mathsf{w}}(j) \quad \wedge \quad \delta_{\mathsf{b}}(j-1) \leq \delta_{\mathsf{b}}(j) \quad \wedge \quad \eta = \eta.$$

Given this notation, the action sequence ρ_v and the partial orders $>_{v,w_1}, \ldots, >_{v,w_k}$ are constructed as follows.

[$\langle \mathsf{w} \rangle = 0, \langle \mathsf{b} \rangle = 0$] In this case, the constructed plan ρ should perform no value changes of v, and thus ρ_v is set to an empty action sequence, and, consequently, both $>_v$ and all $>_{v,w}$ are set to empty sets.

[$\langle \mathsf{w} \rangle > 0, \langle \mathsf{b} \rangle = 0$] In this case, the constructed plan ρ should perform exactly one value change of v (from b_v to w_v), and thus ρ_v is set to contain exactly one action a_v^1 with $\mathsf{eff}(a) = \{\mathsf{w}_v\}$, and

$$\mathsf{pre}(a_v^1) = \begin{cases} \{\mathsf{b}_v, \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}\}, & a_{\mathsf{w}_v | \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}} \in A_v \\ \{\mathsf{b}_v, \mathsf{w}_{w_{\langle \mathsf{w} \rangle}}\}, & \text{otherwise} \end{cases} \tag{30}$$

Note that a_v^1 is well-defined, as $\alpha < \infty$ and Eq. 22 together imply that $\{a_{\mathsf{w}_v | \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, a_{\mathsf{w}_v | \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}\} \cap A_v \neq \emptyset$ (see case (2) in Eq. 22). In both outcomes of Eq. 30, we set $>_v = \emptyset$. If $a_v^1 = a_{\mathsf{w}_v | \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}$, we set

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \{a_v^1 < a^1_{w_{\langle \mathsf{w} \rangle}} \mid a^1_{w_{\langle \mathsf{w} \rangle}} \in \rho_{w_{\langle \mathsf{w} \rangle}}\} \tag{31}$$

Otherwise, if $a_v^1 = a_{\mathsf{w}_v | \mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}$, then from case (2) in Eq. 22, $a_{\mathsf{w}_v | \mathsf{b}_{w_{\langle \mathsf{w} \rangle}}} \notin A_v$, and $\alpha < \infty$, we have $|\sigma_{w_{\langle \mathsf{w} \rangle}}| > 1$, and thus $|\rho_{w_{\langle \mathsf{w} \rangle}}| \geq 1$. Given that, we set

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \{a^1_{w_{\langle \mathsf{w} \rangle}} < a_v^1\} \cup \{a_v^1 < a^2_{w_{\langle \mathsf{w} \rangle}} \mid a^2_{w_{\langle \mathsf{w} \rangle}} \in \rho_{w_{\langle \mathsf{w} \rangle}}\} \tag{32}$$

In both cases, it is easy to verify that $>_v \cup >_{v,w_{\langle \mathsf{w} \rangle}} \cup >_{w_{\langle \mathsf{w} \rangle}}$ constitutes a strict total order over the action elements of ρ_v and $\rho_{w_{\langle \mathsf{w} \rangle}}$. (In particular, this trivially implies that $>_v \cup >_{v,w}$ and $>_{v,w} \cup >_w$ are strict partial orderings over their domains.)

From Eqs. 27, 31, 32 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, Eq. 18 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

[$\langle \mathsf{w} \rangle > 0, \langle \mathsf{b} \rangle > 0, \langle \mathsf{w} \rangle = \langle \mathsf{b} \rangle$] In this case, the constructed plan ρ should perform more than one value change of v, and all these value changes should be performed by (a pair of types of) actions prevailed by the value of $w_{\langle \mathsf{w} \rangle}$. From $\alpha < \infty$, we have $\varphi([\![\delta_{\mathsf{w}}(\langle \mathsf{w} \rangle), \delta_{\mathsf{b}}(\langle \mathsf{w} \rangle), \eta]\!], \sigma_{w_{\langle \mathsf{w} \rangle}}) = 0$. The specification of the case in question (that is, $\langle \mathsf{w} \rangle = \langle \mathsf{b} \rangle > 0$) thus implies that one of the conditions of the cases (4-6) of Eq. 22 should hold. Given that, we distinguish between the following four settings.

(1) If $\{a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}\} \subseteq A_v$, then ρ_v is specified according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is even} \end{cases}. \tag{33}$$

The relation $>_v$ is set according to Eq. 27, and $>_{v,w_{\langle \mathsf{w} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \{a_v^i < a_{w_{\langle \mathsf{w} \rangle}}^1 \mid a_v^i \in \rho_v, a_{w_{\langle \mathsf{w} \rangle}}^1 \in \rho_{w_{\langle \mathsf{w} \rangle}}\} \tag{34}$$

Finally, for all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w} \rangle}\}$, we set $>_{v,w} = \emptyset$.

(2) Otherwise, if $\{a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}, a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}\} \subseteq A_v$ and $|\sigma_{w_{\langle \mathsf{w} \rangle}}| > 1$, then we have $|\rho_{w_{\langle \mathsf{w} \rangle}}| \geq 1$. Given that, we again set ρ_v according to Eq. 25, but now with its action elements being set as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is even} \end{cases}. \tag{35}$$

The relation $>_v$ is set according to Eq. 27, and $>_{v,w_{\langle \mathsf{w} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \{a_{w_{\langle \mathsf{w} \rangle}}^1 < a_v^i \mid a_v^i \in \rho_v\} \cup \{a_v^i < a_{w_{\langle \mathsf{w} \rangle}}^2 \mid a_v^i \in \rho_v, a_{w_{\langle \mathsf{w} \rangle}}^2 \in \rho_{w_{\langle \mathsf{w} \rangle}}\} \tag{36}$$

Finally, here as well, for all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w} \rangle}\}$, we set $>_{v,w} = \emptyset$.

(3) Otherwise, if $\{a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}\} \subseteq A_v$, and $|\sigma_{w_{\langle \mathsf{w} \rangle}}| \geq |\sigma_v| - 1$, then ρ_v is specified according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is even} \end{cases}. \tag{37}$$

The relation $>_v$ is set according to Eq. 27, and $>_{v,w_{\langle \mathsf{w} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \bigcup_{a_v^i \in \rho_v, a_{w_{\langle \mathsf{w} \rangle}}^j \in \rho_{w_{\langle \mathsf{w} \rangle}}} \{a_v^i < a_{w_{\langle \mathsf{w} \rangle}}^j \mid i \leq j\} \cup \{a_{w_{\langle \mathsf{w} \rangle}}^j < a_v^i \mid i > j\} \tag{38}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w} \rangle}\}$, we set $>_{v,w} = \emptyset$.

(4) Otherwise, if $\{a_{\mathsf{w}_v|\mathsf{w}_{w_1}}, a_{\mathsf{b}_v|\mathsf{b}_{w_1}}\} \subseteq A_v$, and $|\sigma_{w_{\langle \mathsf{w} \rangle}}| \geq |\sigma_v|$, then ρ_v is specified according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is even} \end{cases}. \tag{39}$$

The relation $>_v$ is set according to Eq. 27, and $>_{v,w_{\langle \mathsf{w} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \bigcup_{a_v^i \in \rho_v, a^j_{w_{\langle \mathsf{w} \rangle}} \in \rho_{w_{\langle \mathsf{w} \rangle}}} \{a_v^i < a^j_{w_{\langle \mathsf{w} \rangle}} \mid i < j\} \cup \{a^j_{w_{\langle \mathsf{w} \rangle}} < a_v^i \mid i \geq j\} \tag{40}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w} \rangle}\}$, we set $>_{v,w} = \emptyset$.

In all the four cases above, $>_v \cup >_{v,w_{\langle \mathsf{w} \rangle}} \cup >_{w_{\langle \mathsf{w} \rangle}}$ constitutes a strict total order over the elements of ρ_v and $\rho_{w_{\langle \mathsf{w} \rangle}}$.

From Eqs. 27, 34, 36, 38, 40 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, Eq. 18 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

[$\langle \mathsf{w} \rangle > 0, \langle \mathsf{b} \rangle > 0, \langle \mathsf{w} \rangle \neq \langle \mathsf{b} \rangle$] In this case, the constructed plan ρ should perform more than one value change of v, with changes of v to w_v and b_v being performed by (a pair of types of) actions prevailed by the value of $w_{\langle \mathsf{w} \rangle}$ and $w_{\langle \mathsf{b} \rangle}$, respectively. From $\alpha < \infty$, we have $\varphi([\![\delta_{\mathsf{w}}(\langle \mathsf{w} \rangle), \delta_{\mathsf{b}}(\langle \mathsf{w} \rangle), \eta]\!], \sigma_{w_{\langle \mathsf{w} \rangle}}) = \varphi([\![\delta_{\mathsf{w}}(\langle \mathsf{b} \rangle), \delta_{\mathsf{b}}(\langle \mathsf{b} \rangle), \eta]\!], \sigma_{w_{\langle \mathsf{b} \rangle}}) = 0$, and this is due to the respective satisfaction of the conditions of cases (2) and (3) in Eq. 22. Given that, we distinguish between the following four settings[4].

(1) If $\{a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{b} \rangle}}}\} \subseteq A_v$, then ρ_v is specified according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w} \rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{b} \rangle}}}, & i \text{ is even} \end{cases}. \tag{41}$$

The relation $>_v$ over the action elements of ρ_v is set according to Eq. 27, the relation $>_{v,w_{\langle \mathsf{w} \rangle}}$ over the action elements of ρ_v and $\rho_{w_{\langle \mathsf{w} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w} \rangle}} = \{a_v^i < a^1_{w_{\langle \mathsf{w} \rangle}} \mid i \text{ is odd}, a_v^i \in \rho_v, a^1_{w_{\langle \mathsf{w} \rangle}} \in \rho_{w_{\langle \mathsf{w} \rangle}}\} \tag{42}$$

and the relation $>_{v,w_{\langle \mathsf{b} \rangle}}$ over the action elements of ρ_v and $\rho_{w_{\langle \mathsf{b} \rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{b} \rangle}} = \{a_v^i < a^1_{w_{\langle \mathsf{b} \rangle}} \mid i \text{ is even}, a_v^i \in \rho_v, a^1_{w_{\langle \mathsf{b} \rangle}} \in \rho_{w_{\langle \mathsf{b} \rangle}}\} \tag{43}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w} \rangle}, w_{\langle \mathsf{b} \rangle}\}$, we set $>_{v,w} = \emptyset$.

4. While the details are slightly different, the four settings here are conceptually similar to these in the previously considered case of $\langle \mathsf{w} \rangle > 0, \langle \mathsf{b} \rangle > 0, \langle \mathsf{w} \rangle = \langle \mathsf{b} \rangle$.

(2) Otherwise, if $\{a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w}\rangle}}}, a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{b}\rangle}}}\} \subseteq A_v$ and $|\sigma_{w_{\langle \mathsf{w}\rangle}}| > 1$, then we have $|\rho_{w_{\langle \mathsf{w}\rangle}}| \geq 1$. Given that, we again set ρ_v according to Eq. 25, but now with its action elements being set as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w}\rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{b}\rangle}}}, & i \text{ is even} \end{cases}. \tag{44}$$

The relation $>_v$ is set according to Eq. 27, $>_{v,w_{\langle \mathsf{w}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w}\rangle}} = \bigcup_{a_v^i \in \rho_v,\ i \text{ is odd}} \{a^1_{w_{\langle \mathsf{w}\rangle}} < a_v^i\} \cup \{a_v^i < a^2_{w_{\langle \mathsf{w}\rangle}} \mid a^2_{w_{\langle \mathsf{w}\rangle}} \in \rho_{w_{\langle \mathsf{w}\rangle}}\} \tag{45}$$

and $>_{v,w_{\langle \mathsf{b}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{b}\rangle}} = \{a_v^i < a^1_{w_{\langle \mathsf{b}\rangle}} \mid i \text{ is even}, a_v^i \in \rho_v, a^1_{w_{\langle \mathsf{b}\rangle}} \in \rho_{w_{\langle \mathsf{b}\rangle}}\} \tag{46}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w}\rangle}, w_{\langle \mathsf{b}\rangle}\}$, we set $>_{v,w} = \emptyset$.

(3) Otherwise, if $\{a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{b}\rangle}}}, a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w}\rangle}}}\} \subseteq A_v$, and $|\sigma_{w_{\langle \mathsf{b}\rangle}}| > 1$, then we have $|\rho_{w_{\langle \mathsf{b}\rangle}}| \geq 1$. Given that, we ρ_v is specified according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_{w_{\langle \mathsf{w}\rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{b}\rangle}}}, & i \text{ is even} \end{cases}. \tag{47}$$

The relation $>_v$ is set according to Eq. 27, $>_{v,w_{\langle \mathsf{w}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w}\rangle}} = \{a_v^i < a^1_{w_{\langle \mathsf{w}\rangle}} \mid i \text{ is odd}, a_v^i \in \rho_v, a^1_{w_{\langle \mathsf{w}\rangle}} \in \rho_{w_{\langle \mathsf{w}\rangle}}\} \tag{48}$$

and $>_{v,w_{\langle \mathsf{b}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{b}\rangle}} = \bigcup_{a_v^i \in \rho_v,\ i \text{ is even}} \{a^1_{w_{\langle \mathsf{b}\rangle}} < a_v^i\} \cup \{a_v^i < a^2_{w_{\langle \mathsf{b}\rangle}} \mid a^2_{w_{\langle \mathsf{b}\rangle}} \in \rho_{w_{\langle \mathsf{b}\rangle}}\} \tag{49}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w}\rangle}\}$, we set $>_{v,w} = \emptyset$.

(4) Otherwise, if $\{a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w}\rangle}}}, a_{\mathsf{b}_v|\mathsf{w}_{w_{\langle \mathsf{b}\rangle}}}\} \subseteq A_v$, $|\sigma_{w_{\langle \mathsf{w}\rangle}}| > 1$, and $|\sigma_{w_{\langle \mathsf{b}\rangle}}| > 1$, then we have both $|\rho_{w_{\langle \mathsf{w}\rangle}}| \geq 1$ and $|\rho_{w_{\langle \mathsf{b}\rangle}}| \geq 1$. Given that, we again set ρ_v according to Eq. 25, and its action elements are specified as

$$a_v^i = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_{w_{\langle \mathsf{w}\rangle}}}, & i \text{ is odd} \\ a_{\mathsf{b}_v|\mathsf{b}_{w_{\langle \mathsf{b}\rangle}}}, & i \text{ is even} \end{cases}. \tag{50}$$

The relation $>_v$ is set according to Eq. 27, $>_{v,w_{\langle \mathsf{w}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{w}\rangle}} = \bigcup_{a_v^i \in \rho_v,\ i \text{ is odd}} \{a^1_{w_{\langle \mathsf{w}\rangle}} < a_v^i\} \cup \{a_v^i < a^2_{w_{\langle \mathsf{w}\rangle}} \mid a^2_{w_{\langle \mathsf{w}\rangle}} \in \rho_{w_{\langle \mathsf{w}\rangle}}\} \tag{51}$$

and $>_{v,w_{\langle \mathsf{b}\rangle}}$ is set to

$$>_{v,w_{\langle \mathsf{b}\rangle}} = \bigcup_{a_v^i \in \rho_v,\ i \text{ is even}} \{a^1_{w_{\langle \mathsf{b}\rangle}} < a_v^i\} \cup \{a_v^i < a^2_{w_{\langle \mathsf{b}\rangle}} \mid a^2_{w_{\langle \mathsf{b}\rangle}} \in \rho_{w_{\langle \mathsf{b}\rangle}}\} \tag{52}$$

For all $w \in \mathsf{pred}(v) \setminus \{w_{\langle \mathsf{w}\rangle}\}$, we set $>_{v,w} = \emptyset$.

In all the four cases above, both $>_v \cup >_{v,w_{\langle \mathsf{w} \rangle}} \cup >_{w_{\langle \mathsf{w} \rangle}}$ and $>_v \cup >_{v,w_{\langle \mathsf{b} \rangle}} \cup >_{w_{\langle \mathsf{b} \rangle}}$ constitute strict total orders over their respective domains.

From Eqs. 27, 42, 43, 45, 46, 48, 49, 51, 52 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, Eq. 18 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Until now, for each variable $v \in V$, we have specified the action sequence ρ_v and the order $>_v$ over the elements of ρ_v. For each $w \in \mathsf{pred}(v)$, we have specified the order $>_{v,w}$, and proved that all $>_v \cup >_{v,w}$ and $>_w \cup >_{v,w}$ form strict partial orders over their domains, and any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$ and provides to v the value $G[v]$ if the latter is specified. This construction allows us to apply now Theorem 1 on the (considered as sets) sequences ρ_v and orders $>_v$ and $>_{v,w}$, proving that

$$>= \bigcup_{v \in V} (>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w})$$

forms a strict partial order over the union of $\rho_{v_1}, \ldots, \rho_{v_n}$.

Here we also note that the plan extraction step of the algorithm polytree-1-dep-uniform corresponds exactly to the above construction along Eqs. 25-52, providing us in poly-time with concrete cost-optimal plan corresponding to the optimal solution for COP_Π.

(II) We now prove that if Π is solvable, then we must have $\alpha < \infty$. Assume to the contrary that this is not the case. Let Π be a solvable $\mathbf{P}(1)$ problem, and let (using Theorem 4) ρ be an irreducible, post-unique plan for Π. Given such ρ, let a COP assignment $\overline{x}_\rho$ be defined as follows.

1. For each COP variable x_v, the assignment $\overline{x}_\rho$ provides the value $\sigma_v \in \trianglerighteq^*[\sigma(v)]$ such that $|\sigma_v| = |\rho\!\downarrow_v| + 1$.
2. For each COP variable x_v^w, the assignment $\overline{x}_\rho$ provides the value $[\![\delta^w_{\mathsf{w}_v}, \delta^w_{\mathsf{b}_v}, |\sigma_v| - 1]\!]$, where $\delta^w_{\mathsf{w}_v} = 1$ if some action in $\rho\!\downarrow_v$ preconditioned by the value of w changes the value of v to w_v, and $\delta^w_{\mathsf{w}_v} = 0$, otherwise. $\delta^w_{\mathsf{b}_v}$ is defined similarly to $\delta^w_{\mathsf{w}_v}$, *mutatis mutandis.*

From Eq. 20-24 we then directly have that, for all $v \in V$, $\varphi_{x_v}(\overline{x}_\rho) = |\rho\!\downarrow_v|$, and for all $w \in \mathsf{pred}(v)$, $\varphi_{x_v^w}(\overline{x}_\rho) = 0$. Therefore, we have

$$\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}_\rho) \;=\; \sum_{v \in V} \mathcal{C}(\rho\!\downarrow_v) \;=\; \mathcal{C}(\rho),$$

which is what we had to prove. ■

Theorem 5 *Cost-optimal planning for* $\mathbf{P}(1)$ *with uniform cost actions is tractable.*

Proof: The correctness of the polytree-1-dep-uniform algorithm is given by Lemma 3. We now show that, given a planning problem $\Pi \in \mathbf{P}(1)$ with uniform cost actions, the corresponding constraint optimization problem COP_Π can be constructed and solved in time polynomial in the description size of Π.

Let n be the number of state variables in Π. In polytree-1-dep-uniform, we first construct the constraint optimization problem COP_Π over $\Theta(n^2)$ variables $\mathcal{X}$ with domain sizes bounded by $O(n)$, and $\Theta(n^2)$ functional components $\mathcal{F}$, each defined over at most three COP variables. The construction is linear in the size of the resulting COP, and thus is accomplished in time $O(n^5)$.

Applying then to COP_Π a tree-decomposition that clusters the scopes of the functional components $\mathcal{F}$, we arrive into an equivalent, tree-structured constraint optimization problem over $\Theta(n^2)$ variables with domains of size $O(n^3)$. Such a tree-structured COP can be solved in time $O(xy^2)$ where x is the number of variables and y is an upper bound on the size of a variable's domain (Dechter, 2003). Therefore, solving COP_Π can be done in time $O(n^8)$. As this dominates both the time complexity of constructing COP_Π, and the time complexity of extracting a plan from the optimal solution to COP_Π (see the proof of (I) in Lemma 3), the overall complexity of the algorithm polytree-1-dep-uniform is $O(n^8)$, and therefore polynomial in the description size of Π. ■

5. Cost-Optimal Planning for P(1) with General Action Costs

We now consider cost-optimal planning for $\mathbf{P}(1)$ problems without an additional constraint on actions having all the same cost. While Theorem 4 in Section 4 shows that any solvable $\mathbf{P}(1)$ problem Π has at least one post-unique plan, it is possible that no such plan is cost-optimal for Π. Fortunately, here we show that any solvable $\mathbf{P}(1)$ problem is guaranteed to have a cost-optimal plan satisfying a certain relaxation of action sequence post-uniqueness, and this weaker property still allows to devise a (slightly more costly than polytree-1-dep-uniform) planning-to-COP scheme for general $\mathbf{P}(1)$ problems.

5.1 Post-3/2 Plans and P(1) Problems

We now proceed with introducing the notion of post-3/2 property for action sequences that relaxes the post-uniqueness property exploited in the previous section.

Definition 5 *Let $\Pi = (V, A, I, G)$ be a* UB *problem instance. An action sequence $\varrho \in A^*$ is called* **post-3/2** *if, for each $v \in V$, $a \in \varrho_v$, there exist $\alpha \neq \beta \in \{\mathsf{b}_v, \mathsf{w}_v\}$, parent $w \in Pa(v)$, $\gamma, \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\xi \in \{\mathsf{b}_u, \mathsf{w}_u \mid u \in Pa(v)\}$, such that $a \in \{a_{\alpha|\gamma}, a_{\beta|\delta}, a_{\alpha|\xi}\}$. That is, all the changes of each variable are done using at most three types of actions which are prevailed by at most two parents, and if u is different from w, then different actions prevailed by w perform different value changes of v.*

The (possibly empty) set of all **post-3/2 plans** *for Π is denoted by $\mathcal{P}^{3/2}(\Pi)$ (or simply $\mathcal{P}^{3/2}$, if the identity of Π is clear from the context).*

It is easy to verify that post-3/2 is a relaxation of post-uniqueness—if a plan is post-unique, then it is post-3/2, but not necessarily the other way around. Turns out that, for any $\mathbf{P}(1)$ problem Π, this relaxed property is guaranteed to be satisfied by at least one cost-optimal plan for Π.

Theorem 6 *For every solvable $\mathbf{P}(1)$ problem $\Pi = (V, A, I, G)$, the plan set $\mathcal{P}^{3/2}(\Pi)$ contains at least one cost-optimal plan.*

Proof: As the correctness of second claim immediately implies the correctness of the first one, here we focus on the proof the second claim. Given a $\mathbf{P}(1)$ problem $\Pi = (V, A, I, G)$, and cost-optimal plan ρ for Π, we construct a sequence of actions ρ^* such that:

- ρ^* is a *post*-3/2 plan for Π,
- $\mathcal{C}(\rho^*) = \mathcal{C}(\rho)$.

In nutshell, first, for each $v \in V$, we map the subsequence $\rho\!\downarrow_v = \langle a_1, \ldots, a_k \rangle$ of ρ into a sequence of actions $\rho^*_v = \langle a^*_1, \ldots, a^*_k \rangle$ that (i) satisfy the *post*-3/2 property, and (ii) $\mathcal{C}(\rho^*_v) \leq \mathcal{C}(\rho\!\downarrow_v)$. Then, we merge the constructed sequences $\{\rho^*_v\}_{v \in V}$ into ρ^*, and show that ρ^* is a valid plan for Π. Note that the two properties of ρ^* as required above will then hold immediately because $\mathcal{C}(\rho^*) = \mathcal{C}(\rho)$, and ρ^* being post-3/2 is implied by all its per-variable components ρ^*_v being post-3/2.

For each variable $v \in V$, given $\{\sigma_w\}_{w \in \mathsf{pred}(v)}$, such that $|\sigma_w| = |\rho\!\downarrow_w| + 1$, let a^α_i be the i'th cheapest action that changes variable v to $\alpha \in \{\mathsf{b}_v, \mathsf{w}_v\}$ and prevailed by some value from $\{\sigma_w\}_{w \in \mathsf{pred}(v)}$. Lets now focus on the $a^{\mathsf{w}}_1 = a_{\mathsf{w}_v|\gamma}$, $a^{\mathsf{w}}_2 = a_{\mathsf{w}_v|\mu}$, $a^{\mathsf{b}}_1 = a_{\mathsf{b}_v|\delta}$, $a^{\mathsf{b}}_2 = a_{\mathsf{b}_v|\nu}$.

(I) If $\gamma = \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, we set

$$a^*_i = \begin{cases} a_{\mathsf{w}_v|\gamma} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\gamma} & \text{otherwise} \end{cases} \tag{53}$$

In addition, we construct the following sets of ordering constraints. First, we set a binary relation $>_v$ over the action elements of $\rho^*_v = \langle a^*_1, \ldots, a^*_k \rangle$ to

$$>_v = \{a^*_i < a^*_j \mid a^*_i, a^*_j \in \rho^*_v, i < j\}. \tag{54}$$

It is immediate from Eq. 54 that $>_v$ is a strict total order over the elements of ρ^*_v. Likewise, if $\rho^*_w = \langle a_{j_1}, \ldots, a_{j_l} \rangle$, we set

$$>_{v,w} = \begin{cases} \bigcup_{a^*_i \in \rho^*_v} \{a^*_i < a_{j_1}\}, & \gamma = \mathsf{b}_w \\ \bigcup_{a^*_i \in \rho^*_v} \{a^*_i > a_{j_1}\}, & \gamma = \mathsf{w}_w, l = 1 \\ \bigcup_{a^*_i \in \rho^*_v} \{a^*_i > a_{j_1}\} \cup \{a^*_i < a_{j_2}\}, & \gamma = \mathsf{w}_w, l > 1 \end{cases} \tag{55}$$

Finally, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{w\}$ are set to empty sets.

For each $w \in \mathsf{pred}(v)$, it is easy to verify that the relation $>_{v,w}$ defined by Eq. 55 is a strict total order over its domain. Also, from Eqs. 54 and 55, we have that, for each $w \in \mathsf{pred}(v)$, $>_v \cup >_{v,w}$ is a strict total order over the union of the elements of ρ^*_v and ρ^*_w.

From Eqs. 54-55 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho\!\downarrow_v|$ together with Eq. 53 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(II) If $\gamma \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\delta \in \{\mathsf{b}_u, \mathsf{w}_u\}$, such that $w \neq u$, we set

$$a_i^* = \begin{cases} a_{\mathsf{w}_v|\gamma} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\delta} & \text{otherwise} \end{cases} \tag{56}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_l \rangle$, and $\rho_u^* = \langle a'_1, \dots, a'_{l'} \rangle$, we set $>_{v,w}$ according to Eq. 55 above, and $>_{v,u}$ according to Eq. 57 below.

$$>_{v,u} = \begin{cases} \bigcup_{a_i^* \in \rho_v^*} \{a_i^* < a'_1\}, & \nu = \mathsf{b}_u \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a'_1\}, & \nu = \mathsf{w}_u, l' = 1 \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a'_1\} \cup \{a_i^* < a'_2\}, & \delta = \mathsf{w}_u, l' > 1 \end{cases} \tag{57}$$

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets.

The relations $>_v$ here is identical to these in previous case, and relations $>_{v,u}$ and $>_{v,w}$ are effectively identical to the relation $>_{v,w}$ in previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.

From Eqs. 54, 55, 57 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 56 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(III) If $\gamma = \mathsf{b}_w, \delta = \mathsf{w}_w$, we distinguish between a few cases based on σ_w and σ_v.

(1) If $|\rho{\downarrow_v}| = 2y+1$, $|\sigma_w| = 2x$, $|\sigma_w| \leq |\rho{\downarrow_v}|$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_1^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, j < x \\ a_2^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, x \leq j \leq y \\ a_1^{\mathsf{w}} & \text{otherwise} \end{cases} \tag{58}$$

And the cost in this case is

$$(y+1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (x-1) \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{59}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_{2x-1} \rangle$, and $\rho_u^* = \langle a'_1, \dots, a'_{l'} \rangle$, we set $>_{v,w}$ according to Eq. 60, and $>_{v,u}$ according to Eq. 61.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i \leq j < 2x-1\} \cup \{a_i^* < a_{2x-1}\} \cup \{a_j < a_i^* \mid j < i, j < 2x-1\} \tag{60}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set,

$$>_{v,u}= \begin{cases} \bigcup_{a_i^* \in \rho_v^*} \{a_i^* < a_1'\}, & \nu = \mathsf{b}_u \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_1'\}, & \nu = \mathsf{w}_u, l' = 1 \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_1'\} \cup \{a_i^* < a_2'\}, & \nu = \mathsf{w}_u, l' > 1 \\ \emptyset, & \text{otherwise} \end{cases}. \tag{61}$$

It is not hard to verify that the relation $>_{v,w}$ defined by Eq. 60 is a strict total order over its domain. Suppose to the contrary that for some i, j, both $a_j < a_i^*$ and $a_i^* < a_j$. Then from first inequality we have either $i \leq j < 2x-1$ or $j = 2x-1$, and from second we have $j < i, j < 2x-1$.
The relations $>_v$ and $>_{v,u}$ are effectively identical to these in case (II). Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively. From Eqs. 54, 60, 61 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 58 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining changes to w_v being done using action a_2^{w}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_1^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, j \leq x \\ a_2^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, x < j \leq y+1 \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{62}$$

And the cost in this case is

$$x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{63}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1} \rangle$, and $\rho_u^* = \langle a_1', \ldots, a_{l'}' \rangle$, we set $>_{v,w}$ according to Eq. 64, and $>_{v,u}$ according to Eq. 65.

$$>_{v,w}= \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i \leq j\} \cup \{a_j < a_i^* \mid j < i\} \tag{64}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set,

$$>_{v,u}= \begin{cases} \bigcup_{a_i^* \in \rho_v^*} \{a_i^* < a_1'\}, & \mu = \mathsf{b}_u \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_1'\}, & \mu = \mathsf{w}_u, l' = 1 \\ \bigcup_{a_i^* \in \rho_v^*} \{a_i^* > a_1'\} \cup \{a_i^* < a_2'\}, & \mu = \mathsf{w}_u, l' > 1 \\ \emptyset, & \text{otherwise} \end{cases}. \tag{65}$$

The relation $>_{v,w}$ defined by Eq. 64 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to these in case (II). Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ^*_v and ρ^*_u, and ρ^*_v and ρ^*_w, respectively. From Eqs. 54, 64, 65 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 62 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y + x$ actions of both types a^{w}_1 and a^{b}_1 totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y + x + 1$ actions of types a^{w}_1 and a^{b}_1. Then it contains no more than $y - x$ actions of other types. Let $\mathsf{b}_w \cdot \mathsf{w}_w \cdot \ldots \cdot \mathsf{b}_w$ sequence of $2y + 1$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y - x$ actions of other type will decrease the length in at most $2y - 2x$, and we are left with the sequence of length $\geq 2y + 1 - (2y - 2x) = 2x + 1$. Therefore σ_w cannot support more than $y + x$ actions of types a^{w}_1 and a^{b}_1. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a^{w}_1 and β actions of type a^{b}_1. Then

$$\alpha + \beta \leq y + x \tag{66}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y + 1 - \alpha) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + \beta \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y - \beta) \cdot \mathcal{C}(a^{\mathsf{b}}_2) \tag{67}$$

For (59) $\leq$ (63), we have

$$\mathcal{C}(a^{\mathsf{b}}_2) - \mathcal{C}(a^{\mathsf{b}}_1) \leq \mathcal{C}(a^{\mathsf{w}}_2) - \mathcal{C}(a^{\mathsf{w}}_1) \tag{68}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 67 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y + 1 - \alpha) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + \beta \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y - \beta) \cdot \mathcal{C}(a^{\mathsf{b}}_2) < \\ &(y + 1) \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (x - 1) \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y - x + 1) \cdot \mathcal{C}(a^{\mathsf{b}}_2)\end{aligned}$$

and from it

$$(y + 1 - \alpha) \cdot (\mathcal{C}(a^{\mathsf{w}}_2) - \mathcal{C}(a^{\mathsf{w}}_1)) < (\beta - x + 1) \cdot (\mathcal{C}(a^{\mathsf{b}}_2) - \mathcal{C}(a^{\mathsf{b}}_1)) \tag{69}$$

From Eq. 66 we have $y + 1 - \alpha \geq \beta - x + 1$, together with Eq. 68 contradicting Eq. 69.

For (63) $\leq$ (59), we have

$$\mathcal{C}(a^{\mathsf{w}}_2) - \mathcal{C}(a^{\mathsf{w}}_1) \leq \mathcal{C}(a^{\mathsf{b}}_2) - \mathcal{C}(a^{\mathsf{b}}_1) \tag{70}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 67 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y + 1 - \alpha) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + \beta \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y - \beta) \cdot \mathcal{C}(a^{\mathsf{b}}_2) < \\ &x \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y - x + 1) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + y \cdot \mathcal{C}(a^{\mathsf{b}}_1)\end{aligned}$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}))<(\alpha-x)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})) \tag{71}$$

From Eq. 66 we have $y-\beta\geq\alpha-x$, together with Eq. 70 contradicting Eq. 71.

(2) If $|\rho_v|=2y+1$, $|\sigma_w|=2x$, $|\sigma_w|>|\rho_v|$, then the actions of ρ_v^* are set to

$$a_i^*=\begin{cases}a_{\mathsf{w}_v|\mathsf{b}_w} & i=2j-1, j\in\mathbb{N}\\ a_{\mathsf{b}_v|\mathsf{w}_w} & \text{otherwise}\end{cases} \tag{72}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^*=\langle a_1,\ldots,a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 64 above.

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w'\in\mathsf{pred}(v)\setminus\{u,w\}$ are set to empty sets.

The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v\cup>_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 64 we can now derive that any linearization of $>_v\cup\bigcup_{w\in\mathsf{pred}(v)}>_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\}\cup\mathsf{pred}(v)$. In addition, $|\rho_v^*|=|\rho\downarrow_v|$ together with Eq. 72 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(3) If $|\rho_v|=2y$, $|\sigma_w|=2x$, $|\sigma_w|<|\rho_v|$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho\downarrow_v)$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^*=\begin{cases}a_2^{\mathsf{b}} & i=2j, j\in\mathbb{N}, j\leq y-x\\ a_1^{\mathsf{b}} & i=2j, j\in\mathbb{N}, y-x<j\leq y\\ a_1^{\mathsf{w}} & \text{otherwise}\end{cases} \tag{73}$$

And the cost in this case is

$$y\cdot\mathcal{C}(a_1^{\mathsf{w}})+x\cdot\mathcal{C}(a_1^{\mathsf{b}})+(y-x)\cdot\mathcal{C}(a_2^{\mathsf{b}}) \tag{74}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^*=\langle a_1,\ldots,a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 75.

$$>_{v,w}=\bigcup_{a_i^*\in\rho_v^*, a_j\in\rho_w^*}\{a_i^*<a_j\mid i\leq 2y-2x+j\}\cup\{a_j<a_i^*\mid i>2y-2x+j\} \tag{75}$$

For each $u\in\mathsf{pred}(v)\setminus\{w\}$ we set $>_{v,u}$ according to Eq. 61. It is easy to verify that the relation $>_{v,w}$ defined by Eq. 75 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous

case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ^*_v and ρ^*_u, and ρ^*_v and ρ^*_w, respectively.
From Eqs. 54, 61, 75 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 73 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a^{b}_1, and then the maximally possible number of changes to w_v are done using action a^{w}_1, with the remaining changes to w_v being done using action a^{w}_2.* For this candidate for ρ^*_v, we set

$$a^*_i = \begin{cases} a^{\mathsf{w}}_1 & i = 2j-1, j \in \mathbb{N}, j \leq x \\ a^{\mathsf{w}}_2 & i = 2j-1, j \in \mathbb{N}, x < j \leq y \\ a^{\mathsf{b}}_1 & \text{otherwise} \end{cases} \tag{76}$$

And the cost in this case is

$$x \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y - x) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + y \cdot \mathcal{C}(a^{\mathsf{b}}_1) \tag{77}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho^*_w = \langle a_1, \ldots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 64 above.
For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65.
The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ^*_v and ρ^*_u, and ρ^*_v and ρ^*_w, respectively.
From Eqs. 54, 64, 65 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 76 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y + x$ actions of both types a^{w}_1 and a^{b}_1 totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y + x + 1$ actions of types a^{w}_1 and a^{b}_1. Then it contains no more than $y - x - 1$ actions of other types. Let $\mathsf{b}_w \cdot \mathsf{w}_w \cdot \ldots \cdot \mathsf{w}_w$ sequence of $2y$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y - x - 1$ actions of other type will decrease the length in at most $2y - 2x - 2$, and we are left with the sequence of length $\geq 2y - (2y - 2x - 2) = 2x + 2$. Therefore σ_w cannot support more than $y + x$ actions of types a^{w}_1 and a^{b}_1. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a^{w}_1 and β actions of type a^{b}_1. Then

$$\alpha + \beta \leq y + x \tag{78}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a^{\mathsf{w}}_1) + (y - \alpha) \cdot \mathcal{C}(a^{\mathsf{w}}_2) + \beta \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y - \beta) \cdot \mathcal{C}(a^{\mathsf{b}}_2) \tag{79}$$

For (74) $\leq$ (77), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{80}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 79 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{b}})\end{aligned}$$

and from it

$$(y-\alpha) \cdot (\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) < (\beta - x) \cdot (\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) \tag{81}$$

From Eq. 78 we have $y - \alpha \geq \beta - x$, together with Eq. 80 contradicting Eq. 81. For (77) $\leq$ (74), we have

$$\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \leq \mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \tag{82}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 79 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}})\end{aligned}$$

and from it

$$(y-\beta) \cdot (\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) < (\alpha - x) \cdot (\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) \tag{83}$$

From Eq. 78 we have $y - \beta \geq \alpha - x$, together with Eq. 82 contradicting Eq. 83.

(4) If $|\rho_v| = 2y$, $|\sigma_w| = 2x$, $|\sigma_w| \geq |\rho_v|$, then the actions of ρ_v^* are set to

$$a_i^* = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_w} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\mathsf{w}_w} & \text{otherwise} \end{cases} \tag{84}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 64 above. Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets. The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 64 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 84 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(5) If $|\rho_v| = 2y+1$, $|\sigma_w| = 2x+1$, $|\sigma_w| < |\rho_v|$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, j \leq y - x \\ a_1^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, y - x < j \leq y \\ a_1^{\mathsf{w}} & \text{otherwise} \end{cases} \tag{85}$$

And the cost in this case is

$$(y+1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y - x) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{86}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 75 above. For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61 above. The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 61, 75 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow}_v|$ together with Eq. 85 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining changes to w_v being done using action a_2^{w}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_1^{\mathsf{w}} & i = 2j - 1, j \in \mathbb{N}, j \leq x \text{ or } j = y + 1 \\ a_2^{\mathsf{w}} & i = 2j - 1, j \in \mathbb{N}, x < j \leq y \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{87}$$

And the cost in this case is

$$(x+1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y - x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{88}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 89.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \begin{array}{c} \{a_i^* < a_j \mid i \leq j < 2x\} \cup \{a_i^* < a_{2x} \mid i \leq 2y\} \\ \cup \{a_j < a_i^* \mid j < i, j < 2x\} \cup \{a_{2x} < a_{2y+1}^*\} \end{array} \tag{89}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65 above.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 89 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.

From Eqs. 54, 65, 89 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 87 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y+x+1$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y+x+2$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x-1$ actions of other types. Let $\mathsf{b}_w \cdot \mathsf{w}_w \cdot \ldots \cdot \mathsf{b}_w$ sequence of $2y+1$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x-1$ actions of other type will decrease the length in at most $2y-2x-2$, and we are left with the sequence of length $\geq 2y+1-(2y-2x-2) = 2x+3$. Therefore σ_w cannot support more than $y+x+1$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha + \beta \leq y + x + 1 \tag{90}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{91}$$

For (86) $\leq$ (88), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{92}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 91 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &(y+1)\cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x)\cdot \mathcal{C}(a_2^{\mathsf{b}})\end{aligned}$$

and from it

$$(y+1-\alpha)\cdot(\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) < (\beta - x)\cdot(\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) \tag{93}$$

From Eq. 90 we have $y+1-\alpha \geq \beta - x$, together with Eq. 92 contradicting Eq. 93. For (88) $\leq$ (86), we have

$$\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \leq \mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \tag{94}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 91 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &(x+1 \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}})\end{aligned}$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) < (\alpha - x - 1)\cdot(\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) \tag{95}$$

From Eq. 90 we have $y-\beta \geq \alpha - x - 1$, together with Eq. 94 contradicting Eq. 95.

(6) If $|\rho_v| = 2y+1$, $|\sigma_w| = 2x+1$, $|\sigma_w| \geq |\rho_v|$, the actions of ρ^*_v are set to

$$a^*_i = \begin{cases} a_{\mathsf{w}_v|\mathsf{b}_w} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\mathsf{w}_w} & \text{otherwise} \end{cases} \tag{96}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho^*_w = \langle a_1, \dots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 64 above. Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets. The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,w}$ forming strict partial order over the union of the elements of ρ^*_v and ρ^*_w.

From Eqs. 54, 64 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 96 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(7) If $|\rho_v| = 2y$, $|\sigma_w| = 2x+1$, $|\sigma_w| \leq |\rho_v|$, then we construct two post-3/2 candidates for ρ^*_v, and then assign ρ^*_v to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a^{w}_1, and then the maximally possible number of changes to b_v are done using action a^{b}_1, with the remaining changes to b_v being done using action a^{b}_2.* For this candidate for ρ^*_v, we set

$$a^*_i = \begin{cases} a^{\mathsf{b}}_2 & i = 2j, j \in \mathbb{N}, j \leq y-x \\ a^{\mathsf{b}}_1 & i = 2j, j \in \mathbb{N}, y-x < j \leq y \\ a^{\mathsf{w}}_1 & \text{otherwise} \end{cases} \tag{97}$$

And the cost in this case is

$$y \cdot \mathcal{C}(a^{\mathsf{w}}_1) + x \cdot \mathcal{C}(a^{\mathsf{b}}_1) + (y-x) \cdot \mathcal{C}(a^{\mathsf{b}}_2) \tag{98}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho^*_w = \langle a_1, \dots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 75 above. For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61 above. The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ^*_v and ρ^*_u, and ρ^*_v and ρ^*_w, respectively.

From Eqs. 54, 61, 75 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho^*_v| = |\rho{\downarrow_v}|$ together with Eq. 97 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a^{b}_1, and then the maximally possible number of changes to w_v are done using action a^{w}_1, with the remaining changes to w_v being done using action a^{w}_2.* For this candidate for ρ^*_v, we set

$$a^*_i = \begin{cases} a^{\mathsf{w}}_1 & i = 2j-1, j \in \mathbb{N}, j \leq x \\ a^{\mathsf{w}}_2 & i = 2j-1, j \in \mathbb{N}, x < j \leq y \\ a^{\mathsf{b}}_1 & \text{otherwise} \end{cases} \tag{99}$$

And the cost in this case is

$$x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y - x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{100}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 101.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i \leq j < 2x\} \cup \{a_j < a_i^* \mid j < i, j < 2x\} \cup \{a_i^* < a_{2x}\} \tag{101}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65 above.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 101 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 65, 101 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 99 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y+x$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y+x+1$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x-1$ actions of other types. Let $\mathsf{b}_w \cdot \mathsf{w}_w \cdot \ldots \cdot \mathsf{w}_w$ sequence of $2y$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x-1$ actions of other type will decrease the length in at most $2y-2x-2$, and we are left with the sequence of length $\geq 2y-(2y-2x-2) = 2x+2$. Therefore σ_w cannot support more than $y+x$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha + \beta \leq y + x \tag{102}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{103}$$

For (98) $\leq$ (100), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{104}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 103 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{b}})\end{aligned}$$

and from it

$$(y-\alpha)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}}))<(\beta-x)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}})) \tag{105}$$

From Eq. 102 we have $y-\alpha\geq\beta-x$, together with Eq. 104 contradicting Eq. 105. For (100) $\leq$ (98), we have

$$\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})\leq\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}) \tag{106}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 103 we have

$$\alpha\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-\alpha)\cdot\mathcal{C}(a_2^{\mathsf{w}})+\beta\cdot\mathcal{C}(a_1^{\mathsf{b}})+(y-\beta)\cdot\mathcal{C}(a_2^{\mathsf{b}})<$$
$$x\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-x)\cdot\mathcal{C}(a_2^{\mathsf{w}})+y\cdot\mathcal{C}(a_1^{\mathsf{b}})$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}))<(\alpha-x)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})) \tag{107}$$

From Eq. 102 we have $y-\beta\geq\alpha-x$, together with Eq. 106 contradicting Eq. 107.

(8) If $|\rho_v|=2y$, $|\sigma_w|=2x+1$, $|\sigma_w|>|\rho_v|$, then the actions of ρ_v^* are set to

$$a_i^*=\begin{cases}a_{\mathsf{w}_v|\mathsf{b}_w} & i=2j-1, j\in\mathbb{N}\\ a_{\mathsf{b}_v|\mathsf{w}_w} & \text{otherwise}\end{cases} \tag{108}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^*=\langle a_1,\ldots,a_{2x}\rangle$, we set $>_{v,w}$ according to Eq. 64 above. Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w'\in\mathsf{pred}(v)\setminus\{u,w\}$ are set to empty sets. The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v\cup>_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 64 we can now derive that any linearization of $>_v\cup\bigcup_{w\in\mathsf{pred}(v)}>_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\}\cup\mathsf{pred}(v)$. In addition, $|\rho_v^*|=|\rho{\downarrow_v}|$ together with Eq. 108 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(IV) If $\gamma=\mathsf{w}_w,\delta=\mathsf{b}_w$, we distinguish between a few cases based on σ_w and σ_v.

(1) If $|\rho_v|=2y+1$, $|\sigma_w|=2x$, $|\sigma_w|\leq|\rho_v|$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^*=\begin{cases}a_2^{\mathsf{b}} & i=2j, j\in\mathbb{N}, j\leq y-x+1\\ a_1^{\mathsf{b}} & i=2j, j\in\mathbb{N}, y-x+1<j\leq y\\ a_1^{\mathsf{w}} & \text{otherwise}\end{cases} \tag{109}$$

And the cost in this case is

$$(y+1)\cdot\mathcal{C}(a_1^{\mathsf{w}}) + (x-1)\cdot\mathcal{C}(a_1^{\mathsf{b}}) + (y-x+1)\cdot\mathcal{C}(a_2^{\mathsf{b}}) \tag{110}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 111.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i < j\} \cup \{a_j < a_i^* \mid j \leq i\} \tag{111}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 111 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 61, 111 we can now derive that any linearization of $>_v \cup \bigcup_{w\in\mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 109 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining changes to w_v being done using action a_2^{w}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, j \leq y-x+1 \\ a_1^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, y-x+1 < j \leq y+1 \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{112}$$

And the cost in this case is

$$x\cdot\mathcal{C}(a_1^{\mathsf{w}}) + (y+1-x)\cdot\mathcal{C}(a_2^{\mathsf{w}}) + y\cdot\mathcal{C}(a_1^{\mathsf{b}}) \tag{113}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 114.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i \leq 2y-2x+1+j\} \cup \{a_j < a_i^* \mid i > 2y-2x+1+j\} \tag{114}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 114 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 65, 114 we can now derive that any linearization of $>_v \cup \bigcup_{w\in\mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 112 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y+x$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y+x+1$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x$ actions of other types. Let $\mathsf{w}_w \cdot \mathsf{w}_b \cdot \ldots \cdot \mathsf{w}_w$ sequence of $2y+1$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x$ actions of other type will decrease the length in at most $2y-2x$, and we are left with the sequence of length $\geq 2y+1-(2y-2x) = 2x+1$. Therefore σ_w cannot support more than $y+x$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha + \beta \leq y + x \tag{115}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{116}$$

For (110) $\leq$ (113), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{117}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 116 we have

$$\begin{aligned} &\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &(y+1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (x-1) \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \end{aligned}$$

and from it

$$(y+1-\alpha) \cdot (\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) < (\beta - x + 1) \cdot (\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) \tag{118}$$

From Eq. 115 we have $y+1-\alpha \geq \beta - x + 1$, together with Eq. 117 contradicting Eq. 118.
For (113) $\leq$ (110), we have

$$\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \leq \mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \tag{119}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 116 we have

$$\begin{aligned} &\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \end{aligned}$$

and from it

$$(y-\beta) \cdot (\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) < (\alpha - x) \cdot (\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) \tag{120}$$

From Eq. 115 we have $y-\beta \geq \alpha - x$, together with Eq. 119 contradicting Eq. 120.

(2) If $|\rho_v| = 2y+1$, $|\sigma_w| = 2x$, $|\sigma_w| > |\rho_v|$, then the actions of ρ_v^* are set to

$$a_i^* = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_w} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\mathsf{b}_w} & \text{otherwise} \end{cases} \tag{121}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 111 above.

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets.

The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 111 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 121 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(3) If $|\rho_v| = 2y$, $|\sigma_w| = 2x$, $|\sigma_w| \leq |\rho_v|+1$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_1^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, j < x \\ a_2^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, x \leq j \leq y \\ a_1^{\mathsf{w}} & \text{otherwise} \end{cases} \tag{122}$$

And the cost in this case is

$$y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (x-1) \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{123}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 111 above.

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61.

The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.

From Eqs. 54, 61, 111 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 122 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining* *changes to* w_v *being done using action* a_2^{w}. For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, j \leq y-x+1 \\ a_1^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, y-x+1 < j \leq y \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{124}$$

And the cost in this case is

$$(x-1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{125}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 114 above.
For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65.
The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 65, 114 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho\downarrow_v|$ together with Eq. 124 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho\downarrow_v$ cannot contain more than $y+x-1$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho\downarrow_v$ contain at least $y+x$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x$ actions of other types. Let $\mathsf{w}_w \cdot \mathsf{b}_w \cdot \ldots \cdot \mathsf{b}_w$ sequence of $2y$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x$ actions of other type will decrease the length in at most $2y-2x$, and we are left with the sequence of length $\geq 2y-(2y-2x) = 2x$, which have to be a subsequence of σ_w, contradicting with the fact that σ_w is of the same or smaller size and starts with a different character. Therefore σ_w cannot support more than $y+x$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho\downarrow_v$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha + \beta \leq y + x \tag{126}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{127}$$

For (123) $\leq$ (125), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{128}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 127 we have

$$\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) <$$
$$y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{b}})$$

and from it

$$(y-\alpha)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}}))<(\beta-x)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}})) \tag{129}$$

From Eq. 126 we have $y-\alpha\geq\beta-x$, together with Eq. 128 contradicting Eq. 129. For (125) $\leq$ (123), we have

$$\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})\leq\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}) \tag{130}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 127 we have

$$\alpha\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-\alpha)\cdot\mathcal{C}(a_2^{\mathsf{w}})+\beta\cdot\mathcal{C}(a_1^{\mathsf{b}})+(y-\beta)\cdot\mathcal{C}(a_2^{\mathsf{b}})<$$
$$x\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-x)\cdot\mathcal{C}(a_2^{\mathsf{w}})+y\cdot\mathcal{C}(a_1^{\mathsf{b}})$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}))<(\alpha-x)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})) \tag{131}$$

From Eq. 126 we have $y-\beta\geq\alpha-x$, together with Eq. 130 contradicting Eq. 131.

(4) If $|\rho_v|=2y$, $|\sigma_w|=2x$, $|\sigma_w|>|\rho_v|+1$, then the actions of ρ_v^* are set to

$$a_i^*=\begin{cases}a_{\mathsf{w}_v|\mathsf{w}_w} & i=2j-1, j\in\mathbb{N}\\ a_{\mathsf{b}_v|\mathsf{b}_w} & \text{otherwise}\end{cases} \tag{132}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^*=\langle a_1,\ldots,a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 111 above.

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w'\in\mathsf{pred}(v)\setminus\{u,w\}$ are set to empty sets.

The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v\cup>_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 111 we can now derive that any linearization of $>_v\cup\bigcup_{w\in\mathsf{pred}(v)}>_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\}\cup\mathsf{pred}(v)$. In addition, $|\rho_v^*|=|\rho{\downarrow_v}|$ together with Eq. 132 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(5) If $|\rho_v|=2y+1$, $|\sigma_w|=2x+1$, $|\sigma_w|\leq|\rho_v|+1$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^*=\begin{cases}a_1^{\mathsf{b}} & i=2j, j\in\mathbb{N}, j<x\\ a_2^{\mathsf{b}} & i=2j, j\in\mathbb{N}, x\leq j\leq y\\ a_1^{\mathsf{w}} & \text{otherwise}\end{cases} \tag{133}$$

And the cost in this case is

$$(y+1) \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (x-1) \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x+1) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{134}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 135.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i < j < 2x\} \cup \{a_j < a_i^* \mid j \leq i, j < 2x\} \cup \{a_i^* < a_{2x}\} \tag{135}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 135 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 61, 135 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 133 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining changes to w_v being done using action a_2^{w}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, j \leq y-x \text{ or } j = y+1 \\ a_1^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, y-x < j \leq y \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{136}$$

And the cost in this case is

$$x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{137}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 138.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i < 2y - 2x + j\} \cup \{a_j < a_i^* \mid i \geq 2y - 2x + j\} \tag{138}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65.
It is easy to verify that the relation $>_{v,w}$ defined by Eq. 138 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 65, 138 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 136 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y+x+1$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y+x+2$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x-1$ actions of other types. Let $\mathsf{w}_w \cdot \mathsf{b}_w \cdot \ldots \cdot \mathsf{w}_w$ sequence of $2y+1$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x-1$ actions of other type will decrease the length in at most $2y-2x-2$, and we are left with the sequence of length $\geq 2y+1-(2y-2x-2) = 2x+3$. Therefore σ_w cannot support more than $y+x+1$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha+\beta \leq y+x+1 \tag{139}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{140}$$

For (134) $\leq$ (137), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{141}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 140 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &(y+1)\cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x)\cdot \mathcal{C}(a_2^{\mathsf{b}})\end{aligned}$$

and from it

$$(y+1-\alpha)\cdot(\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) < (\beta - x)\cdot(\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) \tag{142}$$

From Eq. 139 we have $y+1-\alpha \geq \beta - x$, together with Eq. 141 contradicting Eq. 142.
For (137) $\leq$ (134), we have

$$\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \leq \mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \tag{143}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 140 we have

$$\begin{aligned}&\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y+1-\alpha)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta)\cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &(x+1 \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x)\cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}})\end{aligned}$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}})) < (\alpha - x - 1)\cdot(\mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}})) \tag{144}$$

From Eq. 139 we have $y-\beta \geq \alpha - x - 1$, together with Eq. 143 contradicting Eq. 144.

(6) If $|\rho_v| = 2y+1$, $|\sigma_w| = 2x+1$, $|\sigma_w| > |\rho_v|+1$, then the actions of ρ_v^* are set to

$$a_i^* = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_w} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\mathsf{b}_w} & \text{otherwise} \end{cases} \tag{145}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 111 above.

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets.

The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

From Eqs. 54, 111 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 145 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(7) If $|\rho_v| = 2y$, $|\sigma_w| = 2x+1$, $|\sigma_w| \leq |\rho_v|$, then we construct two post-3/2 candidates for ρ_v^*, and then assign ρ_v^* to the cheapest among the two, proving that its cost has to be lower than $\mathcal{C}(\rho{\downarrow_v})$.

(a) *All the changes of v to w_v are done using action a_1^{w}, and then the maximally possible number of changes to b_v are done using action a_1^{b}, with the remaining changes to b_v being done using action a_2^{b}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, j \leq y-x \\ a_1^{\mathsf{b}} & i = 2j, j \in \mathbb{N}, y-x < j \leq y \\ a_1^{\mathsf{w}} & \text{otherwise} \end{cases} \tag{146}$$

And the cost in this case is

$$y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{147}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \ldots, a_{2x} \rangle$, we set $>_{v,w}$ according to Eq. 148.

$$>_{v,w} = \bigcup_{a_i^* \in \rho_v^*, a_j \in \rho_w^*} \{a_i^* < a_j \mid i < 2y-2x+j, j > 1\} \cup \{a_j < a_i^* \mid i \geq 2y-2x+j\} \cup \{a_1 < a_i^*\} \tag{148}$$

For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 61.

It is easy to verify that the relation $>_{v,w}$ defined by Eq. 148 is a strict total order over its domain. The relations $>_v$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.

From Eqs. 54, 61, 148 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$. In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 146 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

(b) *All the changes of v to b_v are done using action a_1^{b}, and then the maximally possible number of changes to w_v are done using action a_1^{w}, with the remaining changes to w_v being done using action a_2^{w}.* For this candidate for ρ_v^*, we set

$$a_i^* = \begin{cases} a_2^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, j \leq y-x \\ a_1^{\mathsf{w}} & i = 2j-1, j \in \mathbb{N}, y-x < j \leq y \\ a_1^{\mathsf{b}} & \text{otherwise} \end{cases} \tag{149}$$

And the cost in this case is

$$x \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + y \cdot \mathcal{C}(a_1^{\mathsf{b}}) \tag{150}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_{2x-1} \rangle$, we set $>_{v,w}$ according to Eq. 138 above.
For each $u \in \mathsf{pred}(v) \setminus \{w\}$ we set $>_{v,u}$ according to Eq. 65 above.
The relations $>_v$, $>_{v,w}$ and $>_{v,u}$ are effectively identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,u}$ and $>_v \cup >_{v,w}$ forming strict partial orders over the unions of the elements of ρ_v^* and ρ_u^*, and ρ_v^* and ρ_w^*, respectively.
From Eqs. 54, 65, 138 we can now derive that any linearization of $>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w}$ defines a sequence of actions that is applicable with respect to $\{v\} \cup \mathsf{pred}(v)$.
In addition, $|\rho_v^*| = |\rho{\downarrow_v}|$ together with Eq. 149 implies that this action sequence provides to v the value $G[v]$ if the latter is specified.

Now, for each cost-optimal plan ρ, $\rho{\downarrow_v}$ cannot contain more than $y+x$ actions of both types a_1^{w} and a_1^{b} totally. Suppose to the contrary that $\rho{\downarrow_v}$ contain at least $y+x+1$ actions of types a_1^{w} and a_1^{b}. Then it contains no more than $y-x-1$ actions of other types. Let $\mathsf{w}_w \cdot \mathsf{b}_w \cdot \ldots \cdot \mathsf{b}_w$ sequence of $2y$ values of w that support cost-optimal plan for v given that w can change it value any number of times. Then each action of other type will decrease the needed length of this sequence in at most 2. Therefore at most $y-x-1$ actions of other type will decrease the length in at most $2y-2x-2$, and we are left with the sequence of length $\geq 2y-(2y-2x-2) = 2x+2$. Therefore σ_w cannot support more than $y+x$ actions of types a_1^{w} and a_1^{b}. Now, suppose that in some given cost-optimal plan $\rho{\downarrow_v}$ for v there are α actions of type a_1^{w} and β actions of type a_1^{b}. Then

$$\alpha + \beta \leq y + x \tag{151}$$

and

$$\mathcal{C}(\rho_v) \geq \alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \tag{152}$$

For (147) $\leq$ (150), we have

$$\mathcal{C}(a_2^{\mathsf{b}}) - \mathcal{C}(a_1^{\mathsf{b}}) \leq \mathcal{C}(a_2^{\mathsf{w}}) - \mathcal{C}(a_1^{\mathsf{w}}) \tag{153}$$

Now suppose to the contrary that the plan in first case is not cost-optimal. Then from Eq. 152 we have

$$\begin{aligned} &\alpha \cdot \mathcal{C}(a_1^{\mathsf{w}}) + (y-\alpha) \cdot \mathcal{C}(a_2^{\mathsf{w}}) + \beta \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-\beta) \cdot \mathcal{C}(a_2^{\mathsf{b}}) < \\ &y \cdot \mathcal{C}(a_1^{\mathsf{w}}) + x \cdot \mathcal{C}(a_1^{\mathsf{b}}) + (y-x) \cdot \mathcal{C}(a_2^{\mathsf{b}}) \end{aligned}$$

and from it

$$(y-\alpha)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}}))<(\beta-x)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}})) \tag{154}$$

From Eq. 151 we have $y-\alpha \geq \beta-x$, together with Eq. 153 contradicting Eq. 154. For (150) $\leq$ (147), we have

$$\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}}) \leq \mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}) \tag{155}$$

Now suppose to the contrary that the plan in second case is not cost-optimal. Then from Eq. 152 we have

$$\alpha\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-\alpha)\cdot\mathcal{C}(a_2^{\mathsf{w}})+\beta\cdot\mathcal{C}(a_1^{\mathsf{b}})+(y-\beta)\cdot\mathcal{C}(a_2^{\mathsf{b}})<$$
$$x\cdot\mathcal{C}(a_1^{\mathsf{w}})+(y-x)\cdot\mathcal{C}(a_2^{\mathsf{w}})+y\cdot\mathcal{C}(a_1^{\mathsf{b}})$$

and from it

$$(y-\beta)\cdot(\mathcal{C}(a_2^{\mathsf{b}})-\mathcal{C}(a_1^{\mathsf{b}}))<(\alpha-x)\cdot(\mathcal{C}(a_2^{\mathsf{w}})-\mathcal{C}(a_1^{\mathsf{w}})) \tag{156}$$

From Eq. 151 we have $y-\beta \geq \alpha-x$, together with Eq. 155 contradicting Eq. 156.

(8) If $|\rho_v| = 2y$, $|\sigma_w| = 2x+1$, $|\sigma_w| > |\rho_v|$, then the actions of ρ_v^* are set to

$$a_i^* = \begin{cases} a_{\mathsf{w}_v|\mathsf{w}_w} & i = 2j-1, j \in \mathbb{N} \\ a_{\mathsf{b}_v|\mathsf{b}_w} & \text{otherwise} \end{cases} \tag{157}$$

Here as well, the ordering constraints $>_v$ are set according to Eq. 54. Likewise, if $\rho_w^* = \langle a_1, \dots, a_{2x-1}\rangle$, we set $>_{v,w}$ according to Eq. 111 above.

Finally, here as well, the ordering constraints $>_{v,w'}$ for the rest of the parents $w' \in \mathsf{pred}(v) \setminus \{u, w\}$ are set to empty sets.

The relations $>_v$ and $>_{v,w}$ are identical to the previous case. Thus, here as well, we have $>_v \cup >_{v,w}$ forming strict partial order over the union of the elements of ρ_v^* and ρ_w^*.

Until now, we have specified the sequences ρ_v^*, the orders $>_v$ induced by these sequences, the orders $>_{v,w}$, and proved that all $>_v \cup >_{v,w}$ and $>_w \cup >_{v,w}$ form strict partial orders over their domains. This construction allows us to apply now Theorem 1 to the (considered as sets) sequences ρ_v^* and orders $>_v$ and $>_{v,w}$, proving that

$$>= \bigcup_{v\in V}(>_v \cup \bigcup_{w\in\mathsf{pred}(v)} >_{v,w})$$

forms a strict partial order over the union of $\rho_{v_1}^*, \dots, \rho_{v_n}^*$. Putting thing together, the above implies that any linearization ρ^* of $>$ is a plan for Π, and **post-3/2**ness of all its subsequences $\rho_{v_1}^*, \dots, \rho_{v_n}^*$ then implies $\rho^* \in \mathcal{P}^{3/2}(\Pi)$. Moreover, if ρ is an optimal plan for Π, then $\mathcal{C}(\rho^*) = \mathcal{C}(\rho)$ implies the optimality of ρ^*. ∎

5.2 Construction

Given a post-$\mathbf{3/2}$ action sequence ϱ from A and a variable $v \in V$, we can distinguish between the following roles of parent $w \in \mathsf{pred}(v)$ with respect to v along ϱ.

R1 All the actions in ϱ that change the value of v are supported by the same value of w.
That is, for some $\gamma \in \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\gamma}, a_{\mathsf{w}_v|\gamma}\}$.

R2 All the actions in ϱ that change the value of v to w_v are supported by the same value of w, and all the actions in ϱ that change the value of v to b_v are supported by another value of w.
That is, for some $\gamma \neq \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\gamma}, a_{\mathsf{w}_v|\delta}\}$.

R3 All the actions in ϱ that change the value of v to w_v are supported by the same value of w, and none of the actions in ϱ that change the value of v to b_v are supported by w.
That is, for some $\gamma \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\delta \notin \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\delta}, a_{\mathsf{w}_v|\gamma}\}$.

R4 All the actions in ϱ that change the value of v to b_v are supported by the same value of w, and none of the actions in ϱ that change the value of v to w_v are supported by w.
That is, for some $\gamma \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\delta \notin \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\gamma}, a_{\mathsf{w}_v|\delta}\}$.

R5 All the actions in ϱ that change the value of v to w_v are supported by the same value of w, and all the actions in ϱ that change the value of v to b_v are supported by two values of w.
That is, for some $\gamma \neq \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{w}_v|\gamma}, a_{\mathsf{b}_v|\delta}, a_{\mathsf{b}_v|\gamma}\}$.

R6 All the actions in ϱ that change the value of v to b_v are supported by the same value of w, and all the actions in ϱ that change the value of v to w_v are supported by two values of w.
That is, for some $\gamma \neq \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\gamma}, a_{\mathsf{w}_v|\delta}, a_{\mathsf{w}_v|\gamma}\}$.

R7 All the actions in ϱ that change the value of v to w_v are supported by the same value of w, and some of the actions in ϱ that change the value of v to b_v are supported by another value of w and others are supported by another parent.
That is, for some $\gamma \neq \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\mu \notin \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{w}_v|\gamma}, a_{\mathsf{b}_v|\delta}, a_{\mathsf{b}_v|\mu}\}$.

R8 All the actions in ϱ that change the value of v to b_v are supported by the same value of w, and some of the actions in ϱ that change the value of v to w_v are supported by another value of w and others are supported by another parent.
That is, for some $\gamma \neq \delta \in \{\mathsf{b}_w, \mathsf{w}_w\}$ and $\mu \notin \{\mathsf{b}_w, \mathsf{w}_w\}$, if $a \in \varrho_v$, then $a \in \{a_{\mathsf{b}_v|\gamma}, a_{\mathsf{w}_v|\delta}, a_{\mathsf{w}_v|\mu}\}$.

R9 Part of the actions in ϱ that change the value of v to b_v are supported by the same value of w, and none of the actions in ϱ that change the value of v to w_v are supported by the same value of w.

R10 Part of the actions in ϱ that change the value of v to w_v are supported by the same value of w, and none of the actions in ϱ that change the value of v to b_v are supported by the same value of w.

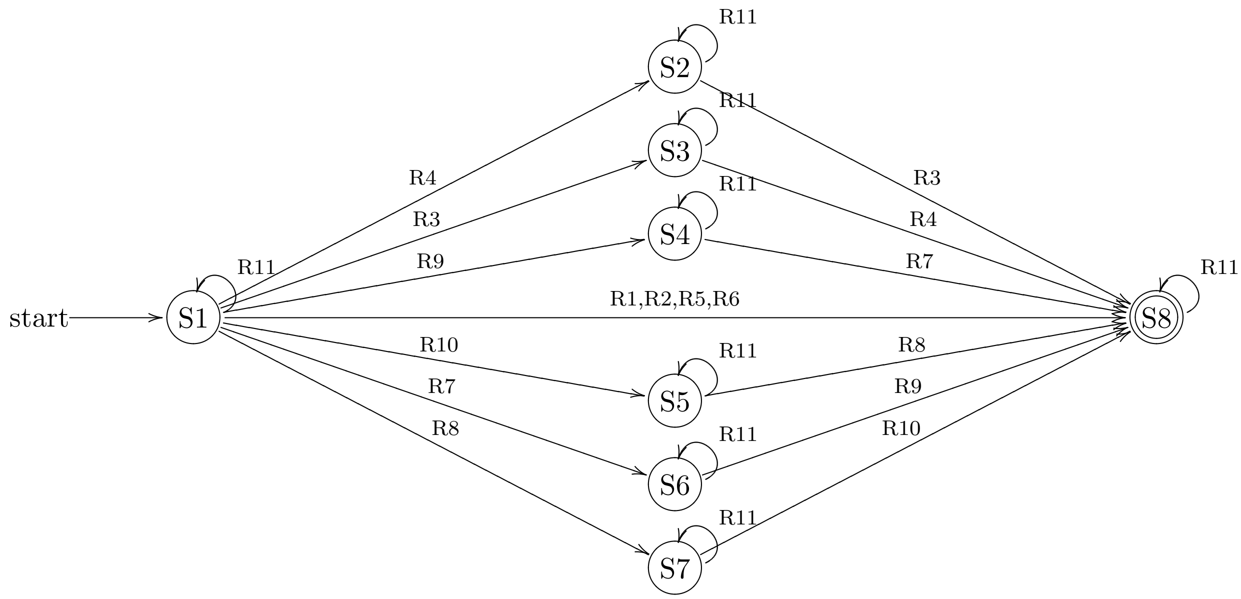


Figure 10: State machine describing the process of role assignment to the parents of v (with respect to v). Each transition is labeled with a set of roles, one of whose is getting assigned to a parent of v at the corresponding step.

R11 None of the actions in ϱ are supported by w.
That is, if $a_{\alpha|\gamma} \in \varrho$, then $\gamma \notin \{\mathsf{b}_w, \mathsf{w}_w\}$.

For a given post-**3/2** action sequence ϱ from A and a variable $v \in V$, each parent of v performs one of the roles R1-R11 with respect to v along ϱ, and each of the roles R1-R10 is performed by at most one of the parents of v. In addition, there are sets of roles that cannot be simultaneously performed by the parents of v with respect to v and the same action sequence ϱ, and there are roles that have to be performed in pairs. Specifically,

- If one of the roles {R1,R2,R5,R6} is played by some parent $w' \in \mathsf{pred}(v)$, then R11 must be played by all other parents $w \in \mathsf{pred}(v) \setminus \{w'\}$.
- If R3/R7/R8 is played by some parent $w_1 \in \mathsf{pred}(v)$, then R4/R9/R10, respectively, must be played by some parent $w_2 \in \mathsf{pred}(v) \setminus \{w_1\}$, and R11 must be played by all other parents $w \in \mathsf{pred}(v) \setminus \{w_1, w_2\}$.

Considering a variable v and its parents $\mathsf{pred}(v)$ through the lens of these eleven roles, suppose we now aim at assigning these roles to $\mathsf{pred}(v)$ by considering them one after another in some arbitrary order. Given the aforementioned constraints on the role assignment, at each step of this sequential process we can be in one of the following eight states, with the whole process being described by a state machine depicted in Fig. 10.

S1 All the roles R1-R11 are still available (to be assigned to the parents of v).

S2 Only roles {R3,R11} are still available.

S3 Only roles {R4,R11} are available.

S4 Only roles {R7,R11} are available.

S5 Only roles {R8,R11} are available.

S6 Only roles {R9,R11} are available.

S7 Only roles {R10,R11} are available.

S8 Only role R11 is available.

Given this language of "roles" and "states", we now proceed with specifying our constraint optimization problem $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ for a problem $\Pi = (V, A, I, G) \in \mathbf{P}(1)$. In what follows, for each variable $v \in V$, we assume a fixed (*arbitrary* chosen) numbering $\{w_1, \ldots, w_k\}$ of $\mathsf{pred}(v)$ with respect to v.

1. Similarly to the uniform-cost case, the variable set $\mathcal{X}$ contains a variable x_v for each planning variable $v \in V$, and a variable x_v^w for each edge $(w, v) \in CG(\Pi)$. That is,

$$\begin{aligned}\mathcal{X} &= \mathcal{X}^V \cup \mathcal{X}^{\mathcal{E}} \\ \mathcal{X}^V &= \{x_v \mid v \in V\} \\ \mathcal{X}^{\mathcal{E}} &= \{x_v^w \mid (w, v) \in CG(\Pi)\}\end{aligned} \tag{158}$$

2. For each variable $x_v \in \mathcal{X}^V$, the domain $Dom(x_v)$ consists of all possible valid prefixes of $\sigma(v)$. For each variable $x_v^{w_i} \in \mathcal{X}^{\mathcal{E}}$, the domain $Dom(x_v^{w_i})$ consists of all possible quadruples satisfying Eq. 159.

$$\begin{aligned}Dom(x_v) &= \{\sigma_v \in \trianglerighteq^*[\sigma(v)]\} \\ Dom(x_v^{w_i}) &= \left\{ [\![\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta]\!] \;\middle|\; \begin{array}{l} 0 \leq \eta \leq n, 0 \leq \#_{\mathsf{w}}, \#_{\mathsf{b}} \leq \lceil \frac{\eta}{2} \rceil \\ \mathbf{S} \in \{\mathrm{S}_1, \ldots, \mathrm{S}_8\} \end{array} \right\}\end{aligned} \tag{159}$$

The semantics of Eq. 159 is as follows. Let $\{w_1, \ldots, w_k\}$ be an *arbitrary* fixed ordering of $\mathsf{pred}(v)$. If x_v takes the value $\sigma_v \in Dom(x_v)$, then v is forced to provide σ_v sequence of values. In turn, if $x_v^{w_i}$ takes the value $[\![\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta]\!]$, then η corresponds to the number of value changes of v, $\#_{\mathsf{w}}$ and $\#_{\mathsf{b}}$ correspond to the number of value changes of v to w_v and b_v, respectively, that should be performed by the actions prevailed by the values of $\{w_1, \ldots, w_i\}$, and the state-component $\mathbf{S}$ captures the roles that can be assigned to the parents $\{w_1, \ldots, w_i\}$.

3. Similarly to the uniform-cost case, for each variable $x \in \mathcal{X}$, the set $\mathcal{F}$ contains a non-negative, real-valued function φ_x with the scope

$$Q_x = \begin{cases} \{x_v\}, & x = x_v, k = 0 \\ \{x_v, x_v^{w_k}\}, & x = x_v, k > 0 \\ \{x_v^{w_1}, x_{w_1}\}, & x = x_v^{w_1}, k > 0 \\ \{x_v^{w_j}, x_v^{w_{j-1}}, x_{w_j}\}, & x = x_v^{w_j}, 1 < j \leq k \end{cases} \tag{160}$$

where $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$ (with $k = 0$ meaning $\mathsf{pred}(v) = \emptyset$).

Proceeding now with specifying the functional components $\mathcal{F}$ of COP_Π, first, for each x_v with $\mathsf{pred}(v) = \emptyset$, and for each $\sigma_v \in \trianglerighteq^*[v]$, we set $\varphi_{x_v}(\sigma_v)$ according to Eq. 161.

$$\varphi_{x_v}(\sigma_v) = \begin{cases} 0, & |\sigma_v| = 1, \\ \mathcal{C}(a_{\mathsf{w}_v}), & |\sigma_v| = 2, a_{\mathsf{w}_v} \in A_v, \\ \lceil \frac{|\sigma_v|-1}{2} \rceil \cdot \mathcal{C}(a_{\mathsf{w}_v}) + \lfloor \frac{|\sigma_v|-1}{2} \rfloor \cdot \mathcal{C}(a_{\mathsf{b}_v}), & |\sigma_v| > 2, a_{\mathsf{w}_v}, a_{\mathsf{b}_v} \in A_v, \\ \infty, & \text{otherwise} \end{cases} \tag{161}$$

In turn, for each planning variable $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, $k > 0$, the function φ_{x_v} is set by Eq. 162.

$$\varphi_{x_v}(\sigma_v, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!]) = \begin{cases} 0, & |\sigma_v| = 1, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!] = [\![\mathrm{S8}, 0, 0, 0]\!], \\ 0, & |\sigma_v| > 1, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!] = \left[\!\!\left[\mathrm{S1}, \lceil \frac{|\sigma_v|-1}{2} \rceil, \lfloor \frac{|\sigma_v|-1}{2} \rfloor, |\sigma_v| - 1\right]\!\!\right], \\ \infty, & \text{otherwise} \end{cases} \tag{162}$$

Now, we proceed with specifying a function φ that, for each $v \in V$, each $w \in \mathsf{pred}(v)$, and each $(\mathrm{R}, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!], \sigma_w) \in \{\mathrm{R1}, \ldots, \mathrm{R10}\} \times Dom(x_v^w) \times Dom(x_w)$, provides the marginal over the actions A_v cost of w taking the role R, and under this role, supporting $\#_\mathsf{w}$ changes of v to w_v and $\#_\mathsf{b}$ changes of v to b_v, out of total η changes of v needed. For ease of presentation, let $\xi(x_1, x_2, y_1, y_2)$ denote the cost of an action sequence with x_1 actions of type $a_{\mathsf{w}_v|\mathsf{b}_w}$, x_2 actions of type $a_{\mathsf{w}_v|\mathsf{w}_w}$, y_1 actions of type $a_{\mathsf{b}_v|\mathsf{w}_w}$, y_2 actions of type $a_{\mathsf{b}_v|\mathsf{b}_w}$, that is

$$\xi(x_1, x_2, y_1, y_2) = x_1 \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) + x_2 \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}) + y_1 \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{w}_w}) + y_2 \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{b}_w}) \tag{163}$$

While the notation $\xi_{v,w}$ is probably more appropriate for the semantics of ξ, we adopt the latter for its shortness because the identity of v and w will always be clear from the context.

$$\varphi(\mathrm{R1}, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!], \sigma_w) = \begin{cases} \xi(\#_\mathsf{w}, 0, 0, \#_\mathsf{b}), & |\sigma_w| = 1, \#_\mathsf{w} = \lceil \frac{\eta}{2} \rceil, \#_\mathsf{b} = \lfloor \frac{\eta}{2} \rfloor \\ \min \left\{ \begin{array}{c} \xi(\#_\mathsf{w}, 0, 0, \#_\mathsf{b}), \\ \xi(0, \#_\mathsf{w}, \#_\mathsf{b}, 0) \end{array} \right\}, & |\sigma_w| > 1, \#_\mathsf{w} = \lceil \frac{\eta}{2} \rceil, \#_\mathsf{b} = \lfloor \frac{\eta}{2} \rfloor \\ \infty, & \text{otherwise} \end{cases} \tag{164}$$

$$\varphi(\mathrm{R2}, [\![\mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta]\!], \sigma_w) = \begin{cases} \xi(\#_\mathsf{w}, 0, \#_\mathsf{b}, 0), & |\sigma_w| = \eta \geq 2, \#_\mathsf{w} = \lceil \frac{\eta}{2} \rceil, \#_\mathsf{b} = \lfloor \frac{\eta}{2} \rfloor \\ \min \left\{ \begin{array}{c} \xi(\#_\mathsf{w}, 0, \#_\mathsf{b}, 0), \\ \xi(0, \#_\mathsf{w}, 0, \#_\mathsf{b}) \end{array} \right\}, & |\sigma_w| > \eta \geq 2, \#_\mathsf{w} = \lceil \frac{\eta}{2} \rceil, \#_\mathsf{b} = \lfloor \frac{\eta}{2} \rfloor \\ \infty, & \text{otherwise} \end{cases} \tag{165}$$

$$\varphi(\mathrm{R3}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}), & |\sigma_w| = 1, \#_{\mathsf{w}} = \lceil \frac{\eta}{2} \rceil, \#_{\mathsf{b}} = 0 \\ \min \left\{ \begin{array}{c} \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}), \\ \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}) \end{array} \right\}, & |\sigma_w| > 1, \#_{\mathsf{w}} = \lceil \frac{\eta}{2} \rceil, \#_{\mathsf{b}} = 0 \\ \infty, & \text{otherwise} \end{cases} \tag{166}$$

$$\varphi(\mathrm{R4}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{b}_w}), & |\sigma_w| = 1, \#_{\mathsf{w}} = 0, \#_{\mathsf{b}} = \lfloor \frac{\eta}{2} \rfloor \\ \min \left\{ \begin{array}{c} \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{b}_w}), \\ \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{w}_w}) \end{array} \right\}, & |\sigma_w| > 1, \#_{\mathsf{w}} = 0, \#_{\mathsf{b}} = \lfloor \frac{\eta}{2} \rfloor \\ \infty, & \text{otherwise} \end{cases} \tag{167}$$

$$\varphi(\mathrm{R5}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \min \left\{ \begin{array}{l} \xi(y+1, 0, x-1, y-x+1), \\ \xi(0, y+1, y-x+1, x-1) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x, 1 < x \leq y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \min \left\{ \begin{array}{l} \xi(y, 0, x, y-x), \\ \xi(0, y, y-x+1, x-1) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x, 1 < x < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \xi(y, 0, 1, y-1), & \begin{array}{l} \eta = 2y, |\sigma_w| = 2, 1 < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \xi(0, y, 1, y-1), & \begin{array}{l} \eta = |\sigma_w| = 2y, 1 < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \min \left\{ \begin{array}{l} \xi(y+1, 0, x, y-x), \\ \xi(0, y+1, y-x+1, x-1) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x+1, 1 < x < y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \xi(y+1, 0, 1, y-1), & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 3, 1 < y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \xi(0, y+1, 1, y-1), & \begin{array}{l} \eta = |\sigma_w| = 2y+1, 1 < y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \min \left\{ \begin{array}{l} \xi(y, 0, x, y-x), \\ \xi(0, y, y-x, x) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x+1, 1 \leq x < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \infty, & \text{otherwise} \end{cases} \tag{168}$$

$$\varphi(\mathrm{R6}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \min\left\{ \begin{array}{l} \xi(x, y+1-x, y, 0), \\ \xi(y+1-x, x, 0, y) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x, 1 \leq x \leq y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \min\left\{ \begin{array}{l} \xi(x, y-x, y, 0), \\ \xi(y-x+1, x-1, 0, y) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x, 1 < x < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \xi(1, y-1, y, 0), & \begin{array}{l} \eta = 2y, |\sigma_w| = 2, 1 < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \xi(1, y-1, 0, y), & \begin{array}{l} \eta = |\sigma_w| = 2y, 1 < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \min\left\{ \begin{array}{l} \xi(x+1, y-x, y, 0), \\ \xi(y-x+1, x, 0, y) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x+1, 1 \leq x < y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \xi(1, y, 0, y), & \begin{array}{l} \eta = |\sigma_w| = 2y+1, 1 \leq y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = y \end{array} \\ \min\left\{ \begin{array}{l} \xi(x, y-x, y, 0), \\ \xi(y-x, x, 0, y) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x+1, 1 \leq x < y, \\ \#_{\mathsf{w}} = \#_{\mathsf{b}} = y \end{array} \\ \infty, & \text{otherwise} \end{cases} \tag{169}$$

$$\varphi(\text{R7}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \min\left\{ \begin{array}{l} \xi(y+1,0,x-1,0), \\ \xi(0,y+1,0,x-1) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x, 1 < x \leq y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = x-1 \end{array} \\ \\ \xi(y,0,x,0), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) < \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y, |\sigma_w| = 2x, 1 \leq x < y, \\ \#_{\mathsf{w}} = y, \#_{\mathsf{b}} = x \end{array} \\ \\ \xi(0,y,0,x-1), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) \geq \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y, |\sigma_w| = 2x, 1 < x \leq y, \\ \#_{\mathsf{w}} = y, \#_{\mathsf{b}} = x-1 \end{array} \\ \\ \xi(y+1,0,x,0), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) < \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y+1, |\sigma_w| = 2x+1, 1 \leq x < y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = x \end{array} \\ \\ \xi(0,y+1,0,x-1), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) \geq \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y+1, |\sigma_w| = 2x+1, 1 < x \leq y, \\ \#_{\mathsf{w}} = y+1, \#_{\mathsf{b}} = x-1 \end{array} \\ \\ \min\left\{ \begin{array}{l} \xi(y,0,x,0), \\ \xi(0,y,0,x) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x+1, 1 \leq x < y, \\ \#_{\mathsf{w}} = y, \#_{\mathsf{b}} = x \end{array} \\ \\ \infty, & \text{otherwise} \end{cases} \tag{170}$$

$$\varphi(\text{R8}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \min\left\{ \begin{array}{c} \xi(x,0,y,0), \\ \xi(0,x,0,y) \end{array} \right\}, & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x, 1 \le x \le y, \\ \#_{\mathsf{w}} = x, \#_{\mathsf{b}} = y \end{array} \\ \\ \xi(x,0,y,0), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) < \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y, |\sigma_w| = 2x, 1 \le x < y, \\ \#_{\mathsf{w}} = x, \#_{\mathsf{b}} = y \end{array} \\ \\ \xi(0,x-1,0,y), & \begin{array}{l} \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}) \ge \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}), \\ \eta = 2y, |\sigma_w| = 2x, 1 < x \le y, \\ \#_{\mathsf{w}} = x-1, \#_{\mathsf{b}} = y \end{array} \\ \\ \xi(x+1,0,y,0), & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x+1, 1 \le x < y, \\ \#_{\mathsf{w}} = x+1, \#_{\mathsf{b}} = y \end{array} \\ \\ \xi(0,x,0,y), & \begin{array}{l} \eta = 2y+1, |\sigma_w| = 2x+1, 1 \le x \le y, \\ \#_{\mathsf{w}} = x, \#_{\mathsf{b}} = y \end{array} \\ \\ \min\left\{ \begin{array}{c} \xi(x,0,y,0), \\ \xi(0,x,0,y) \end{array} \right\}, & \begin{array}{l} \eta = 2y, |\sigma_w| = 2x+1, 1 \le x < y, \\ \#_{\mathsf{w}} = x, \#_{\mathsf{b}} = y \end{array} \\ \\ \infty, & \text{otherwise} \end{cases} \tag{171}$$

$$\varphi(\text{R9}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{w}_w}), & |\sigma_w| = 1, \#_{\mathsf{w}} = 0, \#_{\mathsf{b}} < \lfloor \frac{\eta}{2} \rfloor \\ \min\left\{ \begin{array}{c} \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{w}_w}), \\ \#_{\mathsf{b}} \cdot \mathcal{C}(a_{\mathsf{b}_v|\mathsf{b}_w}) \end{array} \right\}, & |\sigma_w| > 1, \#_{\mathsf{w}} = 0, \#_{\mathsf{b}} < \lfloor \frac{\eta}{2} \rfloor \\ \infty, & \text{otherwise} \end{cases} \tag{172}$$

$$\varphi(\text{R10}, [\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta], \sigma_w) = \begin{cases} \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}), & |\sigma_w| = 1, \#_{\mathsf{w}} < \lceil \frac{\eta}{2} \rceil, \#_{\mathsf{b}} = 0 \\ \min\left\{ \begin{array}{c} \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{b}_w}), \\ \#_{\mathsf{w}} \cdot \mathcal{C}(a_{\mathsf{w}_v|\mathsf{w}_w}) \end{array} \right\}, & |\sigma_w| > 1, \#_{\mathsf{w}} < \lceil \frac{\eta}{2} \rceil, \#_{\mathsf{b}} = 0 \\ \infty, & \text{otherwise} \end{cases} \tag{173}$$

Having specified the function φ, we now use it, in particular, for specifying the functional component $\varphi_{x_v^{w_1}}$ as in Eq. 174. This equation actually emulates movements in the state machine for v as in Figure 10 to the terminal state S8.

$$\varphi_{x_v^{w_1}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}) = \begin{cases} \min \left\{ \begin{array}{l} \varphi(\mathrm{R1}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), \\ \varphi(\mathrm{R2}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), \\ \varphi(\mathrm{R5}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), \\ \varphi(\mathrm{R6}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}) \end{array} \right\}, & \mathbf{S} = \mathrm{S1}, \\ \varphi(\mathrm{R3}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S2}, \\ \varphi(\mathrm{R4}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S3}, \\ \varphi(\mathrm{R7}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S4}, \\ \varphi(\mathrm{R8}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S5}, \\ \varphi(\mathrm{R9}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S6}, \\ \varphi(\mathrm{R10}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}), & \mathbf{S} = \mathrm{S7}, \\ 0, & \begin{array}{l} \mathbf{S} = \mathrm{S8}, \\ \#_{\mathsf{w}} = 0, \\ \#_{\mathsf{b}} = 0 \end{array} \\ \infty, & \text{otherwise} \end{cases} \tag{174}$$

We now proceed with the rest of the functional components $\varphi_{x_v^{w_2}}, \ldots, \varphi_{x_v^{w_k}}$. For each $2 \leq j \leq k$, the value of $\varphi_{x_v^{w_j}}$ for each $\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket \in \mathit{Dom}(x_v^{w_j})$, each $\llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket \in \mathit{Dom}(x_v^{w_{j-1}})$, and each $\sigma_w \in \mathit{Dom}(x_{w_j}) = \trianglerighteq^*[w_j]$, is set according to Eq. 175. This equation actually emulates movements in the state machine for v as in Figure 10—each sub-case of Eq. 175 deals with a certain transition in that state machine.

$$\varphi_{x_v^{w_j}}\left(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}\right) =$$

$$\begin{cases}
\min\left\{\begin{array}{l}
\varphi(\mathrm{R1}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), \\
\varphi(\mathrm{R2}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), \\
\varphi(\mathrm{R5}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), \\
\varphi(\mathrm{R6}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})
\end{array}\right\}, & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R4}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S2}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R3}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S3}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R9}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S4}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R10}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S5}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R7}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S6}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R8}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S1}, \mathbf{S}' = \mathrm{S7}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R3}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S2}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R4}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S3}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R7}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S4}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R8}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S5}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R9}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S6}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
\varphi(\mathrm{R10}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}), & \begin{array}{l}\mathbf{S} = \mathrm{S7}, \mathbf{S}' = \mathrm{S8}, \eta = \eta', \\ \#_{\mathsf{w}} \geq \#'_{\mathsf{w}}, \#_{\mathsf{b}} \geq \#'_{\mathsf{b}}\end{array} \\
0, & \begin{array}{l}\mathbf{S} = \mathbf{S}', \eta = \eta', \\ \#_{\mathsf{w}} = \#'_{\mathsf{w}}, \#_{\mathsf{b}} = \#'_{\mathsf{b}}\end{array} \\
\infty, & \text{otherwise}
\end{cases} \tag{175}$$

procedure polytree-1-dep($\Pi = (V, A, I, G)$)
takes a problem $\Pi \in \mathbf{P}(1)$
returns a cost-optimal plan for Π if Π is solvable, and fails otherwise
create a set of variables $\mathcal{X}$ as in Eqs. 158-159
create a set of functions $\mathcal{F} = \{\varphi_x \mid x \in \mathcal{X}\}$ with scopes as in Eq. 160
for each $x \in \mathcal{X}$ **do**
  specify φ_x according to Eqs. 161-175
endfor
set $\mathsf{COP}_\Pi := (\mathcal{X}, \mathcal{F})$ with global objective $\min \sum_{\varphi \in \mathcal{F}} \varphi(\mathcal{X})$
$\overline{x} :=$ solve-tree-cop(COP_Π)
if $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \infty$ **then return** failure
extract plan ρ from $\overline{x}$ with $\mathcal{C}(\rho) = \sum_{\varphi \in \mathcal{F}} \varphi(\overline{x})$
return ρ

Figure 11: Algorithm for cost-optimal planning for $\mathbf{P}(1)$ problems.

This finalizes the construction of COP_Π, and this construction constitutes the first three steps of the algorithm polytree-1-dep in Figure 11(a). The subsequent steps of this algorithm are conceptually similar to these of the polytree-k-dep-uniform algorithm in Section 4. It is not hard to verify from Eqs. 158-160, and the fact that the causal graph of $\Pi \in \mathbf{P}(1)$ forms a polytree that

(i) for each variable $x \in \mathcal{X}$, $|Dom(x)| = poly(n)$,

(ii) the tree-width of the cost network of $\mathcal{F}$ is ≤ 3, and

(iii) the optimal tree-decomposition of the COP_Π's cost network is given by any topological ordering of the causal graph that is consistent with the (arbitrary yet fixed at the time of the COP_Π's construction) orderings of each planning variable's parents in the causal graph.

5.3 Correctness and Complexity

Lemma 4 *Let Π be a $\mathbf{P}(1)$ problem, $\mathsf{COP}_\Pi = (\mathcal{X}, \mathcal{F})$ be the corresponding constraint optimization problem, and $\overline{x}$ be an optimal assignment to $\mathcal{X}$ with $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \alpha$.*

(I) If $\alpha < \infty$, then a plan of cost α for Π can be reconstructed from $\overline{x}$ in time polynomial in the description size of Π.

(II) If Π has a plan, then $\alpha < \infty$.

Proof:
(I) Given a COP solution $\overline{x}$ with $\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}) = \alpha < \infty$, we construct a plan ρ for Π with $\mathcal{C}(\rho) = \alpha$. We construct this plan by

1. Traversing the planning variables in a topological ordering of the causal graph $CG(\Pi)$, and associating each variable v with a sequence $\rho_v \in A_v^*$.

2. Merging the constructed sequences $\rho_{v_1}, \ldots, \rho_{v_n}$ into the desired plan ρ.

For each $v \in V$ with $\mathsf{pred}(v) = \emptyset$ we set $\rho_v = \langle a_1 \cdot \ldots \cdot a_l \rangle$, where $l = |\overline{x}_v| - 1$, and a_i is defined as in Eq 176 below.

$$a_i = \begin{cases} a_{\mathsf{w}_v}, & i \text{ is odd}, \\ a_{\mathsf{b}_v}, & i \text{ is even}, \end{cases} \tag{176}$$

In turn, for each $v \in V$ with $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, given $\overline{x}_v^{w_1}, \ldots, \overline{x}_v^{w_k}$, we distinguish between the following cases.

[**R1 is played**] R1 is played by one of the parents, while all other parents play role R11.

[**R1 is played by** w_1] Eq. 174 then implies

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R1}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and $\overline{x}_v^{w_1} \ni \mathbf{S} = \mathrm{S1}$. From Eq. 175 we then have $\overline{x}_v^{w_j} \ni \mathbf{S}' = \mathrm{S1}$ for each $1 < j \leq k$, giving us

$$\varphi_{x_v^{w_j}}(\overline{x}_v^{w_j}, \overline{x}_v^{w_{j-1}}, \overline{x}_{w_j}) = 0$$

[**R1 is played by** w_j**,** $j > 1$] Eq. 175 then implies

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \\ \varphi(\mathrm{R1}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0.$$

From Eq. 174 we also have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

In both these sub-cases, ρ_v, $>_v$ and $>_{v,w}$ are specified as in the proof of Theorem 6, case I.

[**R2 is played**] R2 is played by one of the parents, while all other parents play role R11.

[**R2 is played by** w_1] Eq. 174 then implies

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R2}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and, for each $1 < j \leq k$, Eq. 175 implies

$$\varphi_{x_v^{w_j}}(\overline{x}_v^{w_j}, \overline{x}_v^{w_{j-1}}, \overline{x}_{w_j}) = 0$$

If $\varphi(\mathrm{R2}, \llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_1}) = \xi(\#_{\mathsf{w}}, 0, \#_{\mathsf{b}}, 0)$, then ρ_v, $>_v$ and $>_{v,w}$ are specified as in the proof of Theorem 6, case III.2, otherwise, as in the case IV.2.

[**R2 is played by** w_j**,** $j > 1$] Eq. 175 then implies

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R2}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0.$$

From Eq. 174 we also have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

If $\varphi(\mathrm{R2}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j}) = \xi(\#_{\mathsf{w}} - \#'_{\mathsf{w}}, 0, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, 0)$, then ρ_v, $>_v$ and $>_{v,w}$ are specified as in the proof of Theorem 6, case III.2, otherwise, as in the case IV.2.

[**R3 and R4 are played**] Those roles are played by two of the parents, while all other parents play role R11.

[**R3 is played by** w_1**, R4 is played by** w_j**,** $j > 1$] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R3}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R4}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \leq k$, such that $i \neq j$:

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

[**R4 is played by** w_1**, R3 is played by** w_j**,** $j > 1$] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R4}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R3}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

[**R**3 **is played by** w_j**, R**4 **is played by** w_t**,** $j \neq t$**,** $j, t > 1$] From Eqs. 174 and 175 we have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_\mathsf{w}, \#'_\mathsf{b}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R3}, \llbracket \mathbf{S}, \#_\mathsf{w} - \#'_\mathsf{w}, \#_\mathsf{b} - \#'_\mathsf{b}, \eta \rrbracket, \sigma_{w_j})$$

and

$$\varphi_{x_v^{w_t}}(\llbracket \mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_\mathsf{w}, \#'_\mathsf{b}, \eta' \rrbracket, \sigma_{w_t}) = \varphi(\mathrm{R4}, \llbracket \mathbf{S}, \#_\mathsf{w} - \#'_\mathsf{w}, \#_\mathsf{b} - \#'_\mathsf{b}, \eta \rrbracket, \sigma_{w_t})$$

and, for all $1 < i \leq k$ such that $i \notin \{j, t\}$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

In all these three sub-cases, ρ_v, $>_v$ and $>_{v,w}$ are specified as in the proof of Theorem 6, case II.

[**R**5 **is played**] R5 is played by one of the parents, while all other parents play role R11.

[**R**5 **is played by** w_1] Eqs. 174 and 175 imply

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R5}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and, for each $1 < j \leq k$,

$$\varphi_{x_v^{w_j}}(\overline{x}_v^{w_j}, \overline{x}_v^{w_{j-1}}, \overline{x}_{w_j}) = 0$$

Considering now the specification of the function φ in Eq. 168,

- If the first case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.1.a.
- If the first case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.1.a.
- If the second case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.a.
- If the second case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.a.
- If the third case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.a.
- If the forth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.a.
- If the fifth case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.a.

- If the fifth case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.5.a.
- If the sixth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.a.
- If the seventh case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.5.a.
- If the eighth case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.7.a.
- If the eighth case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.7.a.

[**R5 is played by** w_j**,** $j > 1$] Eq. 175 then implies

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_\mathsf{w}, \#_\mathsf{b}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_\mathsf{w}, \#'_\mathsf{b}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R5}, \llbracket \mathbf{S}, \#_\mathsf{w} - \#'_\mathsf{w}, \#_\mathsf{b} - \#'_\mathsf{b}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0.$$

From Eq. 174 we also have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

Here ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the previous case.

[**R6 is played**] R6 is played by one of the parents, while all other parents play role R11.

[**R6 is played by** w_1] Eqs. 174 and 175 imply

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R6}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and, for each $1 < j \leq k$,

$$\varphi_{x_v^{w_j}}(\overline{x}_v^{w_j}, \overline{x}_v^{w_{j-1}}, \overline{x}_{w_j}) = 0$$

Considering now the specification of the function φ in Eq. 169,

- If the first case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.1.b.
- If the first case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.1.b.
- If the second case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.b.
- If the second case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.b.

- If the third case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.b.
- If the forth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.b.
- If the fifth case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.b.
- If the fifth case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.5.b.
- If the sixth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.b.
- If the seventh case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.7.b.
- If the seventh case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.7.b.

[**R6 is played by** w_j**,** $j > 1$] Eq. 175 then implies

$$\varphi_{x_v^{w_j}}([\![\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta]\!], [\![\mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta']\!], \sigma_{w_j}) = \\ \varphi(\mathrm{R6}, [\![\mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta]\!], \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0.$$

From Eq. 174 we also have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

Here ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the previous case.

[**R7 and R9 are played**] Those roles are played by two of the parents, while all other parents play role R11.

[**R7 is played by** w_1**, R9 is played by** w_j**,** $j > 1$] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R7}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}([\![\mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta]\!], [\![\mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta']\!], \sigma_{w_j}) = \\ \varphi(\mathrm{R9}, [\![\mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta]\!], \sigma_{w_j})$$

and ,for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Considering now the specification of the function φ in Eq. 170,

- If the first case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.1.a.
- If the first case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.1.a.
- If the second case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.a.
- If the third case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.a.
- If the forth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.a.
- If the fifth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.5.a.
- If the sixth case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.7.a.
- If the sixth case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.7.a.

[**R9 is played by w_1, R7 is played by w_j, $j > 1$**] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\mathrm{R9}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R7}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Here ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the previous case.

[**R7 is played by w_j, R9 is played by w_t, $j \neq t$, $j, t > 1$**] From Eqs. 174 and 175 we have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R7}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and

$$\varphi_{x_v^{w_t}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_t}) = \varphi(\mathrm{R9}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_t})$$

and, for all $1 < i \leq k$, such that $i \notin \{j, t\}$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Then, ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the two previous cases.

[**R8 and R10 are played**] Those roles are played by two of the parents, while all other parents play role R11.

[**R8 is played by w_1, R10 is played by w_j, $j > 1$**] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\text{R8}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\text{R10}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Considering now the specification of the function φ in Eq. 171,

- If the first case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.1.b.
- If the first case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.1.b.
- If the second case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.3.b.
- If the third case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.3.b.
- If the forth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.5.b.
- If the fifth case holds, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.5.b.
- If the sixth case holds, and the minimum is obtained at the first expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case III.7.b.
- If the sixth case holds, and the minimum is obtained at the second expression, then ρ_v, $>_v$ and $>_{v,w}$ are defined as in the proof of Theorem 6, case IV.7.b.

[**R10 is played by w_1, R8 is played by w_j, $j > 1$**] From Eqs. 174 and 175 we then have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = \varphi(\text{R10}, \overline{x}_v^{w_1}, \overline{x}_{w_1})$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\text{R8}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and, for all $1 < i \neq j \leq k$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Here ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the previous case.

[**R8 is played by w_j, R10 is played by w_t, $j \neq t$, $j,t > 1$**] From Eqs. 174 and 175 we have

$$\varphi_{x_v^{w_1}}(\overline{x}_v^{w_1}, \overline{x}_{w_1}) = 0$$

and

$$\varphi_{x_v^{w_j}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_j}) = \varphi(\mathrm{R8}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_j})$$

and

$$\varphi_{x_v^{w_t}}(\llbracket \mathbf{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, \eta \rrbracket, \llbracket \mathbf{S}', \#'_{\mathsf{w}}, \#'_{\mathsf{b}}, \eta' \rrbracket, \sigma_{w_t}) = \varphi(\mathrm{R10}, \llbracket \mathbf{S}, \#_{\mathsf{w}} - \#'_{\mathsf{w}}, \#_{\mathsf{b}} - \#'_{\mathsf{b}}, \eta \rrbracket, \sigma_{w_t})$$

and, for all $1 < i \leq k$, such that $i \notin \{j, t\}$,

$$\varphi_{x_v^{w_i}}(\overline{x}_v^{w_i}, \overline{x}_v^{w_{i-1}}, \overline{x}_{w_i}) = 0$$

Then, ρ_v, $>_v$ and $>_{v,w}$ are specified exactly as in the two previous cases.

Until now, for each variable $v \in V$, we have specified the action sequence ρ_v and the order $>_v$ over the elements of ρ_v. For each $w \in \mathsf{pred}(v)$, we have specified the order $>_{v,w}$, and proved that all $>_v \cup >_{v,w}$ and $>_w \cup >_{v,w}$ form strict partial orders over their domains. Similarly to the uniform cost case, this construction allows us to apply now Theorem 1 on the (considered as sets) sequences ρ_v and orders $>_v$ and $>_{v,w}$, proving that

$$>= \bigcup_{v \in V} (>_v \cup \bigcup_{w \in \mathsf{pred}(v)} >_{v,w})$$

forms a strict partial order over the union of $\rho_{v_1}, \ldots, \rho_{v_n}$.

Finally, we note that the plan extraction step of the algorithm polytree-1-dep corresponds exactly to the above construction along Eqs. 53-58, 60-62, 64-65, 72-73, 75-76, 84-85, 87, 89, 96-97, 99, 101, 108-109, 111-112, 114, 121-122, 124, 132-133, 135-136, 138, 145-146, 148-149, 157, providing us in poly-time with concrete cost-optimal plan corresponding to the optimal solution for COP_Π.

(II) We now prove that if Π is solvable, then we must have $\alpha < \infty$. Assume to the contrary that this is not the case. Let Π be a solvable $\mathbf{P}(1)$ problem, and let (using Theorem 6) ρ be an irreducible, post-3/2 plan for Π. Given such ρ, let a COP assignment $\overline{x}_\rho$ be defined as follows.

1. For each COP variable x_v, the assignment $\overline{x}_\rho$ provides the value $\sigma_v \in \trianglerighteq^*[\sigma(v)]$ such that $|\sigma_v| = |\rho{\downarrow}_v| + 1$.
2. For each variable $v \in V$, such that $\mathsf{pred}(v) \neq \emptyset$, find the (at most two) parents that prevail the actions in $\rho{\downarrow}_v$. Let w be such a parent that performs a role $\mathrm{R} \in \{\mathrm{R1}, \mathrm{R2}, \mathrm{R3}, \mathrm{R5}, \mathrm{R6}, \mathrm{R7}, \mathrm{R8}\}$, and w' be the other such parent that performs one of the roles $\mathrm{R}' \in \{\mathrm{R4}, \mathrm{R9}, \mathrm{R10}, \mathrm{R11}\}$. (By definition of post-3/2 action sequences, the

rest of the parents all perform role R11.) Given that, if $|\mathsf{pred}(v)| = k > 0$, we adopt an ordering of $\mathsf{pred}(v)$ such that $w_1 = w$ and $w_k = w'$. First, the assignment $\overline{x}_v^{w_k}$ to COP variable $x_v^{w_k}$ provides the value $\left[\!\left[\text{S1}, \lceil \frac{|\sigma_v|-1}{2} \rceil, \lfloor \frac{|\sigma_v|-1}{2} \rfloor, |\sigma_v| - 1 \right]\!\right]$. Then, for $1 \leq i < k$, the assignment $\overline{x}_v^{w_i}$ to COP variable $x_v^{w_i}$ provides the value $[\![\text{S}, \#_{\mathsf{w}}, \#_{\mathsf{b}}, |\sigma_v| - 1]\!]$, where

$$\text{S} = \begin{cases} \text{S2}, & \text{R}' = \text{R4} \\ \text{S4}, & \text{R}' = \text{R9} \\ \text{S5}, & \text{R}' = \text{R10} \\ \text{S1}, & \text{R}' = \text{R11} \end{cases}$$

and $\#_{\mathsf{w}}$ and $\#_{\mathsf{b}}$ are the numbers of actions in $\rho\!\downarrow_v$ that change the value of v to w_v and b_v, respectively, while being prevailed by the value of w_1.

From Eq. 161-175 we then have that, for each $v \in V$, if $\mathsf{pred}(v) = \emptyset$, then $\varphi_{x_v}(\overline{x}_v) = \mathcal{C}(\rho\!\downarrow_v)$. Otherwise, if $\mathsf{pred}(v) = \{w_1, \ldots, w_k\}$, then $\varphi_{x_v}(\overline{x}_v, \overline{x}_v^{w_k}) = 0$, and $\sum_{w \in \mathsf{pred}(v)} \varphi_{x_v^w}(\overline{x}_\rho) = \mathcal{C}(\rho\!\downarrow_v)$. Therefore, we have

$$\sum_{\varphi \in \mathcal{F}} \varphi(\overline{x}_\rho) \;=\; \sum_{v \in V} \mathcal{C}(\rho\!\downarrow_v) \;=\; \mathcal{C}(\rho),$$

which is what we had to prove. ■

Theorem 7 *Cost-optimal planning for* $\mathbf{P}(1)$ *is tractable.*

Proof: The correctness of the $\mathsf{polytree\text{-}1\text{-}dep}$ algorithm is given by Lemma 4. We now show that, given a planning problem $\Pi \in \mathbf{P}(1)$, the corresponding constraint optimization problem COP_Π can be constructed and solved in time polynomial in the description size of Π.

Let n be the number of state variables in Π. In $\mathsf{polytree\text{-}1\text{-}dep\text{-}uniform}$, we first construct the constraint optimization problem COP_Π over $\Theta(n^2)$ variables $\mathcal{X}$ with domain sizes being bounded either by $O(n)$ or by $O(n^3)$ (for COP variables representing state variables and causal graph edges, respectively). The number of functional components in COP_Π is $\Theta(n^2)$, each defined over one variable with domain size of $O(n)$ and either one or two variables with domain sizes of $O(n^3)$. The construction is linear in the size of the resulting COP, and thus is accomplished in time $O(n^9)$.

Applying then to COP_Π a tree-decomposition that clusters the scopes of the functional components $\mathcal{F}$, we arrive into an equivalent, tree-structured constraint optimization problem over $\Theta(n^2)$ variables with domains of size $O(n^7)$. Such a tree-structured COP can be solved in time $O(xy^2)$ where x is the number of variables and y is an upper bound on the size of a variable's domain (Dechter, 2003). Therefore, solving COP_Π can be done in time $O(n^{16})$. As this dominates both the time complexity of constructing COP_Π, and the time complexity of extracting a plan from the optimal solution to COP_Π (see the proof of (I) in Lemma 4), the overall complexity of the algorithm $\mathsf{polytree\text{-}1\text{-}dep\text{-}uniform}$ is $O(n^{16})$, and therefore polynomial in the description size of Π. ■

6. Drawing the Limits of k-Dependence

Having read this far, the reader may wonder whether 1-dependence is not a strong enough property to make the cost-planning tractable even for some more complex than polytree forms of the causal graph. In this last technical section we discuss the limits of the power of k-dependence (and, in particular, of 1-dependence), and present some negative results that draw a boundary between the tractable and intractable k-dependent UB problems.

- In Theorem 8 we show that cost-optimal planning is already hard for the $\mathbf{S}_b^b(1)$ problem class, that is, the class of 1-dependent UB problems inducing directed-path singly connected causal graphs with both in- and out-degrees being bounded by a constant. This result further stresses the connection between the undirected cycles in the causal graph and the complexity of various planning tasks, which has been first discussed by Brafman and Domshlak (2003).
- In Theorem 9 we show that even non-optimal planning is hard for the $\mathbf{S}_b^b(2)$ problem class. This results suggests that 1-dependence is a very special special case of k-dependence in terms of the connection to computational tractability. However, given the (still) empty entries in Figures 4a and 4b, further analysis of the "criticality" of 1-dependence is needed.

Theorem 8 *Cost-optimal planning for $\mathbf{S}_b^b(1)$ is NP-complete.*

Proof: The membership in NP is by Theorem 2 of Brafman and Domshlak (2003). The proof of NP-hardness is by a polynomial reduction from the seminal Vertex Cover problem (Garey & Johnson, 1978). The problem of Vertex Cover is: given an undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, find a minimal-size subset $\mathbf{V}'$ of $\mathbf{V}$ such that each edge in $\mathbf{E}$ has at least one of its two end-nodes in $\mathbf{V}'$. Given an undirected graph $\mathbf{G} = (\mathbf{V}, \mathbf{E})$, let the planning problem $\Pi_{\mathbf{G}} = \langle V_{\mathbf{G}}, A_{\mathbf{G}}, I_{\mathbf{G}}, G_{\mathbf{G}} \rangle$ be defined as follows.

- $V_{\mathbf{G}} = \{v_1, \ldots, v_{|\mathbf{V}|}, u_1, \ldots, u_{|\mathbf{E}|}\}$, and, for all v_i, u_j, $Dom(v_i) = Dom(u_j) = \{T, F\}$,
- $I_{\mathbf{G}} = \{v_i = F \mid v_i \in V_{\mathbf{G}}\} \bigcup \{u_i = F \mid u_i \in V_{\mathbf{G}}\}$,
- $G_{\mathbf{G}} = \{u_i = T \mid u_i \in V_{\mathbf{G}}\}$,
- Actions $A_{\mathbf{G}} = A_{\mathbf{V}} \cup A_{\mathbf{E}}$, where $A_{\mathbf{V}} = \{a_{v_1}, \ldots, a_{v_{|\mathbf{V}|}}\}$ with

$$\mathsf{pre}(a_{v_i}) = \{v_i = F\}, \mathsf{eff}(a_{v_i}) = \{v_i = T\}, \mathcal{C}(a_{v_i}) = 1 \tag{177}$$

  and $A_{\mathbf{E}} = \{a_{u_1}, a'_{u_1}, \ldots, a_{u_{|\mathbf{E}|}}, a'_{u_{|\mathbf{E}|}}\}$ with

$$\begin{aligned} \mathsf{pre}(a_{u_i}) &= \{u_i = F, v_{i_1} = T\}, \\ \mathsf{pre}(a'_{u_i}) &= \{u_i = F, v_{i_2} = T\}, \\ \mathsf{eff}(a_{u_i}) = \mathsf{eff}(a'_{u_i}) &= \{u_i = T\}, \\ \mathcal{C}(a_{u_i}) = \mathcal{C}(a'_{u_i}) &= 0 \end{aligned} \tag{178}$$

  where the variables v_{i_1}, v_{i_2} correspond to the endpoints of the edge corresponding to the variable u_i.

Given this construction of $\Pi_{\mathbf{G}}$, it is easy to see that (i) any plan ρ for $\Pi_{\mathbf{G}}$ provides us with a vertex cover $\mathbf{V}_\rho$ for $\mathbf{G}$ such that $|\mathbf{V}_\rho| = \mathcal{C}(\rho)$ and vice versa, and thus (ii) cost-optimal plans for $\Pi_{\mathbf{G}}$ (and only such plans for $\Pi_{\mathbf{G}}$) provide us with minimal vertex covers for $\mathbf{G}$. The topology of the causal graph of $\Pi_{\mathbf{G}}$ is as required, and 1-dependence of $\Pi_{\mathbf{G}}$ is immediate from Eqs. 177-178. This finalizes the proof of NP-completeness of cost-optimal planning for $\mathbf{S}_b^b(1)$. ∎

Theorem 9 *Planning for* $\mathbf{S}_b^b(2)$ *is NP-complete.*

Proof: The proof is basically given by the proof of Brafman and Domshlak for Theorem 2 in (Brafman & Domshlak, 2003). The polynomial reduction there is from 3-SAT to planning for $\mathbf{S}$. Observing that 3-SAT remains hard even if no variable participates in more than five clauses of the formula (Garey & Johnson, 1978), and that the reduction of Brafman and Domshlak from *such* 3-SAT formulas is effectively to planning for $\mathbf{S}_b^b(2)$, accomplishes the proof of our claim. ∎

References

Bacchus, F., & Yang, Q. (1994). Downward refinement and the efficiency of hierarchical problem solving. *Artificial Intelligence*, *71*(1), 43–100.

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.

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

Boutilier, C., Brafman, R., Domshlak, C., Hoos, H., & Poole, D. (2004). CP-nets: A tool for representing and reasoning about conditional *ceteris paribus* preference statements. *Journal of Artificial Intelligence Research*, *21*, 135–191.

Brafman, R. I., & Domshlak, C. (2003). Structure and complexity of planning with unary operators. *Journal of Artificial Intelligence Research*, *18*, 315–349.

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

Chapman, D. (1987). Planning for conjunctive goals. *Artificial Intelligence*, *32*(3), 333–377.

Cormen, T. H., Leiserson, C. E., & Rivest, R. L. (1990). *Introduction to Algorithms*. MIT Press.

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

Edelkamp, S. (2001). Planning with pattern databases. In *Proceedings of the European Conference on Planning (ECP)*, pp. 13–34.

Erol, K., Nau, D. S., & Subrahmanian, V. S. (1995). Complexity, decidability and undecidability results for domain-independent planning. *Artificial Intelligence, Special Issue on Planning*, *76*(1–2), 75–88.

Garey, M. R., & Johnson, D. S. (1978). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Company, New-York.

Haslum, P. (2006). *Admissible Heuristics for Automated Planning*. Ph.D. thesis, Linköping University, Department of Computer and Information Science.

Haslum, P., Bonet, B., & Geffner, H. (2005). New admissible heuristics for domain-independent planning. In *Proceedings of the Twentieth National Conference on Artificial Intelligence (AAAI)*, pp. 1163–1168.

Haslum, P., & Geffner, H. (2000). Admissible heuristics for optimal planning. In *Proceedings of the Fifth International Conference on Artificial Intelligence Planning Systems (ICAPS)*, pp. 140–149.

Helmert, M. (2003). Complexity results for standard benchmark domains in planning. *Artificial Intelligence*, *146*(2), 219–262.

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

Hoffmann, J. (2003). *Utilizing Problem Structure in Planning: A Local Search Approach.* No. 2854 in LNAI. Springer-Verlag.

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

Hoffmann, J., & Edelkamp, S. (2005). The deterministic part of ipc-4: An overview. *Journal of Artificial Intelligence Research*, *24*, 519–579.

Jonsson, A., & Gimenez, O. (2007). On the hardness of planning problems with simple causal graphs. In *Proceedings of the 17th International Conference on Automated Planning and Scheduling (ICAPS)*.

Jonsson, P., & Bäckström, C. (1995). Incremental planning. In *New Directions in AI Planning: EWSP'95-3rd European Workshop on Planning*, pp. 79–90, Assisi, Italy.

Jonsson, P., & Bäckström, C. (1998). Tractable plan existence does not imply tractable plan generation. *Annals of Mathematics and Artificial Intelligence*, *22*(3-4), 281–296.

Klein, I., Jonsson, P., & Bäckström, C. (1998). Efficient planning for a miniature assembly line. *Artificial Intelligence in Engineering*, *13*(1), 69–81.

Knoblock, C. (1994). Automatically generating abstractions for planning. *Artificial Intelligence*, *68*(2), 243–302.

Newell, A., & Simon, H. A. (1961). GPS, a program that simulates human thought. In Billing, H. (Ed.), *Lernende Automaten*, pp. 109–124. R. Oldenbourg, Munich, Germany.

Sacerdoti, E. (1974). Planning in a hierarchy of abstraction spaces. *Artificial Intelligence*, *5*, 115–135.

Tenenberg, J. (1988). Abstraction in planning. Tech. rep. TR250, University of Rochester, Rochester, New York.

Williams, B., & Nayak, P. (1996). A model-based approach to reactive self-configuring systems. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence*, pp. 971–977, Portland, OR. AAAI Press.

Williams, B., & Nayak, P. (1997). A reactive planner for a model-based executive. In *Proceedings of the 15th International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 1178–1185, Nagoya, Japan.