Proceedings of the Twenty-Sixth International Conference on
Automated Planning and Scheduling (ICAPS 2016)

A Multi-Parameter Complexity Analysis of Cost-Optimal and Net-Benefit Planning

Meysam Aghighi and **Christer Bäckström**
Department of Computer and Information Science
Linköping University, 581 83 Linköping, Sweden
meysam.aghighi@liu.se christer.backstrom@liu.se

Abstract

Aghighi and Bäckström have previously studied cost-optimal planning (COP) and net-benefit planning (NBP) for three action cost domains: the positive integers ($\mathbb{Z}_+$), the non-negative integers ($\mathbb{Z}_0$) and the positive rationals ($\mathbb{Q}_+$). These were indistinguishable under standard complexity analysis for both problems, but separated for COP using parameterised complexity analysis. With the plan cost, k, as parameter, COP was **W**[2]-complete for $\mathbb{Z}_+$, but para-**NP**-hard for both $\mathbb{Z}_0$ and $\mathbb{Q}_+$, i.e. presumably much harder. NBP was para-**NP**-hard for all three domains, thus remaining unseparable. We continue by considering combinations with several additional parameters and also the non-negative rationals ($\mathbb{Q}_0$). Examples of new parameters are the plan length, ℓ, and the largest denominator of the action costs, d. Our findings include: (1) COP remains **W**[2]-hard for all domains, even if combining all parameters; (2) COP for $\mathbb{Z}_0$ is in **W**[2] for the combined parameter $\{k, \ell\}$; (3) COP for $\mathbb{Q}_+$ is in **W**[2] for $\{k, d\}$ and (4) COP for $\mathbb{Q}_0$ is in **W**[2] for $\{k, d, \ell\}$. For NBP we consider further additional parameters, where the most crucial one for reducing complexity is the sum of variable utilities. Our results help to understand the previous results, eg. the separation between $\mathbb{Z}_+$ and $\mathbb{Q}_+$ for COP, and to refine the previous connections with empirical findings.

1 Introduction

Length-optimal planning (LOP) is often successfully solved in practice by modern planners. It is also well studied theoretically, it is **PSPACE**-complete in the general case both for STRIPS (Bylander 1994) and for SAS$^+$ (Bäckström and Nebel 1995). Using parameterised complexity analysis, LOP is also **W**[2]-complete when parameterised with plan length (Bäckström et al. 2015). Cost-optimal planning (COP) has proven more difficult to solve in practice than LOP. Actions with zero cost may result in very long plans with very low cost which are very expensive to find in practice (Richter and Westphal 2010; Benton et al. 2010). Also rational action costs and big differences in action costs seem to cause similar problems, even without zero-cost actions (Cushing, Benton, and Kambhampati 2010; Wilt and Ruml 2011). These are observations from the viewpoint of actual algorithms. Aghighi and Bäckström (2015) provided a complementary problem analysis of COP for three different cost domains: the positive integers ($\mathbb{Z}_+$), the non-negative integers ($\mathbb{Z}_0$) and the positive rationals ($\mathbb{Q}_+$), the latter two being the situations observed to cause problems in practice. They first found that standard complexity analysis is not sufficient for explaining these observations; COP is **PSPACE**-complete for all three cost domains. However, using parameterised complexity analysis with the plan cost (k) as parameter, they demonstrated a separation in complexity: COP for $\mathbb{Z}_+$ is **W**[2]-complete, i.e. no harder than LOP, while COP for $\mathbb{Z}_0$ and $\mathbb{Q}_+$ is para-**NP**-hard, i.e. presumably much harder. Their results thus suggest that the observed problems are inherent in COP and not artifacts of the actual algorithms used. This begs for a better understanding of these problems, in particular since the problems can arise in practical applications; for instance, big differences in action cost can arise in robotics (Likhachev and Ferguson 2009) and zero-cost actions are even artificially introduced in some cases (Cooper, de Roquemaurel, and Régnier 2011).

While the results of Aghighi and Bäckström (2015) were consistent with the observations about search algorithms in the literature, they did not provide much understanding or any additional knowledge. By using combinations of many parameters, instead of one, and using more complex techniques, we give a more fine-grained picture that does help to explain, and even refine, the previous observations. Examples of parameters are plan length (ℓ), maximum number of zero-cost actions in a plan (z), inverse minimum action cost $\frac{1}{c_{\min}}$ and the maximum denominator for rational costs (d). We also use parameters like the number of different actions costs ($\#c$), since the 'number of numbers' has proven a useful parameter in some cases (Fellows, Gaspers, and Rosamond 2012). We further add two cost domains, the non-negative rationals ($\mathbb{Q}_0$) and the rationals greater than or equal to one ($\mathbb{Q}_1$). We provide an almost complete complexity map for all combinations of parameters and cost domains. Some of our major results are the following: (1) COP remains **W**[2]-hard for all cost domains even when combining all parameters; (2) COP for $\mathbb{Z}_0$ is in **W**[2] for the parameter combinations $\{k, \ell\}$ and $\{k, z\}$; (3) COP for $\mathbb{Q}_+$ is in **W**[**P**] for the parameter combination $\{k, \frac{1}{c_{\min}}\}$ and in **W**[2] for $\{k, d\}$, i.e. parameter d seems more relevant than $\frac{1}{c_{\min}}$ and (4) COP for $\mathbb{Q}_0$ is in **W**[2] for the parameter combinations $\{k, \ell, d\}$ and

$\{k, z, d\}$. These results provide a deeper understanding of the previously mentioned problems. In particular, (3) provides a better characterization of the problem observed by Cushing, Benton, and Kambhampati (2010).

We also consider the *net-benefit planning (NBP)* problem which assigns utility values to the goals and asks for a plan that maximises the net benefit, i.e. the utility of the plan minus its cost. This problem is also **PSPACE**-complete (van den Briel et al. 2004) in the general case, although simpler subclasses exist (Aghighi and Jonsson 2014). It is, however, known to be harder than COP in the sense that it is para-**NP**-hard for all three cost domains considered by Aghighi and Bäckström (2015). Hence, we analyse this problem for combinations of the parameters we use for COP plus some additional parameters, where the most relevant additional parameter is the sum of all goal utilities, t.

The remainder of the paper is organized as follows. Sections 2 and 3 contain overviews of parameterised complexity theory and SAS$^+$ planning. Section 4 discusses parameterisation of COP, introduces our parameters and summarizes our complexity results for COP. The actual complexity results appear in Section 5, which also includes some explicit upper and lower bounds as an example of how to interpret the complexity results. We continue by analysing NBP in Section 6. The paper ends by a discussion section.

2 Parameterised Complexity

Parameterised complexity theory allows for more fine-grained complexity analyses than traditional complexity theory, and it was invented with the purpose of delivering complexity results that conform better with practical experience. We briefly recall the most important details and refer the reader to the literature, cf. Downey and Fellows (1999) or Flum and Grohe (2006), for an in-depth treatment.

A *parameterised problem* is a language $L \subseteq \Sigma^* \times \Sigma^*$ over some finite alphabet Σ. The *instances* of L are tuples $\langle \mathbb{I}, k \rangle$, where k is called the *parameter*. The parameter is often a non-negative integer, but it can be anything, e.g. a rational number, three integers or a graph. For simplicity, we first assume the parameter is a non-negative integer, i.e. $L \subseteq \Sigma^* \times \mathbb{Z}_0$. A parameterised problem is *fixed-parameter tractable (fpt)* if there exists an algorithm that solves every instance $\langle \mathbb{I}, k \rangle$ of size $n = ||\mathbb{I}||$ in time $f(k) \cdot n^c$ where f is an arbitrary computable function and c is a constant independent of both n and k. **FPT** is the class of all fixed-parameter tractable decision problems. In contrast to classical tractability, some exponentiality is allowed, but confined to the parameter only, thus better reflecting reality.

There is a hierarchy of parameterised complexity classes

$$\mathbf{FPT} \subseteq \mathbf{W}[1] \subseteq \mathbf{W}[2] \subseteq \mathbf{W}[3] \subseteq \cdots \subseteq \mathbf{W}[\mathbf{P}],$$

known as the *W hierarchy*, which can be used for hardness results. The **W**[i] classes are defined by the WEIGHTED SATISFIABILITY PROBLEM for certain restricted circuits, where **W**[**P**] is the case of arbitrary circuits. Hardness for parameterised classes is proven in the usual way, but using fpt reductions instead of ordinary polynomial-time reductions. An *fpt reduction* from a parameterised language $L \subseteq \Sigma^* \times \mathbb{Z}_0$ to another parameterised language $L' \subseteq \Sigma^* \times \mathbb{Z}_0$ is a mapping $R : \Sigma^* \times \mathbb{Z}_0 \to \Sigma^* \times \mathbb{Z}_0$ such that: (1) $\langle \mathbb{I}, k \rangle \in L$ if and only if $\langle \mathbb{I}', k' \rangle = R(\mathbb{I}, k) \in L'$; (2) there is a computable function f and a constant c such that R can be computed in time $f(k) \cdot n^c$, where $n = ||\mathbb{I}||$; and (3) there is a computable function g such that $k' \leq g(k)$. It is known that $\mathbf{P} \subseteq \mathbf{FPT}$, but otherwise the parameterised complexity classes are mainly orthogonal to the classical ones. For instance, there are **NP**-complete problems that are **W**[**P**]-complete and there are **PSPACE**-complete problems that are in **FPT**. We will also consider the class para-**NP**, which consists of all parameterised problems that can be solved in *non*-deterministic time $f(k) \cdot n^c$, where f is an arbitrary computable function and c is a constant independent of both n and k. It is known that $\mathbf{W}[\mathbf{P}] \subseteq$ para-**NP**,

Using more than one parameter is usually straightforward, since the general definition allows the parameter to be any string. Consider a problem with two parameters, k_1 and k_2. This problem is fixed-parameter tractable if it can be solved in time $f(k_1, k_2) \cdot n^c$ for some computable function f and some constant c. It is equivalent to say that it is fixed-parameter tractable if it can be solved in time $f(k_1 + k_2) \cdot n^c$ for some computable function f and some constant c. The same principle works for fpt reductions.

If all parameters of a parameterised language L are set to constants, the result is a *slice* of L, which is an ordinary non-parameterised language. It follows from Thm. 2.14 in Flum and Grohe (2006) that if a slice of L is **NP**-hard, then L is para-**NP**-hard, i.e. one can prove that L is para-**NP**-hard by polynomial reduction from some **NP**-hard language to a slice of L, which we will tacitly use in some proofs.

3 SAS$^+$ Planning

We assume the reader is familiar with the SAS$^+$ planning framework (Bäckström and Nebel 1995), and only briefly recapitulate it. A SAS$^+$ *planning instance* is a tuple $\mathbb{P} = \langle V, A, I, G \rangle$, where V is a set of *variables*, A is set of *actions*, I is the *initial state* and G is the partial *goal state*. Each variable $v \in V$ has a finite *domain* $D(v)$, and each action $a \in A$ has a *precondition* $\mathrm{pre}(a)$ and an *effect* $\mathrm{eff}(a)$. A *plan* (i.e. a solution) for $\mathbb{P}$ is a sequence of actions from A that transforms I into a state that satisfies G. We write $a : P \Rightarrow E$ to define an action a with precondition P and effect E. The set of variables with a defined value in a state s is denoted $\mathrm{vars}(s)$ and $s[v]$ is the value of v in s. The following unparameterised problems are commonly studied. The PLAN SATISFIABILITY problem (PSAT) takes a SAS$^+$ instance $\mathbb{P}$ as input and ask if $\mathbb{P}$ has a plan or not. The LENGTH-OPTIMAL PLANNING problem (LOP) takes a SAS$^+$ instance $\mathbb{P}$ and a non-negative integer ℓ as input, and asks if $\mathbb{P}$ has a plan ω of length $|\omega| \leq \ell$?

In cost-optimal planning we additionaly specify a cost $c(a)$ for each action a and ask for the minimum cost for a plan. The cost of a plan $\omega = a_1, \ldots, a_n$ is $c(\omega) = \sum_{i=1}^{n} c(a_i)$. We also specify a domain $\mathbb{D}$ for the costs.

COST-OPTIMAL PLANNING (COP($\mathbb{D}$))
Instance: A tuple $\mathbb{P} = \langle V, A, I, G, c \rangle$, where $\langle V, A, I, G \rangle$ is a SAS$^+$ instance and $c : A \to \mathbb{D}$ is a

cost function. A value $k \in \mathbb{D}$.
Question: Does $\mathbb{P}$ have a plan ω of cost $c(\omega) \leq k$?

The numeric domains we consider for $\mathbb{D}$ are: The positive integers ($\mathbb{Z}_+$), the non-negative integers ($\mathbb{Z}_0$), the positive rationals ($\mathbb{Q}_+$), the non-negative rationals ($\mathbb{Q}_0$) and the set $\mathbb{Q}_1 = \{x \in \mathbb{Q} \mid x \geq 1\}$. The reason for including $\mathbb{Q}_1$ is to see if a complexity result for $\mathbb{Q}_+$ depends on values smaller than 1 or only on the values being rational.

4 Parameterised Cost-optimal Planning

For parameterised planning problems, we add a list of parameters, i.e. we write the parameterised versions of problems LOP and COP($\mathbb{D}$) as LOP(π) and COP($\mathbb{D}, \pi$), where π is a set of parameters. We will consider the following parameters (where only ℓ is relevant for LOP):

k: Max. plan cost.
ℓ: Max. plan length.
z: Max. number of zero-cost actions in the plan.
c_{min}: Min. positive action cost in instance.
c_{max}: Max. action cost in instance.
d: Max. denominator of positive action costs in instance.
$\#c$: Max. number of different action costs in instance.
$\#d$: Max. number of different denominators in instance.

Parameter z is only relevant for domains $\mathbb{Z}_0$ and $\mathbb{Q}_0$, since it has value zero otherwise. Similarily, parameters d and $\#d$ are only relevant for $\mathbb{Q}_+$, $\mathbb{Q}_1$ and $\mathbb{Q}_0$, since they have value one otherwise. We will use c_{min} in its inverted form $\frac{1}{c_{\text{min}}}$, which gives the same correlation between parameter value and running time as for the other parameters.

We refer to parameters k, ℓ and z as *solution parameters*, since they refer to properties of the solutions, and we refer to the other parameters as *instance parameters*, since they refer to properties of the instance. Instance parameters can be checked in advance and only influence the time complexity, not the solutions, e.g. problem COP($\mathbb{D}, \{k, c_{\text{max}}\}$) asks for a plan of cost k or less, where we guarantee that no action has a higher cost than c_{max}. While parameter k restricts the set of solutions, parameter c_{max} only matters for the time complexity, which is measured in the combined parameter $\{k, c_{\text{max}}\}$ (or equivalently $k + c_{\text{max}}$). That is, COP($\mathbb{D}, \{k\}$) and COP($\mathbb{D}, \{k, c_{\text{max}}\}$) have the same solutions, but the second problem could have a lower complexity.

We usually cannot know the value of solution parameters in advance, so they are typically constraints. This is straightforward for one parameter, e.g. Bäckström et al. (2015) study the complexity of LOP($\{\ell\}$) and Aghighi and Bäckström (2015) study the complexity of COP($\mathbb{D}, \{k\}$). Kronegger, Pfandler, and Pichler (2013) study two solution parameters for LOP, but never simultaneously.

We will consider also combinations of solution parameters, e.g. problem COP($\mathbb{D}, \{k, \ell\}$). We then have a choice whether to treat both parameters k and ℓ as optimised or not. If we optimise both, then we ask for a plan ω that satisfies both that $c(\omega) \leq k$ and that $|\omega| \leq \ell$. If we instead optimise $k+\ell$, then we ask for a plan ω such that $c(\omega) + |\omega| \leq k + \ell$. These problems are not necessarily equivalent; there may be a plan that satisfies the latter criterion, but not the former. Hence, we loose precision compared to optimising both parameters. We may also treat only one of the parameters as optimised, which is yet another problem; there may be two different plans ω_1 and ω_2 such that $c(\omega_1) \leq k$ and $|\omega_2| \leq \ell$, but no plan that satisfies both criteria simultaneously. This is a problematic approach for planning though. Suppose we optimise k but not ℓ. Then ℓ is a parameter that does not restrict the solutions but affect the complexity. However, we usually cannot know any non-trivial a priori bound for the plan length, making the parameter pointless. We will, thus, make the choice in this paper to always treat all solution parameters as optimised, noting that other choices are possible. We also choose to always include k as a parameter, although it is possible to also analyse COP with other parameters only.

5 Complexity Results for COP

Our major complexity results for COP are summarized in Figure 1, which focuses on separations. Table 1 provides more details for the main results. Only parameters that affect the complexity appear in the table, and results are only stated for those entries where the parameter combination is relevant. Note that some results are implicitly derived, for instance, hardness for domain $\mathbb{Q}_1$ implies hardness for $\mathbb{Q}_+$.

5.1 Hardness Results for COP

In order to make the hardness results as strong as possible they should hold for as many instance parameters as possible, since removing instance parameters cannot result in an easier problem. When discussing two instances simultaneously, we refer to them as $\mathbb{P}$ and $\mathbb{P}'$ and distinguish their actual parameters in the same way, i.e. parameter k refers to $\mathbb{P}$ and parameter k' refers to $\mathbb{P}'$. Furthermore, when a solution parameter is optional for a result, i.e. the result holds both with and without this parameter, we will often enclose it in parantheses. For instance, we write COP($\mathbb{Z}_+, \{k, (\ell)\}$) as a shorthand for both problems COP($\mathbb{Z}_+, \{k\}$) and COP($\mathbb{Z}_+, \{k, \ell\}$).

The following construction and lemma will be repeatedly used for hardness results.

Construction 1. *Let $\mathbb{P} = \langle V, A, I, G\rangle$ be a SAS$^+$ instance. Construct the SAS$^+$ instance $\mathbb{P}' = \langle V', A', I', G'\rangle$ such that $V' = V \cup \{v_g\}$, where $v_g \notin V$; $A' = A \cup \{a_g\}$, where $a_g : G, (v_g{=}0) \Rightarrow (v_g{=}1)$; $I'[v_g] = 0$; $I'[v] = I[v]$ for $v \neq v_g$; $G'[v_g] = 1$ and G' is otherwise undefined.*

Lemma 2. *Let $\mathbb{P}$ be a SAS$^+$ instance. (1) If ω is a plan for $\mathbb{P}$, then ω followed by a_g is a plan for $\mathbb{P}'$. (2) If ω' is a plan for $\mathbb{P}'$, then ω' with action a_g removed is a plan for $\mathbb{P}$.*

We first prove some para-**NP**-hardness results.

Theorem 3. *The following problems are* para-**NP**-*hard:*
1. *COP*($\mathbb{Z}_0, \{k, \frac{1}{c_{min}}, c_{max}, \#c\}$),
2. *COP*($\mathbb{Q}_0, \{k, \frac{1}{c_{min}}, c_{max}, \#c, d, \#d\}$).

Proof. 1. Proof by polynomial reduction from PSAT to a slice of the problem. Let $\mathbb{P} = \langle V, A, I, G\rangle$ be a SAS$^+$ instance and define $\mathbb{P}' = \langle V', A', I', G', c'\rangle$ as in Construction 1, where $c'(a_g) = 1$ and $c'(a) = 0$ for all $a \in A$. It follows from Lemma 2 that $\mathbb{P}$ has a plan if

$\mathbb{Z}_+$:	$\mathbb{Z}_0$:	$\mathbb{Q}_1$:	$\mathbb{Q}_+$:	$\mathbb{Q}_0$:	
	$k\frac{1}{c_{\min}}$		k	$k(z)$ $\quad kd\frac{1}{c_{\min}}$	para-**NP**-hard
		$k(\ell)$ $\quad k\frac{1}{c_{\min}}$	$k\ell$ $\quad k\frac{1}{c_{\min}}$	$k\ell$	in **W**[P]
$k(\ell)$	$k\ell$ $\quad kz$	$k(\ell)d$	$k(\ell)d$	$k\ell d$ $\quad kzd$	in **W**[2]

Figure 1: Major separation results for COP.

Table 1: Summary of major results for COP (non-helpful parameters are omitted).

parameters	Integer costs		Rational costs		
	$\mathbb{Z}_+$ $(c(a) \geq 1)$	$\mathbb{Z}_0$ $(c(a) \geq 0)$	$\mathbb{Q}_1$ $(c(a) \geq 1)$	$\mathbb{Q}_+$ $(c(a) > 0)$	$\mathbb{Q}_0$ $(c(a) \geq 0)$
$\{k\}$	**W**[2]-complete (Thm. 5+Thm. 8)	para-**NP**-hard (Thm. 3)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)	para-**NP**-hard (Thm. 4)	para-**NP**-hard (Thm. 4)
$\{k, \ell\}$	**W**[2]-complete (Thm. 5+Thm. 9)	**W**[2]-complete (Thm. 5+Thm. 9)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)
$\{k, z\}$	-	**W**[2]-complete (Thm. 5+Thm. 10)	-	-	para-**NP**-hard (Thm. 4)
$\{k, d\}$	-	-	**W**[2]-complete (Thm. 5+Thm. 11)	**W**[2]-complete (Thm. 5+Thm. 11)	para-**NP**-hard (Thm. 3)
$\{k, \ell, d\}$	-	-	**W**[2]-complete (Thm. 5+Cor. 12)	**W**[2]-complete (Thm. 5+Cor. 12)	**W**[2]-complete (Thm. 5+Cor. 12)
$\{k, z, d\}$	-	-	-	-	**W**[2]-complete (Thm. 5+Cor. 12)
$\{k, \frac{1}{c_{\min}}\}$	**W**[2]-complete (Thm. 5+Thm. 8)	para-**NP**-hard (Thm. 3)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)	**W**[2]-hard, in **W**[**P**] (Thm. 5+Thm. 7)	para-**NP**-hard (Thm. 3)

and only if $\mathbb{P}'$ has a plan of cost 1. Obviously, $\mathbb{P}'$ always satisfies the parameter values $k' = \frac{1}{c'_{\min}} = c'_{\max} = 1$ and $\#c' = 2$ so this is a reduction from PSAT to a slice of COP$(\mathbb{Z}_0, \{k, \frac{1}{c_{\min}}, c_{\max}, \#c\})$. Para-**NP**-hardness follows since PSAT is **NP**-hard (Bäckström and Nebel 1995, Thm. 5). 2. Analogous, setting $d = \#d = 1$. □

Theorem 4. *The following problems are* para-**NP**-*hard:*
1. *COP*$(\mathbb{Q}_+, \{k, c_{max}, \#c, \#d\})$,
2. *COP*$(\mathbb{Q}_0, \{k, (z), c_{max}, \#c, \#d\})$

Proof. 1. Proof by polynomial reduction from LOP to a slice of the problem. Let $\mathbb{I} = \langle \mathbb{P}, \ell \rangle$ be a LOP instance, where $\mathbb{P} = \langle V, A, I, G \rangle$. Let $\mathbb{P}' = \langle V', A', I', G', c' \rangle$ be as specified in Construction 1, where $c'(a_g) = 1$ and $c'(a) = 1/\ell$ for all $a \in A$. It follows from Lemma 2 that $\mathbb{P}$ has a plan of length ℓ if and only if $\mathbb{P}'$ has a plan of cost 2 or less. Obviously, $\mathbb{P}'$ always satisfies the parameter values $c'_{\max} = 1$ and $k' = \#c' = \#d' = 2$ so this is a reduction from LOP to a slice of COP$(\mathbb{Q}_+, \{k, c_{\max}, \#c, \#d\})$. Para-**NP**-hardness follows since LOP is **NP**-hard (Bäckström and Nebel 1995, Thm. 5). 2. Analogous, set $z = 0$. □

We then prove that COP is **W**[2]-hard for all cost domains, even when all relevant parameters are combined.

Theorem 5. *The following problems are* **W**[2]-*hard:*
1. *COP*$(\mathbb{Z}_+, \{k, (\ell), \frac{1}{c_{min}}, c_{max}, \#c\})$
2. *COP*$(\mathbb{Z}_0, \{k, (\ell), (z), \frac{1}{c_{min}}, c_{max}, \#c\})$
3. *COP*$(\mathbb{Q}_1, \{k, (\ell), \frac{1}{c_{min}}, c_{max}, \#c, d, \#d\})$
4. *COP*$(\mathbb{Q}_0, \{k, (\ell), (z), \frac{1}{c_{min}}, c_{max}, \#c, d, \#d\})$

Proof. 1. Proof by fpt reduction from LOP$(\{\ell\})$. Let $\mathbb{P} = \langle V, A, I, G \rangle$ be an instance of LOP$(\{\ell\})$. Construct a corresponding COP$(\mathbb{Z}_+, \{k, \ell, \frac{1}{c_{\min}}, c_{\max}, \#c\})$ instance $\mathbb{P}' = \langle V, A, I, G, c' \rangle$, i.e. $\mathbb{P}'$ is identical to $\mathbb{P}$ except for the additional cost function c'. Define c' as $c'(a) = 1$ for all $a \in A$. The parameters for $\mathbb{P}'$ are defined as $k' = \ell' = \ell$ and $\frac{1}{c'_{min}} = c'_{max} = \#c' = 1$. Clearly, $|\omega| = c'(\omega)$ for all plans ω, so $\mathbb{P}$ has a plan of length ℓ if and only if $\mathbb{P}'$ has a plan of cost k' and length ℓ'. This is thus an fpt reduction since the parameters k', ℓ', $\frac{1}{c'_{min}}$, c'_{max} and $\#c'$ of $\mathbb{P}'$ are bounded in the parameter ℓ of $\mathbb{P}$. The theorem follows since LOP$(\{\ell\})$ is **W**[2]-hard (Bäckström et al. 2015, Thm. 1). Removing parameter ℓ' does not change the solutions, so ℓ' is optional. 2–4. Analogous, set $z = 0$ and $d = \#d = 1$. □

5.2 Membership Results for COP

To make membership results strong, there should be as few instance parameters as possible, since adding instance parameters cannot result in a harder problem. We start with membership results for the class $\mathbf{W[P]}$, which can be characterised as follows (Flum and Grohe 2006, Def. 3.1).

Definition 6. A parameterised problem is in $\mathbf{W[P]}$ if it can be solved by some NTM in $f(k) \cdot n^c$ steps of which at most $h(k) \cdot \log n$ steps are non-deterministic, where f and h are computable functions, c is a constant and n the instance size.

Theorem 7. *The following problems are in* $\mathbf{W[P]}$*:*
1. *COP*$(\mathbb{Q}_0, \{k, \ell\})$,
2. *COP*$(\mathbb{Q}_+, \{k, (\ell), \frac{1}{c_{min}}\})$,
3. *COP*$(\mathbb{Q}_1, \{k, (\ell)\})$.

Proof. 1. Let n be the instance size. Guess a plan ω with $|\omega| \leq \ell$, which requires guessing at most $\ell \cdot \log n$ bits since each action can be indexed by $\log n$ bits or less. Then verify that ω is a plan, which is polynomial time in ℓ and n. Checking that $c(\omega) \leq k$ requires adding ℓ action costs and compare with k. Let $b_1, \ldots, b_m$ be all different denominators in the instance. All numbers in the instance take at most n bits in total so $\sum_1^m ||b_i|| < n$. The action costs in ω and k can be normalized by multiplying each with the factor $\alpha = \prod_1^m b_i$. We get $||\alpha|| = ||\prod_1^m b_i|| \leq \sum_1^m ||b_i|| \leq n$, so all resulting numbers will still be of size $O(n)$ bits. Hence, the check can be done in time polynomial in ℓ and n. In total, all this takes non-deterministic time $f(k, \ell) \cdot n^c$ for some computable function f and some constant c, so it follows from Def. 6 that the problem is in $\mathbf{W[P]}$. 2. Every plan ω satisfies that $c(\omega) \geq |\omega| \cdot c_{\min}$, i.e. we need to guess at most $\frac{c(\omega)}{c_{\min}} \leq \frac{k}{c_{\min}}$ actions. Hence, we can set $\ell = \frac{k}{c_{\min}}$ and use (1). 3. Immediate from (2), since $\frac{1}{c_{\min}} \leq 1$. □

We continue by membership results for $\mathbf{W}[2]$.

Theorem 8. *COP*$(\mathbb{Z}_+, \{k\})$ *is in* $\mathbf{W}[2]$.

Proof. Aghighi and Bäckström (2015, Thm. 5) prove this result for polynomially bounded action costs, using an fpt reduction from COP$(\mathbb{Z}_+, \{k\})$ to LOP$(\{\ell\})$. We note that the restriction to polynomial costs is not necessary since we can first remove all actions a such that $c(a) > k$. □

Theorem 9. *COP*$(\mathbb{Z}_0, \{k, \ell\})$ *is in* $\mathbf{W}[2]$.

Proof. Proof by fpt reduction from COP$(\mathbb{Z}_0, \{k, \ell\})$ to COP$(\mathbb{Z}_+, \{k\})$. Let $\mathbb{P} = \langle V, A, I, G, c\rangle$ be a COP$(\mathbb{Z}_0, \{k, \ell\})$ instance. Construct a corresponding COP$(\mathbb{Z}_+, \{k\})$ instance $\mathbb{P} = \langle V', A', I', G', c'\rangle$ with parameter k' as follows. Add $\ell + 1$ new binary variables $t_0, \ldots, t_\ell$, where t_0 is initially true and $t_1, \ldots, t_\ell$ are initially false. Variables $t_1, \ldots, t_\ell$ correspond to ℓ time slots, each of which can be filled with one action. Replace each action a with ℓ new actions $a^1, \ldots, a^\ell$, where $a^i : \mathrm{pre}(a), (t_{i-1}{=}1) \Rightarrow \mathrm{eff}(a), (t_{i-1}{=}0), (t_i{=}1)$ with cost $c'(a^i) = \ell \cdot c(a) + 1$, for all i $(1 \leq i \leq \ell)$, i.e., a is replaced with one copy for each slot it can occur in. Clearly, no plan can contain more than ℓ actions. Define $k' = \ell \cdot (k + 1)$.

First suppose $\mathbb{P}$ has a plan $\omega = \langle a_1, \ldots, a_n\rangle$. Then $c(\omega) \leq k$ and $n \leq \ell$. Let $\omega' = \langle a_1^1, a_2^2, \ldots, a_n^n\rangle$. We get $c'(\omega') = c'(a_1^1) + \cdots + c'(a_n^n) = (\ell \cdot c(a_1) + 1) + \cdots + (\ell \cdot c(a_n) + 1) = \ell \cdot c(\omega) + n$, but $c(\omega) \leq k$ and $n \leq \ell$ so we get $c'(\omega') \leq \ell \cdot k + \ell = k'$. Hence, ω' is a plan for $\mathbb{P}'$. Instead suppose $\mathbb{P}'$ has a plan $\omega' = \langle a_1^1, a_2^2, \ldots, a_n^n\rangle$ such that $c'(\omega') \leq k' = \ell \cdot (k + 1)$. Let $\omega = \langle a_1, \ldots, a_n\rangle$. Then $n \leq \ell$ by design of $\mathbb{P}'$ (there are ℓ slots), so this need not be verified. Suppose $c(\omega) > k$. Then $c(\omega) \geq k + 1$ so $c'(\omega') = \ell \cdot c(\omega) + n \geq \ell \cdot (k+1) + n$. Since $c'(\omega') \leq \ell \cdot (k+1)$ we get $\ell \cdot (k+1) + n \leq c'(\omega') \leq \ell \cdot (k+1)$, but then $n = 0$ so ω' is an empty plan with non-zero cost, which is impossible. We conclude that $c(\omega) \leq k$ and, thus, that ω is a plan for $\mathbb{P}$.

It follows that $\mathbb{P}$ has a plan of maximum cost k and length ℓ if and only if $\mathbb{P}$ has a plan of maximum cost $k' = \ell \cdot (k+1)$. Furthermore, $\mathbb{P}'$ can be constructed in time $f(k, \ell) \cdot ||\mathbb{P}'||^c$, for some computable function f and constant c, and k' is bounded in k and ℓ, so this is an fpt reduction. The theorem follows since COP$(\mathbb{Z}_+, \{k\})$ is in $\mathbf{W}[2]$ by Thm. 8. □

Theorem 10. *COP*$(\mathbb{Z}_0, \{k, z\})$ *is in* $\mathbf{W}[2]$.

Proof. Analogous to the proof of Thm. 9. Introduce variables $t_0, \ldots, t_z$ but replace only the zero-cost actions with new actions. Then every plan is limited to at most z zero-cost actions but the total length is not explicitly restricted. Define $c'(a^i) = 1$ if $c(a) = 0$ and $c'(a) = (z + 1)c(a)$ otherwise. Define $k' = z + zk + k$. First suppose $\mathbb{P}$ has a plan ω with n actions of which m are zero-cost actions. The corresponding plan ω' also has n actions and m zero-cost actions, so $c'(\omega') = m + (z + 1)c(\omega) \leq z + (z + 1)k = k'$. Instead suppose $\mathbb{P}'$ has a plan ω' with n actions and m zero-cost actions, such that $c'(\omega') \leq k' = z + zk + k$. Since $m \leq z$ by design of $\mathbb{P}'$, this need not be verified. Let ω be the corresponding plan for $\mathbb{P}$. Suppose $c(\omega) > k$. Then $c(\omega) \geq k + 1$ so $c'(\omega') = m + (z + 1)c(\omega) \geq m + (z + 1)(k + 1) = m + zk + z + k + 1$. We also know that $c'(\omega') \leq z + zk + k$, so we get $m + zk + z + k + 1 \leq z + zk + k$, which is impossible. We conclude that $c(\omega) \leq k$. It follows that $\mathbb{P}$ has a plan of maximum cost k with at most z zero-cost actions if and only if $\mathbb{P}'$ has a plan of maximum cost $k' = z + (z + 1)k$. Furthermore, $\mathbb{P}'$ can be constructed in time $f(k, z) \cdot ||\mathbb{P}'||^c$, for some computable function f and constant c, and k' is bounded in k and z, so this is an fpt reduction. The theorem follows since COP$(\mathbb{Z}_+, \{k\})$ is in $\mathbf{W}[2]$ by Thm. 8. □

Theorem 11. *COP*$(\mathbb{Q}_+, \{k, d\})$ *is in* $\mathbf{W}[2]$.

Proof. Proof by fpt reduction from COP$(\mathbb{Q}_+, \{k, d\})$ to COP$(\mathbb{Z}_+, \{k\})$. Let $\mathbb{P} = \langle V, A, I, G, c\rangle$ be a COP$(\mathbb{Q}_+, \{k, d\})$ instance with parameters k and d. Construct a corresponding COP$(\mathbb{Z}_+, \{k\})$ instance $\mathbb{P} = \langle V, A, I, G, c'\rangle$ with parameter k' as follows, i.e. the instances are identical except for the cost function. Let $C = \{c(a) \mid a \in A\} = \{\frac{a_1}{b_1}, \ldots, \frac{a_n}{b_n}\}$ be all the different action costs in $\mathbb{P}$. Let $b'_1, \ldots, b'_m$ be all different denominators occurring in C, i.e. $m \leq n$. Define the product $\alpha = b'_1 \cdot b'_2 \cdot \ldots \cdot b'_m$ and define c' such that $c'(a) = \alpha \cdot c(a)$ for all $a \in A$. Also define $k' = \alpha \cdot k$. Then $c'(a) \in \mathbb{Z}_+$ for all $a \in A$ and $k' \in \mathbb{Z}_+$. Since $b_i \leq d$ for all i, we get $m \leq d$, so $\alpha \leq d^m \leq d^d$. This is an fpt reduction since the new parameter $k' = \alpha \cdot k \leq d^d k$ is bounded in k and d. The theorem follows since COP$(\mathbb{Z}_+, \{k\})$ is in $\mathbf{W}[2]$ by Thm. 8. □

Note that this reduction is *not* an fpt reduction from COP$(\mathbb{Q}_+, \{k\})$ to COP$(\mathbb{Z}_+, \{k\})$, since k' is not bounded in k alone, even though it is a polynomial reduction.

Corollary 12. *The following problems are in* $\mathbf{W}[2]$*:*
1. *COP*$(\mathbb{Q}_0, \{k, d, \ell\})$, 2. *COP*$(\mathbb{Q}_0, \{k, d, z\})$.

Proof. Do the reduction in the proof of Thm. 11, but let all zero costs remain. Then all costs are in $\mathbb{Z}_0$. Apply either of the reductions in the proofs of Thms. 9 and 10. □

5.3 Some Explicit Time Bounds for COP

That COP$(\mathbb{Q}_+, \{k\})$ is para-**NP**-hard, but COP$(\mathbb{Q}_+, \{k, d\})$ is in $\mathbf{W}[2]$ does not mean that the problem gets easier to solve by adding parameter d; we always know d in advance. This separation in complexity rather indicates that parameter d is important and influences the actual running time of algorithms. In order to give more intuition for this, we derive some explicit time complexity bounds. We first demonstrate two straightforward upper bounds.

Theorem 13.
1. *COP*$(\mathbb{Z}_+, \{k\})$ *can be solved in time* $O(||\mathbb{P}||^k)$
2. *COP*$(\mathbb{Q}_+, \{k, d\})$ *can be solved in time* $O(||\mathbb{P}||^{kd^d})$.

Proof. 1. Let $n = ||\mathbb{P}||$. We have $|\omega| \leq k$ for all plans, so guess at most k actions and verify that it is a plan. We need to guess at most $k \log n$ bits, which takes deterministic time $O(2^{k \log n}) = O(n^k)$. Verifying a plan of length k takes time $O(k \cdot n^c)$ for some small constant c. The total time is $O(n^k + k \cdot n^c)$, which is $O(n^k)$ for $k \geq c$. 2. First use the reduction in the proof of Thm. 11, then apply (1). □

We can also show that COP for $\mathbb{Q}_+$ is strictly harder than for $\mathbb{Z}_+$ by the following lower bound.

Theorem 14. *COP*$(\mathbb{Q}_+, \{k\})$ *cannot be solved in time* $O(||\mathbb{P}||^{ck})$ *for any* $c > 0$*, unless* $\mathbf{P} = \mathbf{NP}$.

Proof. Suppose there is a c such that COP$(\mathbb{Q}_+, \{k\})$ can be solved in time $O(||\mathbb{P}||^{ck})$. Let $\mathbb{I}$ be a 3-SAT instance with n variables and m clauses. Without losing generality, assume that $n \leq m$, since 3-SAT is still **NP**-complete. Make a standard reduction from 3-SAT to LOP, where an optimal plan contains two actions for each variable, and one action for each clause (cf. Bylander (1994), proof of Thm. 4.2). Let the former actions have cost $\frac{1}{n}$ and the latter cost $\frac{1}{m}$. Set $k = 3$. An optimal plan ω is then of length $|\omega| = 2n + m$ and have cost $c(\omega) = 2n \cdot \frac{1}{n} + m \cdot \frac{1}{m} = 3$. Hence, $\mathbb{I}$ is satisfiable if and only $\mathbb{P}$ has a plan of cost 3, so this is a polynomial reduction from 3-SAT to COP. We also have that $||\mathbb{P}|| \leq ||\mathbb{I}||^a$ for some constant a. This means we can solve 3-SAT in time $O(||\mathbb{P}||^{3c})$, i.e. in time $O(||\mathbb{I}||^{3ac})$. However, this means that $\mathbf{P} = \mathbf{NP}$. □

This theorem uses only parameter k, but we note that if adding also parameter d we would need to set $d = m$ in the proof. In other words, the reason that COP$(\mathbb{Q}_+, \{k, d\})$ has a lower complexity than COP$(\mathbb{Q}_+, \{k\})$ is that we may need very large values of d, even if k is small. Choosing a smaller d value in the proof would require a larger k value; if ω is an optimal plan, then $c(\omega) \geq \frac{2n}{d} + \frac{m}{d}$ so we must set $k \geq \frac{2n+m}{d}$. In a more extensive analysis, one could attempt to derive sharper lower bounds, or even so-called **XP** optimal bounds (Downey and Thilikos 2011).

6 Complexity Results for Net-benefit Planning

The *net-benefit problem* (van den Briel et al. 2004) is a so called *oversubscription* problem, where we do not expect to satisfy all of the goal. Instead each goal variable v has a utility value $u(v)$, which is the reward if the goal value is satisfied for v. We generalise this problem to SAS$^+$ as follows, but no complexity result depends on using non-binary variables. The utility of a state s is $u(s) = \sum_{v \in V'} u(v)$, where $V' = \{v \in \text{vars}(G) \mid s[v] = G[v]\}$. If ω is a plan from I to s, then the *net benefit* of ω is the difference $u(s) - c(\omega)$. The objective of the net-benefit problem is to maximise the net benefit over all plans to all states.

NET-BENEFIT PLANNING (NBP($\mathbb{D}$))
Instance: A SAS$^+$ instance $\mathbb{P} = \langle V, A, I, G \rangle$, a cost function $c : A \to \mathbb{D}$, a utility function $u : \text{vars}(G) \to \mathbb{D}$ and a value $b \in \mathbb{D}$.
Question: Is there a state s and a plan ω from I to s such that $u(s) - c(\omega) \geq b$?

We write the parameterised version as NBP$(\mathbb{D}, \pi)$, where π may contain all previously defined parameters and the following additional ones:
b: Min. net benefit of the plan.
u_{min}: Min. variable utility in instance.
u_{max}: Max. variable utility in instance.
$\#u$: Number of different utility values in instance.
t: Sum of all utilities in instance, i.e. $t = \sum_{v \in \text{vars}(G)} u(v)$.
Since the net benefit is the primary objective to optimise in NBP, we choose to always inlcude parameter b, just as we choose to always include parameter k for COP. Parameter d is reinterpreted as the maximum denominator of all numbers, i.e. both action costs and utilities. While maximising the net benefit, b, is the main objective of NBP, it is sometimes combined with a restriction on the plan cost, k, (cf. (Mirkis and Domshlak 2013)) suggesting a multi-objective optimisation of the type we use for COP.

Our major results for NBP are summarised in Figure 2.

6.1 Hardness Results for NBP

We first prove some para-**NP**-hardness results.

Theorem 15. *The following problems are* para-**NP**-*hard:*
1. *NBP*$(\mathbb{Z}_+, \{b, \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, \#u\})$
2. *NBP*$(\mathbb{Z}_0, \{b, (z), \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, \#u\})$
3. *NBP*$(\mathbb{Q}_1, \{b, \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, \#u, d, \#d\})$
4. *NBP*$(\mathbb{Q}_0, \{b, (z), \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, \#u, d, \#d\})$

Proof. 1. Proof by polynomial reduction from LOP to a slice. Let $\mathbb{I} = \langle \mathbb{P}, \ell \rangle$ be a LOP instance, where $\mathbb{P} = \langle V, A, I, G \rangle$. Let $\mathbb{P}' = \langle V', A', I', G', c', u' \rangle$ be as in Construction 1, where $c'(a) = 1$ for all $a \in A'$ and $u(v_g) = \ell + 2$. It follows from Lemma 2 that $\mathbb{P}$ has a plan of length ℓ if and only if $\mathbb{P}'$ has a plan of cost $\ell + 1$, i.e. if $\mathbb{P}'$ has a plan with net benefit 1. Obviously, $\mathbb{P}'$ always satisfies the parameter values $b' = \frac{1}{c_{\text{min}}} = c'_{\text{max}} = \#c' = \#u' = 1$ and $u_{\text{min}} = 2$ so this is a reduction from LOP to

$\mathbb{Z}_+$:	$\mathbb{Z}_0$:	$\mathbb{Q}_1$:	$\mathbb{Q}_+$:	$\mathbb{Q}_0$:	
$b\frac{1}{c_{\min}}$	$b(z)\frac{1}{c_{\min}}$ $\;$ $b(k)t\frac{1}{c_{\min}}$	$bd\frac{1}{c_{\min}}$	$bd\frac{1}{c_{\min}}$ $\;$ $b(k)t$	$b(z)d\frac{1}{c_{\min}}$ $\;$ $b(k)(z)t$ $\;$ $b(k)dt\frac{1}{c_{\min}}$	para-**NP**-hard
$b\ell$	$b\ell$ $\;$ bkz	$b\ell$ $\;$ bk	$b\ell$ $\;$ $bk\frac{1}{c_{\min}}$ $\;$ bkd	$b\ell$ $\;$ $bkzd$	in **W**[P]
$b(k)t$	$b(k)\ell t$ $\;$ $bkzt$	$bkdt$	$bkdt$	$b(k)\ell dt$ $\;$ $bkzdt$	in **W**[2]

Figure 2: Major separations for NBP.

a slice of NBP($\mathbb{Z}_+, \{b, \frac{1}{c_{\min}}, c_{\max}, \#c, \frac{1}{u_{\min}}, \#u\}$). Para-**NP**-hardness follows since LOP is **NP**-hard (Bäckström and Nebel 1995, Thm. 5). 2-4. Set $z = 0$ and $d = \#d = 1$. $\square$

Theorem 16. *The following problems are* para-**NP**-*hard:*
1. *NBP*($\mathbb{Q}_+, \{b, (k), c_{max}, \#c, \#d, \frac{1}{u_{min}}, u_{max}, \#u, t\}$)
2. *NBP*($\mathbb{Q}_0, \{b, (k), (z), c_{max}, \#c, \#d, \frac{1}{u_{min}}, u_{max}, \#u, t\}$)

Proof. 1. Proof by polynomial reduction from PSAT to a slice. Let $\mathbb{P} = \langle V, A, I, G\rangle$ be a SAS$^+$ instance. Let $\mathbb{P}' = \langle V', A', I', G', c', u'\rangle$ be as in Construction 1, where $c'(a_g) = 1$, $c'(a) = 1/2^{|V|}$ for all $a \in A$ and $u(v_g) = 3$. Since no optimal plan has more than $2^{|V|}$ actions, it follows from Lemma 2 that $\mathbb{P}$ has a plan if and only if $\mathbb{P}'$ has a plan of cost 2 or less, i.e. with net benefit 1 or more. Obviously, $\mathbb{P}'$ always satisfies the parameter values $b' = c'_{\max} = \#u' = 1$, $\#c' = 2$ and $u'_{\min} = u'_{\max} = t' = 3$ so this is a reduction from PSAT to a slice of NBP($\mathbb{Q}_+, \{b, (k), c_{\max}, \#c, \#d, \frac{1}{u_{\min}}, u_{\max}, \#u, t\}$). Para-**NP**-hardness follows since PSAT is **NP**-hard (Bäckström and Nebel 1995, Thm. 5). 2. Set $z = 0$. $\square$

Theorem 17. *The following problems are* para-**NP**-*hard:*
1. *NBP*($\mathbb{Z}_0, \{b, (k), \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, u_{max}, \#u, t\}$)
2. *NBP*($\mathbb{Q}_0, \{b, (k), \frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, u_{max}, \#u, t, d, \#d\}$)

Proof sketch. Analogous to Thm. 16, but define $c'(a_g) = 1$, $c'(a) = 0$ for all $a \in A$ and $u(v_g) = 2$. Then $\mathbb{P}$ has a plan if and only if $\mathbb{P}'$ has a plan of cost 1, i.e. with net benefit 1. Set parameter values parameter values $b' = k' = \frac{1}{c_{\min}}' = c'_{\max} = \#u' = 1$ and $\#c' = u'_{\min} = u'_{\max} = t' = 2$. $\square$

We continue by hardness results for **W**[2].

Theorem 18. *The following problems are* **W**[2]-*hard:*
1. *NBP*($\mathbb{Z}_+, \{b, (\ell), (k), t\} \cup \sigma$)
2. *NBP*($\mathbb{Z}_0, \{b, (\ell), (k), (z), t\} \cup \sigma$)
3. *NBP*($\mathbb{Q}_1, \{b, (\ell), (k), t, d, \#d\} \cup \sigma$)
4. *NBP*($\mathbb{Q}_+, \{b, (\ell), (k), t, d, \#d\} \cup \sigma$)
5. *NBP*($\mathbb{Q}_0, \{b, (\ell), (k), (z), t, d, \#d\} \cup \sigma$)

where $\sigma = \{\frac{1}{c_{min}}, c_{max}, \#c, \frac{1}{u_{min}}, u_{max}, \#u\}$

Proof. 1. Proof by fpt reduction from LOP($\{l\}$). Let $\mathbb{I} = \langle \mathbb{P}, \ell\rangle$ be a LOP($\{l\}$) instance, where $\mathbb{P} = \langle V, A, I, G\rangle$. Let $\mathbb{P}' = \langle V', A', I', G', c', u'\rangle$ be as specified in Construction 1, where $c'(a) = 1$ for all $a \in A'$ and $u(v_g) = \ell + 2$. It follows from Lemma 2 that $\mathbb{P}$ has a plan of length ℓ if and only if $\mathbb{P}'$ has a plan of length $\ell + 1$, cost $\ell + 1$ and net benefit 1. That is, $\mathbb{P}'$ satisfies the parameter values $b' = 1$, $k' = \ell + 1$ and $\ell' = \ell + 1$. It furthermore satisfies the parameter values $\frac{1}{c_{\min}}' = c'_{\max} = \#c' = \#u' = 1$ and $u_{\min} = u_{\max} = t = \ell + 2$. Hence, all new parameter values are bounded in ℓ, so this is an fpt reduction from LOP($\{l\}$) to NBP($\mathbb{Z}_+, \{b, (\ell), (k), t\} \cup \sigma$). The theorem follows since LOP($\{l\}$) is **W**[2]-hard (Bäckström et al. 2015, Thm. 1). 2–5. Set $z = 0$ and $d = \#d = 1$. $\square$

6.2 Membership Results for NBP

Just as for COP, bounding the plan length explicitly or implicitly is sufficient for membership in **W**[**P**].

Theorem 19. *The following problems are in* **W**[**P**]*:*
1. *NBP*($\mathbb{Q}_0, \{b, \ell\}$),
2. *NBP*($\mathbb{Q}_0, \{b, k, z, d\}$),
3. *NBP*($\mathbb{Q}_+, \{b, k, d\}$),
4. *NBP*($\mathbb{Q}_+, \{b, k, \frac{1}{c_{min}}\}$),
5. *NBP*($\mathbb{Q}_1, \{b, k\}$),
6. *NBP*($\mathbb{Z}_0, \{b, k, z\}$).

Proof sketch. 1. Analogous to the proof of Thm. 7(1). 2. Let ω be a plan and let ω^+ be ω with all zero-cost actions removed. We have $c_{\min} \geq 1/d$, so $c(\omega^+) \geq |\omega^+|\frac{1}{d}$, i.e. $|\omega^+| \leq c(\omega^+) \cdot d \leq kd$. Hence, $|\omega| \leq kd + z$, so we can set $\ell = kd + z$ and use (1). 3. Immediate from (2) since $z = 0$. 4. Immediate from (3) since $\frac{1}{c_{\min}} \leq d$. 5. Immediate from (4) since $\frac{1}{c_{\min}} \leq 1$. 6. Immediate from (2) since $d = 1$. $\square$

We finally prove membership results for **W**[2], which all rely on parameter t and use the following reduction[1].

Theorem 20. *NBP*($\mathbb{D}$,$\{b, t\}$) $\leq_{fpt}$*COP*($\mathbb{D}$,$\{k\}$).

Proof. Let $\mathbb{I} = \langle \mathbb{P}, b, t\rangle$ be an NBP($\mathbb{D}$,$\{b, t\}$) instance, where $\mathbb{P} = \langle V, A, I, G, c, u\rangle$ Construct a COP($\mathbb{D}, \{k\}$) instance $\mathbb{I}' = \langle \mathbb{P}', k'\rangle$ as follows. Let $\mathbb{P}' = \langle V', A', I, G, c'\rangle$, where $V' = V \cup \{w\}$; A' contains the following actions:

a' : pre(a), $(w = 0) \Rightarrow$eff(a) for each $a \in A$,
a_w : $(w = 0) \Rightarrow (w = 1)$,
a_v : $(w = 1) \Rightarrow (v = G[v])$ for each $v \in$ vars(G);

$I'[w] = 0$ and $I'[v] = I[v]$ for $v \in V$; $G'[w] = 1$ and $G'[v] = G[v]$ for $v \in V$. Define $c'(a) = c(a)$ for $a \in A$, $c'(a_w) = 1$ and $c'(a_v) = u(v)$ for all $v \in V_u$. Define $k' = t - b + 1$. We claim that $\mathbb{P}$ has a plan with net benefit at least b if and only if $\mathbb{P}'$ has a plan of cost at most k'.

$\Rightarrow$: Suppose $\mathbb{P}$ has a plan ω from I to some state s with net benefit b, i.e. $u(s) - c(\omega) = b$. Let $V'' = \{v \in \text{vars}(G) \mid s[v] \neq G[v]\}$, i.e. V'' are all goal variables that do not have the goal value in s. Then $u(s) = \sum_{v \in \text{vars}(G) \setminus V''} u(v)$. Let ω' be the plan ω followed by a_w and the actions a_v for each $v \in V''$ in arbitrary order. Then ω' is a plan for $\mathbb{P}'$ and $c'(\omega') = c'(\omega) + 1 + \sum_{v \in V''} c'(a_v) =$

[1]Keyder and Geffner (2009) used a similar reduction, but in contrast to ours, it relied on zero-cost actions.

$c(\omega) + 1 + \sum_{v \in V''} u(v) = c(\omega) + (t - u(s)) + 1 = t - (u(s) - c(\omega)) + 1 = t - b + 1 = k'$.

$\Leftarrow$: Suppose $\mathbb{P}'$ has a plan ω' such that $c(\omega') \leq k'$. Let V'' be all variables v such that action a_v occurs in ω'. Let ω be the subsequence of ω' containing only actions from A. We get $c(\omega) = c'(\omega) = c'(\omega') - 1 - \sum_{v \in V''} u(v)$ and $u(\omega) = \sum_{v \in \text{vars}(G) \setminus V''} u(v) = t - \sum_{v \in V''} u(v)$. Hence, the net benefit of ω is $u(\omega) - c(\omega) = (t - \sum_{v \in V''} u(v)) - (c'(\omega') - 1 - \sum_{v \in V''} u(v)) = t - c'(\omega') + 1$, but $c'(\omega') \leq k' = t - b + 1$ so $t - c'(\omega') + 1 \geq t - (t - b + 1) + 1 = b$. □

Corollary 21. *The following problems are in* $\mathbf{W}[2]$*:*

1. *NBP*$(\mathbb{Z}_+, \{b, (k), t\})$
2. *NBP*$(\mathbb{Z}_0, \{b, (k), \ell, t\})$
3. *NBP*$(\mathbb{Z}_0, \{b, k, z, t\})$,
4. *NBP*$(\mathbb{Q}_+, \{b, k, d, t\})$,
5. *NBP*$(\mathbb{Q}_0, \{b, k, z, d, t\})$,
6. *NBP*$(\mathbb{Q}_0, \{b, (k), \ell, d, t\})$.

Proof sketch. Combine Thm. 20 with Thms. 8, 9, 10, 11 and Cor. 12. The reduction works since $b, k \leq t$, $\ell' \leq \ell + t + 1$, no additional zero-cost actions are required for solving $\mathbb{P}'$ (since a zero utility does not contribute to the net benefit) and d does not change. Note that the original k and ℓ values are not preserved here, so only b is optimised, although ℓ could be optimised using the technique from Thm. 9. □

7 Discussion

It is known that COP with zero-cost actions can be problematic in practice since a cost-optimal plan can contain a very large number of such actions. One method to tackle this problem is to somehow also take the plan length into account (Richter and Westphal 2010; Benton et al. 2010). This is a practical approach that is consistent with our findings that parameters that limit the plan length explicitly or implicitly reduce the complexity. We have seen that adding the plan length, ℓ, explicitly makes COP easier for all domains (except $\mathbb{Z}_+$). For $\mathbb{Z}_0$, this can also be achieved by adding the plan length implicitly as combination $\{k, z\}$. Zero-cost actions is thus a case where the theoretical results seem to correlate well with practical experience and intuition.

A set of goal states can be simulated by adding zero-cost actions from these states to a single goal state (cf. Yang et al. 2008). We can answer an open question by Aghighi and Bäckström (2015) whether it is safe to do so? The answer is yes, the complexity will not increase since z is bounded.

Aghighi and Bäckström (2015) suggested the use of a linear combination $c'(a) = \lambda \cdot c(a) + b$, for some positive integer constants λ and b, as a pragmatic type of approximation in practice. This transforms a COP$(\mathbb{Z}_0)$ instance to a COP$(\mathbb{Z}_+)$ instance, which can be solved more efficiently at the expense of overestimating the optimal solutions. This is probably best described as a transformation (but *not* an fpt reduction) from COP$(\mathbb{Z}_0, \{k\})$ to COP$(\mathbb{Z}_+, \{k + \ell\})$. A related technique is used in LAMA, with a heuristic that puts equal weight to the length and the cost of the plan (Richter and Westphal 2010). This uses both parameters k and ℓ, but only k is optimised so ℓ only influences the efficiency via the heuristic. This can thus be viewed as problem COP$(\mathbb{Z}_0, \{k, \ell\})$, but where only k is treated as optimised, i.e. ℓ is not a constraint and we do not know a value for it in advance. While it might seem unconventional to measure the complexity in a property of the solution that we cannot know in advance, this is not new. For instance, an algorithm is said to run in polynomial total time if it runs in polynomial time in the sum of the input size and the output size (Johnson, Papadimitriou, and Yannakakis 1988).

Cushing, Benton, and Kambhampati (2010) as well as Wilt and Ruml (2011) have further shown that COP is very difficult for common heuristic search algorithms, even for strictly positive action costs. They argued that the difficulty arises when there is a big span, or ratio, between the maximum and minimum action costs. While this may be a correct analysis of the particular algorithms, our results indicate that it is not a universal truth for all conceivable algorithms. The COP problem for positive integers is no harder than for unit cost, whatever span or ratio in the costs. The difficulties arise with rational costs, where the minimum cost matters. The maximum cost does not matter, though, and thus neither the span nor the ratio. What matters even more than the minimum cost is the largest cost denominator, even if there are no costs with value lower than 1, which suggests that it is the distribution of rational costs rather than the ratio between maximum and minimum cost that is important. The reason seems to be that the combination $\{k, d\}$ bounds the plan length for $\mathbb{Q}_+$ and $\mathbb{Q}_1$. In practice, we always know the value of d from the instance, so the difference in complexity with and without parameter d should be interpreted as an indication that the actual value of d can have a significant influence on the running time of actual algorithms. As we have seen in Sec. 5.3, we cannot solve COP for positive rationals as efficiently as for positive integers, even if taking the value of d into account. We noted there that the values of k and d are not independent. It should be further noted that while both paramater combinations $\{k, d\}$ and $\{k, \frac{1}{c_{\text{min}}}\}$ reduce the complexity of COP for $\mathbb{Q}_+$, the latter one may not be as effective; the first combination guarantees membership in $\mathbf{W}[2]$ but second one only guarantees membership in $\mathbf{W}[\mathbf{P}]$. It should be noted, of course, that we have not proven a strict separation between the two cases, so it is possible that COP is in $\mathbf{W}[2]$ also for combination $\{k, \frac{1}{c_{\text{min}}}\}$. Furthermore, Cushing, Benton, and Kambhampati (2010) transform all costs to the interval [0,1], i.e. divide them by c_{max}, before doing their analysis. The differences in complexity between domains $\mathbb{Q}_1$ and $\mathbb{Q}_+$ suggests that this transformation could, perhaps, introduce artificial difficulties. To shed more light on these issues, it would be useful to derive more explicit bounds of the type in Section 5.3.

Finally, our complexity results refer to worst-case complexity and it is possible that also other parameters can have an impact on running time in many practical cases. However, this is diffuclt to analyse theoretically without having a formal characterization of such practical cases, so one would typically have to resort to empirics for this.

For NBP, we see that parameter b, the net-benefit, which is the main objective, is not a very helpful parameter for reducing the complexity (as was observed already by Aghighi and Bäckström (2015)). Intuitively, the reason for this is that b does not bound neither k nor t; a plan may have both a very large cost and a very large utility, but a very small difference between these. While parameters and combinations like ℓ, kz and d that reduce the complexity of COP often do so also for NBP, the effect is not so large. We see that it is often necessary to also add parameter t to achieve membership in $\mathbf{W}[2]$, while parameter k has no similar effect. This can be understood in the following way. If we can achieve most of the goals, then the sum $k + b$ is close to t, but for problems where we can only achieve very few goals there will be a large difference, so t is much larger than $k + b$ and has a larger influence on the time needed to find a solution.

In SAT planning (Kautz and Selman 1992; Ghallab, Nau, and Traverso 2004), a number of time slots for actions are fixed in advance (as in the proof of Thm. 9). If doing this for COP we get the problem COP($\mathbb{D}, \{k, \ell\}$) that we study, since both k and ℓ are strict limits on the plans. It is also common to allow two or more actions in parallel in a time slot, if they do not interfere with each other (Kautz, McAllester, and Selman 1996). Then the number of time slots is no longer the number of actions in the plan, but its shortest parallel execution length (aka. makespan). This could be an interesting parameter, since the two measures are not monotonically related; a parallel plan with shortest makespan is not always a plan with the smallest number of actions (Bäckström 1994).

Acknowledgments

Aghighi is partially supported by the National Graduate School in Computer Science (CUGS), Sweden. Bäckström is partially supported by the Swedish Research Council (VR) under grant 621-2014-4086. Sebastian Ordyniak and the anonymous reviewers provided important feedback.

References

Aghighi, M., and Bäckström, C. 2015. Cost-optimal and net-benefit planning - A parameterised complexity view. In *Proc. 24th Int'l Joint Conf. Artif. Intell. (IJCAI-15), Buenos Aires, Argentina*, 1487–1493.

Aghighi, M., and Jonsson, P. 2014. Oversubscription planning: Complexity and compilability. In *Proc. 28th AAAI Conf. Artif. Intell. (AAAI-14), Québec City, QC, Canada.*, 2221–2227.

Bäckström, C., and Nebel, B. 1995. Complexity results for SAS$^+$ planning. *Comput. Intell.* 11:625–656.

Bäckström, C.; Jonsson, P.; Ordyniak, S.; and Szeider, S. 2015. A complete parameterized complexity analysis of bounded planning. *J. Comput. Syst. Sci.* 81(7):1311–1332.

Bäckström, C. 1994. Executing parallel plans faster by adding actions. In *Proc. 11th European Conf. Artif. Intell. (ECAI-94), Amsterdam, Netherlands*, 615–619.

Benton, J.; Talamadupula, K.; Eyerich, P.; Mattmüller, R.; and Kambhampati, S. 2010. G-value plateaus: A challenge for planning. In *Proc. 20th Int'l Conf. Automated Planning and Scheduling, (ICAPS-10), Toronto, ON, Canada*, 259–262.

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

Cooper, M. C.; de Roquemaurel, M.; and Régnier, P. 2011. A weighted CSP approach to cost-optimal planning. *AI Commun.* 24(1):1–29.

Cushing, W.; Benton, J.; and Kambhampati, S. 2010. Cost based search considered harmful. In *Proc. 3rd Ann. Symp. Combinatorial Search (SoCS-10), Stone Mountain, Atlanta, GA, USA*, 140–141. Long ver.: CoRR abs/1103.3687, 2011.

Downey, R., and Fellows, M. 1999. *Parameterized Complexity*. New York: Springer.

Downey, R. G., and Thilikos, D. M. 2011. Confronting intractability via parameters. *Computer Science Review* 5(4):279–317.

Fellows, M. R.; Gaspers, S.; and Rosamond, F. A. 2012. Parameterizing by the number of numbers. *Theory Comput. Syst.* 50(4):675–693.

Flum, J., and Grohe, M. 2006. *Parameterized Complexity Theory*. Berlin: Springer.

Ghallab, M.; Nau, D. S.; and Traverso, P. 2004. *Automated planning - theory and practice*. Elsevier.

Johnson, D. S.; Papadimitriou, C. H.; and Yannakakis, M. 1988. On generating all maximal independent sets. *Inf. Process. Lett.* 27(3):119–123.

Kautz, H. A., and Selman, B. 1992. Planning as satisfiability. In *Proc. 10th European Conf. Artif. Intell. (ECAI-92), Vienna, Austria*, 359–363.

Kautz, H. A.; McAllester, D. A.; and Selman, B. 1996. Encoding plans in propositional logic. In *Proc. 5th Int'l Conf. Principles of Knowledge Repr. and Reasoning (KR-96), Cambridge, MA, USA*, 374–384.

Keyder, E., and Geffner, H. 2009. Soft goals can be compiled away. *J. Artif. Intell. Res.* 36:547–556.

Kronegger, M.; Pfandler, A.; and Pichler, R. 2013. Parameterized complexity of optimal planning: A detailed map. In *Proc. 23rd Int'l Joint Conf. Artif. Intell. (IJCAI-13), Beijing, China*, 954–961.

Likhachev, M., and Ferguson, D. 2009. Planning long dynamically feasible maneuvers for autonomous vehicles. *I. J. Robotic Res.* 28(8):933–945.

Mirkis, V., and Domshlak, C. 2013. Abstractions for oversubscription planning. In *Proc. 23rd Int'l Conf. Aut. Planning and Scheduling (ICAPS-13), Rome, Italy*, 153–161.

Richter, S., and Westphal, M. 2010. The LAMA planner: Guiding cost-based anytime planning with landmarks. *J. Artif. Intell. Res.* 39:127–177.

van den Briel, M.; Nigenda, R. S.; Do, M. B.; and Kambhampati, S. 2004. Effective approaches for partial satisfaction (over-subscription) planning. In *Proc. 19th Nat'l Conf. Artif. Intell. (AAAI-04), San Jose, CA, USA*, 562–569.

Wilt, C. M., and Ruml, W. 2011. Cost-based heuristic search is sensitive to the ratio of operator costs. In *Proc. 4th Ann. Symp. Combinatorial Search (SoCS-11), Castell de Cardona, Barcelona, Spain*, 172–179.

Yang, F.; Culberson, J. C.; Holte, R.; Zahavi, U.; and Felner, A. 2008. A general theory of additive state space abstractions. *J. Artif. Intell. Res.* 32:631–662.