Search Strategies for Partial-Order Causal-Link Planning with Preferences

Pascal Bercher and Fabian Ginter and Susanne Biundo

Institute of Artificial Intelligence

September 16th, 2013

ulm university universität uulm

Search Strategies for POCL Planning with Preferences – **Outline**

(**POCL** Planning == **P**artial-**O**rder **C**ausal-**L**ink Planning)

- Planning with soft goals / at-end preferences (PDDL 3.0)
- Optimize Net-Benefit
- No compilation, but native implementation in PANDA, a POCL Planning Architecture (PhD thesis, Schattenberg)

Search Strategies for POCL Planning with Preferences – **Outline**

1. Problem Formalization
2. POCL Planning: Basics
3. Flaw Selection Strategies
    - Which Preference Flaw to Select?
    - How to Address a Preference Flaw?
4. Empirical Evaluation
5. Summary & Outlook

A planning domain is a tuple $\mathcal{D} = \langle \mathcal{V}, \mathcal{A} \rangle$ with:

- $\mathcal{V}$ is a finite set of state variables, $s \in 2^{\mathcal{V}}$ being a state,
- $\mathcal{A}$ is a finite set of actions,
  an action $a := \langle pre, add, del \rangle \in \mathcal{A}$ consists of:
  - $pre \subseteq \mathcal{V}$, the precondition of a,
  - $add \subseteq \mathcal{V}$, the add list of a,
  - $del \subseteq \mathcal{V}$, the delete list of a
  - $cost(a)$, the cost of a

A planning domain is a tuple $\mathcal{D} = \langle \mathcal{V}, \mathcal{A} \rangle$ with:

- $\mathcal{V}$ is a finite set of state variables, $s \in 2^{\mathcal{V}}$ being a state,
- $\mathcal{A}$ is a finite set of actions,
  an action $a := \langle \mathit{pre}, \mathit{add}, \mathit{del} \rangle \in \mathcal{A}$ is applicable:
  - in a state $s \in 2^{\mathcal{V}}$ iff $\mathit{pre} \subseteq s$,
  - and generates the state $(s \setminus \mathit{del}) \cup \mathit{add}$
  - (applicability of action sequences is defined as usual)

A POCL planning problem is a tuple $\pi = \langle \mathcal{D}, P_{init} \rangle$ with:

- $\mathcal{D}$ is the planning domain
- P_{init} is the initial plan which contains/encodes:
  - $s_{init} \in 2^{\mathcal{V}}$, the initial state
  - $G \subseteq \mathcal{V}$, the goal description

A solution to π is a plan P, s.t.:

- for every action sequence $\bar{a}$ induced by P holds:
  - $\bar{a}$ is applicable in s_{init},
  - $\bar{a}$ generates a state $s' \supseteq G$

A POCL planning problem is a tuple $\pi = \langle \mathcal{D}, P_{init} \rangle$ with:

- $\mathcal{D}$ is the planning domain
- P_{init} is the initial plan which contains/encodes:
  - $s_{init} \in 2^{\mathcal{V}}$, the initial state
  - $G \subseteq \mathcal{V}$, the goal description
  - $O \subseteq 2^{2^{\mathcal{V}}}$, the optional goals, called *preferences*

A solution to π is a plan P, s.t.:

- for every action sequence $\bar{a}$ induced by P holds:
  - $\bar{a}$ is applicable in s_{init},
  - $\bar{a}$ generates a state $s' \supseteq G$

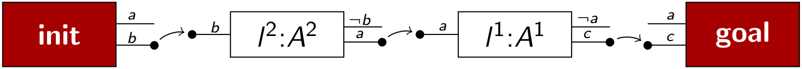
Planning goal: find a plan with best quality:
→ **Optimize the net benefit of a plan P!**

- Each $o \in O$, $o \subseteq 2^{\mathcal{V}}$ is a formula in disjunctive normal form.
- Each $o \in O$ has a benefit, $ben(o)$
- The *net benefit* is defined by:
  $NetBen(P) = \sum_{o \in O, P \models o} ben(o) - cost(P)$

A partial plan is a tuple $P = (PS, \prec, CL)$ with:

- PS is a finite set of plan steps,
  a plan step $l{:}a \in PS$ is a labeled action,
- $\prec$ is a strict partial order on PS,
- CL is a set of causal links between the plan steps in PS

Example for a partial plan:



$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$

Flaws	Modifications

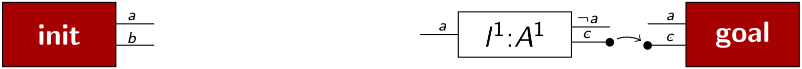
$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$

Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(c, goal)$	Causal Link: $(A^1, c, goal)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$

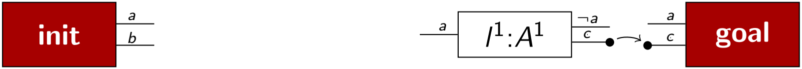
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(c, goal)$	Causal Link: $(A^1, c, goal)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



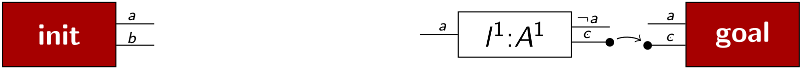
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(c, goal)$	Causal Link: $(A^1, c, goal)$

$$\mathcal{A} = \{A^1, A^2\} \text{ with: } A^1 = (a, \neg a \wedge c) \text{ and } A^2 = (b, \neg b \wedge a)$$
$$\text{initial state } s_{init} = a \wedge b, \quad \text{goal description } g = a \wedge c$$



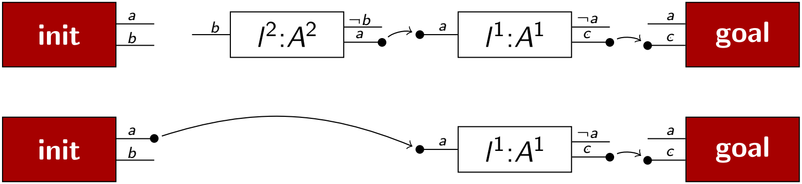
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(a, l^1{:}A^1)$	Causal Link: $(A^2, a, l^1{:}A^1)$ Causal Link: $(init, a, l^1{:}A^1)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



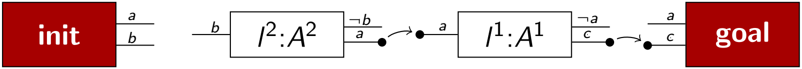
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(a, I^1{:}A^1)$	Causal Link: $(A^2, a, I^1{:}A^1)$ Causal Link: $(init, a, I^1{:}A^1)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



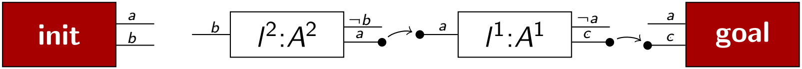
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$
Open Precondition: $(a, l^1{:}A^1)$	Causal Link: $(A^2, a, l^1{:}A^1)$ Causal Link: $(init, a, l^1{:}A^1)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



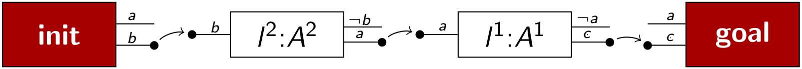
Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$ Causal Link: $(l^2{:}A^2, a, goal)$
Open Precondition: $(b, l^2{:}A^2)$	Causal Link: $(init, b, l^2{:}A^2)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$ Causal Link: $(l^2{:}A^2, a, goal)$
Open Precondition: $(b, l^2{:}A^2)$	Causal Link: $(init, b, l^2{:}A^2)$

$\mathcal{A} = \{A^1, A^2\}$ with: $A^1 = (a, \neg a \wedge c)$ and $A^2 = (b, \neg b \wedge a)$
initial state $s_{init} = a \wedge b$, goal description $g = a \wedge c$



Flaws	Modifications
Open Precondition: $(a, goal)$	Causal Link: $(A^2, a, goal)$ Causal Link: $(init, a, goal)$ Causal Link: $(l^2{:}A^2, a, goal)$
Open Precondition: $(b, l^2{:}A^2)$	Causal Link: $(init, b, l^2{:}A^2)$

Algorithm 1: POCL planning procedure

```
$Fringe \leftarrow \{P_{init}\}$
while $Fringe \neq \emptyset$ do
    $P \leftarrow$ planSel $(Fringe)$
    $Fringe \leftarrow Fringe \setminus \{P\}$

    if $Flaws(P) = \emptyset$ then return $P$

    $f \leftarrow$ flawSel $(Flaws(P))$
    $Fringe \leftarrow Fringe \cup \{$ applyMod$(P, m) \mid m \in Mods(f) \}$
return fail
```

→ two choices:
plan selection (`planSel`) and **flaw selection** (`flawSel`)

plan selection (`planSel`):
A* or weighted A* with heuristics for POCL plans:

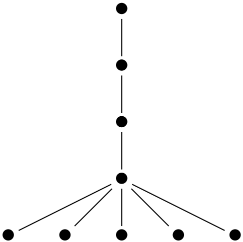
- with preferences: Bercher & Biundo, PuK 2011, FLAIRS 2012
- without preferences: $\underbrace{\text{Bercher et al., KI 2013}}_{\text{Thursday, Planning \& Application at 15:30}}$, ICTAI 2013

flaw selection (`flawSel`):

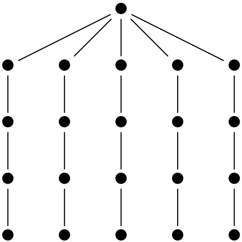
- with preferences: Bercher et al., PuK 2013
- without preferences: Bercher et al., ???

Flaw selection can never be wrong, but it “forms the search space”.

Let the plan P have the flaws: $f_1, \ldots, f_4$.



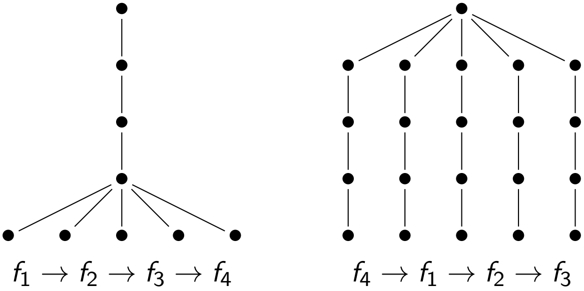
$f_1 \rightarrow f_2 \rightarrow f_3 \rightarrow f_4$



$f_4 \rightarrow f_1 \rightarrow f_2 \rightarrow f_3$

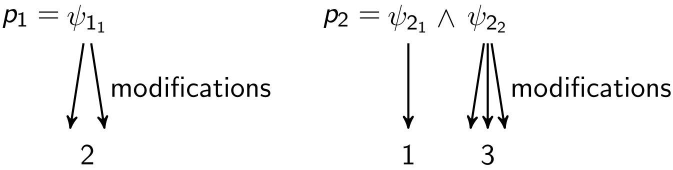
Flaw selection can never be wrong, but it “forms the search space”.

Let the plan P have the flaws: $f_1, \ldots, f_4$.



$LCFR$ → Select a flaw with least number of modifications.

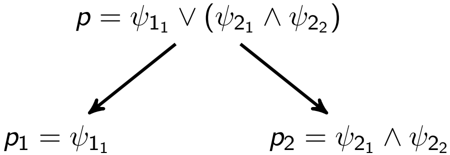
In which order to select the preference flaws?
How to combine estimates of a preference?



$$estimate(p_1) = 2$$
$$estimate(p_2) = 1?\ (min)\ /\ 4?\ (sum)\ /\ 3?\ (product)$$

Splitting: *Focus on disjuncts individually.*

First selection of the preference p (example):

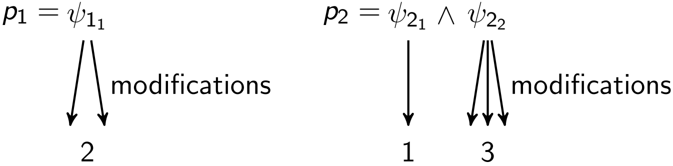


$p = \psi_{1_1} \vee (\psi_{2_1} \wedge \psi_{2_2})$

$p_1 = \psi_{1_1}$

$p_2 = \psi_{2_1} \wedge \psi_{2_2}$

Each subsequent selection of the preference:



$p_1 = \psi_{1_1}$

modifications

2

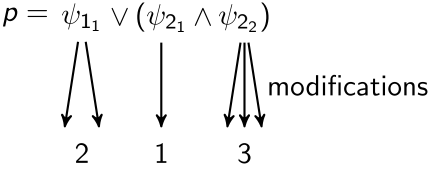
$p_2 = \psi_{2_1} \wedge \psi_{2_2}$

modifications

1

3

No Splitting: *Stay flexible and work on different disjuncts.*



$$p = \psi_{1_1} \vee (\psi_{2_1} \wedge \psi_{2_2})$$

Example: After a modification for ψ_{2_1} was applied, search is not committed to $(\psi_{2_1} \wedge \psi_{2_2})$, but may also support ψ_{1_1}

Evaluation compares:

- Solve ordinary goals first vs. solve soft goals first
- *splitting* vs. *no splitting*
- Estimate combination: *min* vs. *sum* vs. *product*
- "A* NetBenefit", which maximizes $NetBen(P) - h(P)$, h being the additive heuristic for POCL planning.

Evaluation compares:

- Solve ordinary goals first vs. solve soft goals first
- *splitting* vs. *no splitting*
- Estimate combination: *min* vs. *sum* vs. *product*
- "A* NetBenefit", which maximizes $NetBen(P) - h(P)$, h being the additive heuristic for POCL planning.

Evaluated planning domains/problems:

- 5 hand-modeled problem instances of one domain
- 120 randomly generated planning problems

- Solve ordinary goals first vs. solve soft goals first
  - Qualities much better if hard flaws are solved first.
- *splitting* vs. *no splitting*
- Estimate combination: *min* vs. *sum* vs. *product*
  - no clear results

- Solve ordinary goals first vs. solve soft goals first
  - 2/120 problems were solved if preferring soft goals first
- *splitting* vs. *no splitting*
- Estimate combination: *min* vs. *sum* vs. *product*
  - *splitting* with *minimum* produces the worst solutions
  - *no splitting* produces the best solutions
  - *sum* and *product* equally good, *min* performs badly

Summary:

- Introduced how simple preferences can be addressed in a POCL system: splitting vs. no splitting
- Evaluated various strategies:
  - Soft goals first / last
  - How to aggregate flaw cost estimates?

Outlook:

- Evaluate IPC domains
- Integrate heuristic from FLAIRS 2012