A Multi-parameter Complexity Analysis of Cost-optimal and Net-benefit Planning

Meysam Aghighi and Christer Bäckström

Linköping University

{*meysam.aghighi,christer.backstrom*}*@liu.se*

June 17, 2016

Overview – Length Optimal Planning

Length Optimal Planning (LOP): Asks for a plan of bounded length.

Meysam Aghighi and Christer Bäckström

Overview – Length Optimal Planning

Length Optimal Planning (LOP): Asks for a plan of bounded length.

- Standard complexity: **PSPACE**-complete (Bylander, AIJ-1994)

Overview – Length Optimal Planning

Length Optimal Planning (LOP): Asks for a plan of bounded length.

- Standard complexity: **PSPACE**-complete (Bylander, AIJ-1994)
- Parameterised complexity: **W**[2]-complete with plan length as parameter (Bäckström, et. al., AAAI-2012 and JCSS-2015)

Meysam Aghighi and Christer Bäckström

Overview – Cost Optimal Planning

Cost Optimal Planning (COP): Asks for a plan of bounded cost.

Overview – Cost Optimal Planning

Cost Optimal Planning (COP): Asks for a plan of bounded cost.

- Often more difficult than LOP in practice, at least when using zero-cost or rational-cost actions.

Overview – Cost Optimal Planning

Cost Optimal Planning (COP): Asks for a plan of bounded cost.

- Often more difficult than LOP in practice, at least when using zero-cost or rational-cost actions.
- Standard complexity analysis cannot distinguish this: COP is **PSPACE**-complete for all types of costs.

Overview – Cost Optimal Planning

Cost Optimal Planning (COP): Asks for a plan of bounded cost.

- Often more difficult than LOP in practice, at least when using zero-cost or rational-cost actions.
- Standard complexity analysis cannot distinguish this: COP is **PSPACE**-complete for all types of costs.
- Parameterised Complexity analysis can be used instead. With plan cost as parameter, we have:
  - COP is **W**[2]-complete for positive integer costs (i.e. the same complexity as LOP)
  - COP is para-**NP**-hard if the costs are non-negative integers or positive rationals (i.e. much harder than LOP).

  (Aghighi and Bäckström, IJCAI-2015)

Overview – This Paper

In this paper, we refine our previous analysis by:

- Using many other parameters in combination with plan cost
- Analysing more cost domains

Parameterised Complexity Theory

Standard Complexity: Measures time complexity as a function $t(n)$, where n is the input size

The complexity depends only on the instance size.

Parameterised Complexity: Measures time complexity as a function $t(n, k)$, where n is the input size and

k is some parameter we can choose.

The complexity depends both on the instance size and the parameter.

Different parameters can give different complexity!

Parameterised Complexity Theory

Standard tractability:
Solvable in $O(n^c)$ time for some constant c.

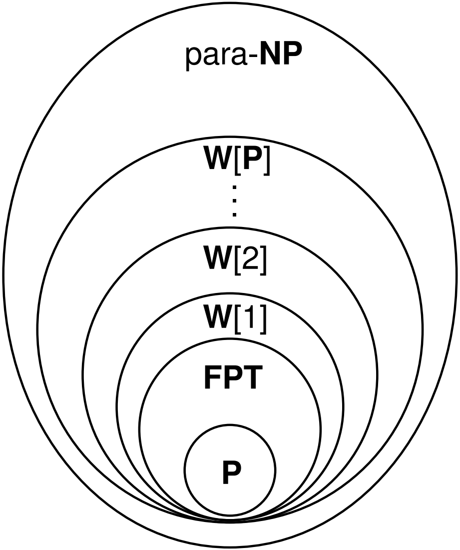
Fixed-parameter tractability (fpt):
solvable in $O(f(k) \cdot n^c)$ time, where

- f is some computable function and
- c is some constant.

This allows exponentiality in the parameter, but not in the instance size. (More relaxed and realistic than standard tractability.)

Parameterised Complexity Classes

There is a hardness concept similar to NP-completeness but using different classes



Several Parameters

Analogous when using more than one parameter.

For example, for two parameters k_1 and k_2 we define fpt as solvable in $O(f(k_1, k_2).n^c)$ time.

- Straighforward for parameters of the instance.
- Some subtle issues if using more than one parameter of the solution, e.g. both plan cost and plan length. (See paper.)

SAS^+ Planning

SAS^+planning framework – Bäckström and Nebel (1995)

- A SAS^+ *planning instance*: $\mathbb{P} = \langle V, A, I, G \rangle$
  - $V \rightarrow$ *variables*
  - $A \rightarrow$ *actions*
  - $I \rightarrow$ *initial state*
  - $G \rightarrow$ *goal state*
- Each variable $v \in V$ has a finite *domain* $D(v)$.
- Each action $a \in A$, $a : \mathrm{pre}(a) \rightarrow \mathrm{eff}(a)$

Domains

- $\mathbb{Z}_+ = \{1, 2, 3, \ldots\}$
- $\mathbb{Z}_0 = \{0, 1, 2, 3, \ldots\}$
- $\mathbb{Q}_+ = \{x \in \mathbb{Q} \mid x > 0\}$
- $\mathbb{Q}_0 = \{x \in \mathbb{Q} \mid x \geq 0\}$
- $\mathbb{Q}_1 = \{x \in \mathbb{Q} \mid x \geq 1\}$

Domain $\mathbb{Q}_1$ is included to distinguish if hardness results for $\mathbb{Q}_+$ are due to rational values or small values.

Cost-optimal Planning – COP($\mathbb{D}$,π)

Let $\mathbb{D}$ be a numeric domain and π a set of parameters. We define the following problem:

COP($\mathbb{D}$,π):

- INSTANCE: A tuple $\langle \mathbb{P}, c, k \rangle$
  - $\mathbb{P} = \langle V, A, I, G \rangle$ is a SAS^+planning instance
  - $c : A \rightarrow \mathbb{D}$ is a cost function
  - $k \in \mathbb{D}$
- PARAMETERS: A set of parameters π.
- QUESTION: Does $\mathbb{P}$ have a plan ω of cost $c(\omega) \leq k$?

The Parameters:

We considered the following parameters:

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

E.g. $\mathrm{COP}(\mathbb{Z}_0, \{k, \ell\})$

COP Major Results

	$\mathbb{Z}_+$				
para-**NP**-hard	-				
in **W**[P]	-				
in **W**[2]	*k*				

COP($\mathbb{Z}_+$) remains **W**[2]-complete for all parameter combinations that include k.

That is, COP($\mathbb{Z}_+$) is no harder than LOP.

COP Major Results

	$\mathbb{Z}_+$	$\mathbb{Z}_0$			
para-**NP**-hard	-	k			
in **W**[P]	-	-			
in **W**[2]	k	$k\ell, kz$			

COP($\mathbb{Z}_0$) is in **W**[2] for the parameter combinations $\{k, \ell\}$ and $\{k, z\}$ but it is para-**NP**-hard for $\{k\}$.

That is, COP($\mathbb{Z}_0$) is very hard unless we somehow restrict the plan length.

COP Major Results

	$\mathbb{Z}_+$	$\mathbb{Z}_0$	$\mathbb{Q}_1$		
para-**NP**-hard	-	k	-		
in **W**[P]	-	-	k		
in **W**[2]	k	$k\ell, kz$	kd		

COP($\mathbb{Q}_1$) is in **W**[2] for $\{k, d\}$ and in **W[P]** for $\{k\}$.

That is, COP for rational costs may be harder than LOP unless we restrict the denominators of the costs, *even if we have no costs smaller than one*!

(We say "may be" since we have not proven hardness for **W[P]**, only membership.)

Meysam Aghighi and Christer Bäckström

COP Major Results

	$\mathbb{Z}_+$	$\mathbb{Z}_0$	$\mathbb{Q}_1$	$\mathbb{Q}_+$	
para-**NP**-hard	-	k	-	k	
in **W**[P]	-	-	k	$k\ell, k\frac{1}{c_{\min}}$	
in **W**[2]	k	$k\ell, kz$	kd	kd	

COP($\mathbb{Q}_+$) is in **W**[2] for $\{k, d\}$, in **W[P]** for $\{k, \ell\}$ and $\{k, \frac{1}{c_{\min}}\}$, but para-**NP**-hard for $\{k\}$.

That is, COP for arbitrary positive rational costs is very hard unless we restrict the plan length or the minimum action cost. If restricting the denominators, it is no harder than LOP.

Even without zero-cost actions, it matters if we allow costs smaller than one or not!

COP Major Results

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

COP($\mathbb{Q}_0$) is in **W**[2] for $\{k, \ell, d\}$ and $\{k, z, d\}$, in **W**[**P**] for $\{k, \ell\}$, but remains para-**NP**-hard for $\{k, d, \frac{1}{c_{\min}}\}$.

If we allow also zero-cost actions, then it does not help to restrict the denominators or the minimum action cost, unless we also restrict the plan length.

COP Major Results

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

An example of how to interpret the results:

What does it mean that COP for positive rationals is easier if we also restrict the denominators?

Some Explicit Bounds – Upper Bounds

We can derive the following upper bounds: ($n = |\mathbb{P}|$)

- $COP(\mathbb{Z}_+, \{k\})$ can be solved in $O(n^k)$ time
- $COP(\mathbb{Q}_+, \{k, d\})$ can be solved in $O(n^{kd^d})$ time

Obviously, the second result is not very useful unless we have a maximum value for d.

Hence, the denominators matter!

Some Explicit Bounds – A Separation

The upper bounds are not sufficient to prove that the latter case is harder.

We can also prove a lower-bound result for $COP(\mathbb{Q}_+, \{k\})$ to get a separation as follows: ($n = |\mathbb{P}|$)

- $COP(\mathbb{Z}_+, \{k\})$ can be solved in time $O(n^k)$
- $COP(\mathbb{Q}_+, \{k\})$ cannot be solved in time $O(n^{ck})$ for any $c > 0$, unless $\mathbf{P} = \mathbf{NP}$

That is, COP is *strictly harder* for positive rationals than for positive integers!

Net-benefit Planning – NBP($\mathbb{D}$,π)

- INSTANCE: A tuple $\langle \mathbb{P}, c, u, b \rangle$
  - $\mathbb{P} = \langle V, A, I, G \rangle$ is a SAS^+planning instance
  - $c : A \to \mathbb{D}$ is a cost function
  - $u : \mathrm{vars}(G) \to \mathbb{D}$ is a utility function
  - $b \in \mathbb{D}$ is a benefit value
- PARAMETER: A set of parameters π.
- QUESTION: Is there a state s and a plan ω from I to s such that $u(s) - c(\omega) \geq b$?

Net-benefit Planning – NBP($\mathbb{D}$,π)

Previously known complexity results:

- NBP is PSPACE-complete for all cost-domains (van den Briel et. al., AAAI-2004)
- NBP is para-NP-hard for all cost domains, using *b* as parameter (Aghighi and Bäckström, IJCAI-2015)

We have made a multi-parameter analysis also for NBP.

Net-benefit Planning – Additional Parameters

In addition to the parameters for COP, we also use:

- b Min. net-benefit of plan
- u_{min} Min. variable utility
- u_{max} Max. variable utility
- $\#u$ Number of different utility values
- t Sum of all utilities, i.e. $t = \sum_{v \in \mathrm{vars}(G)} u(v)$

NBP – Summary of Results

- $u_{\min}$, $u_{\max}$ and $\#u$ didn't have a great impact on the parameterised complexity of NBP

NBP – Summary of Results

- $u_{\min}$, $u_{\max}$ and $\#u$ didn't have a great impact on the parameterised complexity of NBP
- On the other hand, the sum of all utilities, t, had:

$$\text{NBP}(\mathbb{D},\{b, t\}) \leq_{\text{fpt}} \text{COP}(\mathbb{D},\{k\})$$

NBP – Summary of Results

- $u_{\min}$, $u_{\max}$ and $\#u$ didn't have a great impact on the parameterised complexity of NBP
- On the other hand, the sum of all utilities, t, had:

$$\text{NBP}(\mathbb{D},\{b,t\}) \leq_{\text{fpt}} \text{COP}(\mathbb{D},\{k\})$$

- Membership results for NBP follow from the above parameterised reduction.

Discussion – Zero-cost actions

Practical experience shows that zero-cost actions are difficult for COP, and that it helps to somehow take also plan length into account.

Our results indicate that this is not caused by the actual algorithms used for planning, but is an inherent hardness in the problem. Restricting the plan length seems necessary to avoid the hardness.

Discussion – Rational-cost actions

It has been suggested in the literature that positive rational costs are difficult for heuristic search algorithms, and that a large span or ratio of costs is difficult.

Our results show that also this hardness is inherent in the problem. However, neither the span nor the ratio seems important, but the minimum cost is (and the maximum denominator even more).

Thank you!