Planning Complexity—A Parameterized Analysis

Yue Chen
Vienna University of Technology
Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider

February 2012
A little history to begin with

Standard Complexity Analysis

*The computational complexity of propositional STRIPS planning*

by Tom Bylander, 1994
A little history to begin with

Standard Complexity Analysis

*The computational complexity of propositional STRIPS planning*

by Tom Bylander, 1994

It is *PSPACE*-complete to determine if any given planning instance has any solutions.
Parameterized complexity analysis

Standard Complexity Analysis

- Bylander
- Bäckström and Nebel
- P, U, B, S
- ...
Parameterized complexity analysis

Standard Complexity Analysis
- Bylander
- Bäckström and Nebel
- $P, U, B, S$
- ...

Parameterized Complexity Analysis
- Stefan Szeider
- Sebastian Ordyniak
An example of planning

Tower of Hanoi
The language of planning

Definition
An instance of BOUNDED SAS$^+$ PLANNING problem $\mathcal{P}$ is a tuple $\mathcal{P} = (V, D, A, I, G)$ with components defined as follows:
The language of planning

Definition

An instance of BOUNDED SAS$^+$ PLANNING problem $\mathcal{P}$ is a tuple $\mathcal{P} = (V, D, A, I, G)$ with components defined as follows:

- $V = \{v_1, \ldots, v_n\}$ is a set of state variables. $D$ is a domain function for $V$. Each variable has an associated domain which implicitly defines an extended domain $D^+_v = D_v \cup \{u\}$, where $u$ denotes the undefined value. A state $s$ is an $n$-tuple of values. The states that contain $u$ are called partial states, otherwise total states. $s[v]$ denotes the value of the variable $v$ in a state $s$. 
The language of planning Cont’d

Definition

- $A$ is a set of actions. Each action $a \in A$ has a \textit{precondition} $\text{pre}(a)$ and an \textit{effect} $\text{eff}(a)$, both are partial states.
- $I$ as a total state is the \textit{initial state} and $G$ as a partial state is the \textit{goal state}.
Preliminary definitions

Let $a$ be an action and $s$ is a state,

- $a$ is valid in $s$ if $\text{pre}(a)$ is either $s$ or undefined w.r.t all variables;
- The result of $a$ in $s$ is the state $s$ updated by the effect of $a$;
- A state $s$ is a goal state iff $s$ equals $G$ unless $G$ is undefined w.r.t all variables.
Preliminary definitions

Let $a$ be an action and $s$ is a state,

- $a$ is **valid** in $s$ if $\text{pre}(a)$ is either $s$ or undefined w.r.t all variables;
- The **result** of $a$ in $s$ is the state $s$ updated by the effect of $a$;
- A state $s$ is a **goal state** iff $s$ equals $G$ unless $G$ is undefined w.r.t all variables.

Let $s_0, s_f$ be two total states, $\omega = \langle a_1, \ldots, a_n \rangle$ is a possibly empty sequence of actions. $\omega$ is a plan from $s_0$ to $s_f$ iff
Preliminary definitions

Let $a$ be an action and $s$ is a state,

- $a$ is valid in $s$ if $\text{pre}(a)$ is either $s$ or undefined w.r.t all variables;
- The result of $a$ in $s$ is the state $s$ updated by the effect of $a$;
- A state $s$ is a goal state iff $s$ equals $G$ unless $G$ is undefined w.r.t all variables.

Let $s_0, s_l$ be two total states, $\omega = \langle a_1, \ldots, a_n \rangle$ is a possibly empty sequence of actions. $\omega$ is a plan from $s_0$ to $s_l$ iff

- either $\omega$ is an empty sequence or
- there are $l - 1$ total states such that there is a valid action in each of them, and the result of that action updates the states in which it is valid to its successor in the sequence.
Preliminary definitions

Let $a$ be an action and $s$ is a state,

- $a$ is **valid** in $s$ if $\text{pre}(a)$ is either $s$ or undefined w.r.t all variables;
- The **result** of $a$ in $s$ is the state $s$ updated by the effect of $a$;
- A state $s$ is a **goal state** iff $s$ equals $G$ unless $G$ is undefined w.r.t all variables.

Let $s_0$, $s_I$ be two total states, $\omega = \langle a_1, \ldots, a_n \rangle$ is a possibly empty sequence of actions. $\omega$ is a **plan** from $s_0$ to $s_I$ iff

- either $\omega$ is an empty sequence or
- there are $I - 1$ total states such that there is a valid action in each of them, and the result of that action updates the states in which it is valid to its successor in the sequence.

$\omega$ is a **plan for $\mathbb{P}$** iff it is a plan from $I$ to some goal state.
The $\textit{SAS}^+$ BOUNDED PLANNING Problem

\textit{Instance:} A pair $\langle P, k \rangle$ where $P$ is an $\textit{SAS}^+$ instance and $k$ a positive integer.

\textit{Parameter:} $k$

\textit{Question:} Is there a plan for $P$ of length at most $k$?
Syntactic Restrictions

A BOUNDED SAS$^+$ PLANNING instance is:

(P) *post-unique* if no two distinct actions can change the same state variable to the same value;

(U) *unary* if each action changes exactly one state variable;

(B) *binary* if $|D| = 2$;

(S) *single-valued* if any two actions that both change the same state variable from defined to undefined value must require the same defined value.
Hardness Results for Bounded $\text{SAS}^+$ Planning with Syntactic Restrictions

- $\{B, S\}$-BOUNDED $\text{SAS}^+$ PLANNING is $\mathsf{W}[2]$-hard when the actions have no preconditions;
- $\{U, B, S\}$-BOUNDED $\text{SAS}^+$ PLANNING is $\mathsf{W}[1]$-hard when every actions has at most one precondition and one effect;
Membership Results for Bounded $\text{SAS}^+$ Planning with Syntactic Restrictions

- $\text{BOUNDDED SAS}^+$ PLANNING is in $W[2]$;
- $\{U\}$-$\text{BOUNDDED SAS}^+$ PLANNING is in $W[1]$;
- $\{P\}$-$\text{BOUNDDED SAS}^+$ PLANNING is in $\text{FPT}$;
Completeness Results for Bounded $SAS^+$ Planning

- $\{B, S\}$-BOUNDED $SAS^+$ PLANNING is $\mathbf{W}[2]$-complete when the actions have no preconditions;
- $\{U, B, S\}$-BOUNDED $SAS^+$ PLANNING is $\mathbf{W}[1]$-complete when every action has at most one precondition and one effect;
Introduction

Problem Description

Results

PUBS Lattice with Old and New Result

PSPACE-C

NP-C

NP-H

PUBS

PUS

PUB

PBS

UBS

US

UB

BS

P

U

S

B

PU

PS

PB

in P

PUS

PUB

PBS

UBS

Yue Chen Vienna University of Technology Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider

Planning Complexity—A Parameterized Analysis
Introduction
Problem Description
Results

PUBS Lattice with Old and New Result

in FPT

PUBS

PU
PS
PB
PUB
PUBS

US
PBS

UB

BS

W[2]-C

W[1]-C
PUBS Lattice with Old and New Result

- **PSPACE-C**: in FPT
- **PUS**: in P

Diagram showing the lattice with nodes labeled as follows:
- PUBS
- PUB
- PBS
- UBS
- PUBS
- PUB
- PBS
- UBS
- PUS
- PS
- PB
- US
- UB
- BS
- W[2]-C
- W[1]-C
- NP-C
- NP-H
PUBS Lattice with Old and New Result

Planning Complexity—A Parameterized Analysis
Planning Complexity—A Parameterized Analysis

Yue Chen
Vienna University of Technology
Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider
Planning Complexity—A Parameterized Analysis
Planning Complexity—A Parameterized Analysis

Yue Chen Vienna University of Technology Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider
PUBS Lattice with Old and New Result

Yue Chen Vienna University of Technology Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider

Planning Complexity—A Parameterized Analysis
Introduction

Problem Description

Results

Planning Complexity—A Parameterized Analysis

PUBS Lattice with Old and New Result

in FPT

in P

W[2]-C

W[1]-C

NP-C

NP-H
Thank You!
\[ s[v] = 1 \]
\[ s[v] = u \]
\[ s[v] = 0 \]

\[ a \]
\[ a' \]

\[ s \]
\[ s' \]
Introduction

Problem Description

Results

Yue Chen Vienna University of Technology Joint work with Christer Bäckström, Peter Jonsson, Sebastian Ordyniak and Stefan Szeider

Planning Complexity—A Parameterized Analysis