Learning STRIPS Action Models with Classical Planning

D. Aineto S. Jiménez E. Onaindia

June 27, 2018

Universitat Politècnica de València

Contents

Introduction

Learning STRIPS Action Models with Classical Planning

Evaluation

Discussion and Further Work

Introduction

Why learn STRIPS action models?

Extend applicability of AI planning

Many STRIPS compilable models: grammars, automata

Learning Action Models

- Header: name + parameter list
- **Precondition list**
- **Effects list**

Action model

from plan traces

$$t = \langle s_o, a_1, s_1, \ldots, a_n, g \rangle, \forall t \in \mathcal{T}$$

Approaches to learning action models

ARMS, SLAF, and most approaches

$$t = \langle s_o, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

Approaches to learning action models

ARMS, SLAF, and most approaches

$$t = \langle s_o, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

LOCM family

$$t = \langle s_0, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

Approaches to learning action models

ARMS, SLAF, and most approaches

$$t = \langle s_o, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

LOCM family

$$t = \langle s_0, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

Our approach

$$t = \langle s_0, a_1, s_1, a_2, s_2, \ldots, a_n, g \rangle$$

Learning Task

This learning task is defined as $\Lambda = \langle \mathcal{M}, \Psi, \mathcal{T} \rangle$:

- $\mathcal{M}$ is the set of *initial* action models (at least headers)
- Ψ is the set of predicates
- $\mathcal{T}$ is a set of plan traces $t = \langle s_o, a_1, s_1, \ldots, a_n, g \rangle, \forall t \in \mathcal{T}$

A solution to Λ is a set of action models $\mathcal{M}'$
compliant with $\mathcal{M}$, Ψ and $\mathcal{T}$

Learning STRIPS Action Models with Classical Planning

Learning as Planning

Compile Λ into a planning problem $P_\Lambda = \langle F_\Lambda, A_\Lambda, I_\Lambda, G_\Lambda \rangle$

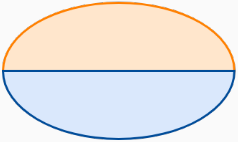
A solution to P_Λ:

1. Edits the action models $\mathcal{M}$ to obtain $\mathcal{M}'$.
2. Validates the learnt models $\mathcal{M}'$ in $\mathcal{T}$.

Learning as Planning

```
(ontable A) (on B A)
(clear B) (holding C)
```

Original domain fluents



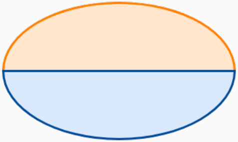
Model representation fluents

```
(pre_holding_put-down_var1)
(add_clear_put-down_var1)
(del_handempty_put-down)
```

Learning as Planning

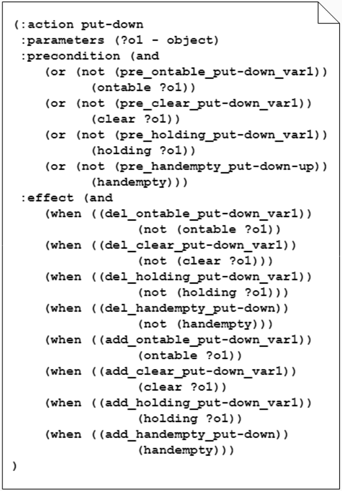
(ontable A) (on B A)
(clear B) (holding C)

Original domain fluents



Model representation fluents

(pre_holding_put-down_var1)
(add_clear_put-down_var1)
(del_handempty_put-down)

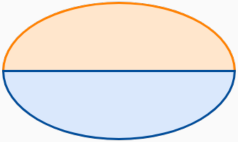


```
(:action put-down
 :parameters (?o1 - object)
 :precondition (and
    (or (not (pre_ontable_put-down_var1))
          (ontable ?o1))
    (or (not (pre_clear_put-down_var1))
          (clear ?o1))
    (or (not (pre_holding_put-down_var1))
          (holding ?o1))
    (or (not (pre_handempty_put-down-up))
          (handempty)))
 :effect (and
    (when ((del_ontable_put-down_var1))
                (not (ontable ?o1))
    (when ((del_clear_put-down_var1))
                (not (clear ?o1)))
    (when ((del_holding_put-down_var1))
                (not (holding ?o1)))
    (when ((del_handempty_put-down))
                (not (handempty)))
    (when ((add_ontable_put-down_var1))
                (ontable ?o1))
    (when ((add_clear_put-down_var1))
                (clear ?o1))
    (when ((add_holding_put-down_var1))
                (holding ?o1))
    (when ((add_handempty_put-down))
                (handempty)))
)
```

Learning as Planning

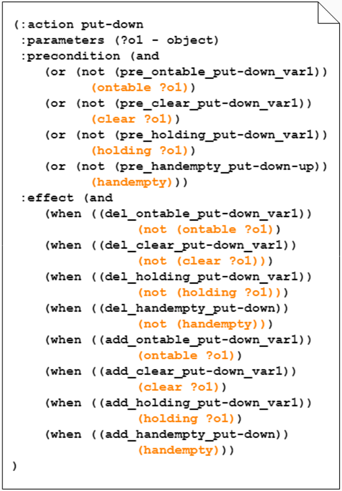
(ontable A) (on B A)
(clear B) (holding C)

Original domain fluents



Model representation fluents

(pre_holding_put-down_var1)
(add_clear_put-down_var1)
(del_handempty_put-down)

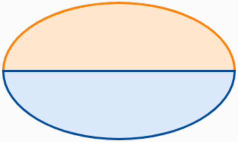


```
(:action put-down
 :parameters (?o1 - object)
 :precondition (and
    (or (not (pre_ontable_put-down_var1))
          (ontable ?o1))
    (or (not (pre_clear_put-down_var1))
          (clear ?o1))
    (or (not (pre_holding_put-down_var1))
          (holding ?o1))
    (or (not (pre_handempty_put-down-up))
          (handempty)))
 :effect (and
    (when ((del_ontable_put-down_var1))
                (not (ontable ?o1))
    (when ((del_clear_put-down_var1))
                (not (clear ?o1)))
    (when ((del_holding_put-down_var1))
                (not (holding ?o1)))
    (when ((del_handempty_put-down))
                (not (handempty)))
    (when ((add_ontable_put-down_var1))
                (ontable ?o1))
    (when ((add_clear_put-down_var1))
                (clear ?o1))
    (when ((add_holding_put-down_var1))
                (holding ?o1))
    (when ((add_handempty_put-down))
                (handempty)))
)
```

Learning as Planning

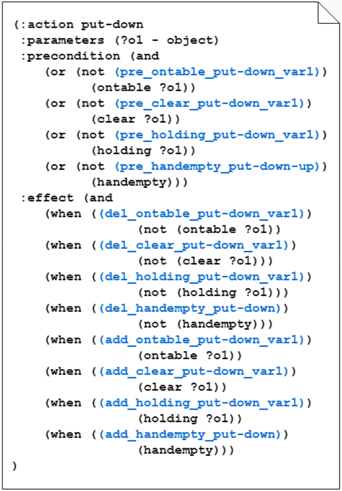
```
(ontable A) (on B A)
(clear B) (holding C)
```

Original domain fluents



Model representation fluents

```
(pre_holding_put-down_var1)
(add_clear_put-down_var1)
(del_handempty_put-down)
```

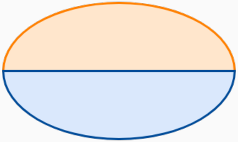


```
(:action put-down
 :parameters (?o1 - object)
 :precondition (and
    (or (not (pre_ontable_put-down_var1))
          (ontable ?o1))
    (or (not (pre_clear_put-down_var1))
          (clear ?o1))
    (or (not (pre_holding_put-down_var1))
          (holding ?o1))
    (or (not (pre_handempty_put-down-up))
          (handempty)))
 :effect (and
    (when ((del_ontable_put-down_var1))
                (not (ontable ?o1))
    (when ((del_clear_put-down_var1))
                (not (clear ?o1)))
    (when ((del_holding_put-down_var1))
                (not (holding ?o1)))
    (when ((del_handempty_put-down))
                (not (handempty)))
    (when ((add_ontable_put-down_var1))
                (ontable ?o1))
    (when ((add_clear_put-down_var1))
                (clear ?o1))
    (when ((add_holding_put-down_var1))
                (holding ?o1))
    (when ((add_handempty_put-down))
                (handempty)))
)
```

Learning as Planning

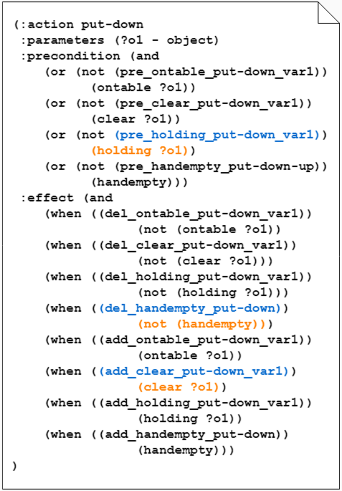
(ontable A) (on B A)
(clear B) (holding C)

Original domain fluents



Model representation fluents

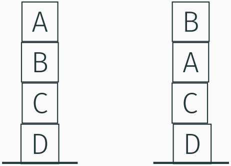
(pre_holding_put-down_var1)
(add_clear_put-down_var1)
(del_handempty_put-down)



```
(:action put-down
 :parameters (?o1 - object)
 :precondition (and
    (or (not (pre_ontable_put-down_var1))
          (ontable ?o1))
    (or (not (pre_clear_put-down_var1))
          (clear ?o1))
    (or (not (pre_holding_put-down_var1))
          (holding ?o1))
    (or (not (pre_handempty_put-down-up))
          (handempty)))
 :effect (and
    (when ((del_ontable_put-down_var1))
                (not (ontable ?o1))
    (when ((del_clear_put-down_var1))
                (not (clear ?o1)))
    (when ((del_holding_put-down_var1))
                (not (holding ?o1)))
    (when ((del_handempty_put-down))
                (not (handempty)))
    (when ((add_ontable_put-down_var1))
                (ontable ?o1))
    (when ((add_clear_put-down_var1))
                (clear ?o1))
    (when ((add_holding_put-down_var1))
                (holding ?o1))
    (when ((add_handempty_put-down))
                (handempty)))
)
```

Learning as Planning

$$\mathcal{T} = \{\langle s_0, \mathit{putdown}\ B, s_1\rangle\}$$



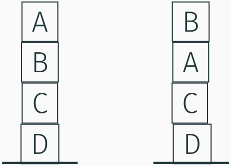
Solution plan

01: program_pre_holding_putdown_var1
02: program_del_holding_putdown_var1
03: program_add_clear_putdown_var1
04: program_add_ontable_putdown_var1
05: program_add_handempty_putdown

06: validate_0
07: unstack A B
08: putdown A
09: unstack B C
10: putdown B
11: pickup A
12: stack A C
13: pickup B
14: stack B A
15: validate_1

Learning as Planning

$$\mathcal{T} = \{\langle s_0, putdown\ B, s_1\rangle\}$$



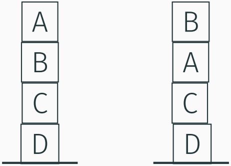
Solution plan

01: program_pre_holding_putdown_var1
02: program_del_holding_putdown_var1
03: program_add_clear_putdown_var1
04: program_add_ontable_putdown_var1
05: program_add_handempty_putdown

06: validate_0
07: unstack A B
08: putdown A
09: unstack B C
10: putdown B
11: pickup A
12: stack A C
13: pickup B
14: stack B A
15: validate_1

Learning as Planning

$$\mathcal{T} = \{\langle s_0, putdown\ B, s_1\rangle\}$$



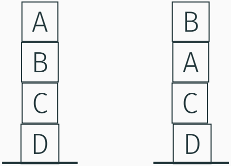
Solution plan

01: program_pre_holding_putdown_var1
02: program_del_holding_putdown_var1
03: program_add_clear_putdown_var1
04: program_add_ontable_putdown_var1
05: program_add_handempty_putdown

06: validate_0
07: unstack A B
08: putdown A
09: unstack B C
10: putdown B
11: pickup A
12: stack A C
13: pickup B
14: stack B A
15: validate_1

Learning as Planning

$$\mathcal{T} = \{\langle \mathbf{s_0}, putdown\ B, s_1\rangle\}$$



Solution plan

01: program_pre_holding_putdown_var1

02: program_del_holding_putdown_var1

03: program_add_clear_putdown_var1

04: program_add_ontable_putdown_var1

05: program_add_handempty_putdown

06: validate_0

07: unstack A B

08: putdown A

09: unstack B C

10: putdown B

11: pickup A

12: stack A C

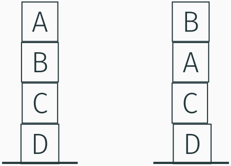
13: pickup B

14: stack B A

15: validate_1

Learning as Planning

$$\mathcal{T} = \{\langle s_0, \texttt{putdown B}, s_1\rangle\}$$



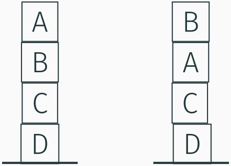
Solution plan

01: program_pre_holding_putdown_var1
02: program_del_holding_putdown_var1
03: program_add_clear_putdown_var1
04: program_add_ontable_putdown_var1
05: program_add_handempty_putdown

06: validate_0
07: unstack A B
08: putdown A
09: unstack B C
10: putdown B
11: pickup A
12: stack A C
13: pickup B
14: stack B A
15: validate_1

Learning as Planning

$$\mathcal{T} = \{\langle s_0, putdown\ B, \mathbf{s_1} \rangle\}$$



Solution plan

```
01: program_pre_holding_putdown_var1
02: program_del_holding_putdown_var1
03: program_add_clear_putdown_var1
04: program_add_ontable_putdown_var1
05: program_add_handempty_putdown

06: validate_0
07: unstack A B
08: putdown A
09: unstack B C
10: putdown B
11: pickup A
12: stack A C
13: pickup B
14: stack B A
15: validate_1
```

Additional Knowledge

Partially Specified Action Models

What is known about an action model can be hardcoded into the problem

Static Predicates

Automated preprocess to identify static predicates from the observations in the traces

Evaluation

Experiments

Tasks:

1. Labeled plans. $t = \langle s_0, a_1, a_2, \ldots, a_n, g \rangle, \forall t \in \mathcal{T}$
2. Labeled plans + static predicates
3. Labeled plans + static predicates + partial action models

Input: 5 plan traces of length 5-7

Evaluated on **12 IPC domains** using the Madagascar planner

- Blocks
- Driverlog
- Ferry
- Floortile
- Grid
- Gripper
- Hanoi
- Miconic
- Satellite
- Transport
- Visitall
- Zenotravel

Evaluation Metrics

wrt the Reference Model

(syntax-based evaluation)

Precision

(correctness)

$$P = \frac{tp}{tp + fp}$$

Recall

(completeness)

$$R = \frac{tp}{tp + fn}$$

Results

Task 1: Labeled plans. $t = \langle s_0, a_1, a_2, \dots, a_n, g \rangle, \forall t \in \mathcal{T}$

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.88	0.50	0.88	0.92	0.95	0.91	0.90	0.78	0.17

Results

Task 1: Labeled plans. $t = \langle s_0, a_1, a_2, \ldots, a_n, g\rangle, \forall t \in \mathcal{T}$

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.88	0.50	0.88	0.92	0.95	0.91	0.90	0.78	0.17

Task 2: Labeled plans + static predicates

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.90	0.74	0.93	0.92	0.96	0.91	0.93	0.86	0.13

Results

Task 1: Labeled plans. $t = \langle s_0, a_1, a_2, \ldots, a_n, g \rangle, \forall t \in \mathcal{T}$

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.88	0.50	0.88	0.92	0.95	0.91	0.90	0.78	0.17

Task 2: Labeled plans + static predicates

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.90	0.74	0.93	0.92	0.96	0.91	0.93	0.86	0.13

Task 3: Labeled plans + static predicates + partial action models

Pre		Add		Del		Global		
P	R	P	R	P	R	P	R	Time
0.98	0.71	1.00	0.98	1.00	0.95	0.99	0.87	0.11

Discussion and Further Work

Learning with unbound observations

Partial observability in actions and intermediate states

Extreme case: $t = \langle s0, g \rangle, \forall t \in \mathcal{T}$

Learning with unbound observations

Partial observability in actions and intermediate states

Extreme case: $t = \langle s0, g \rangle, \forall t \in \mathcal{T}$

Higher complexity task

- Learn the action models
- Fill the gaps in the traces
- Underconstrained search space

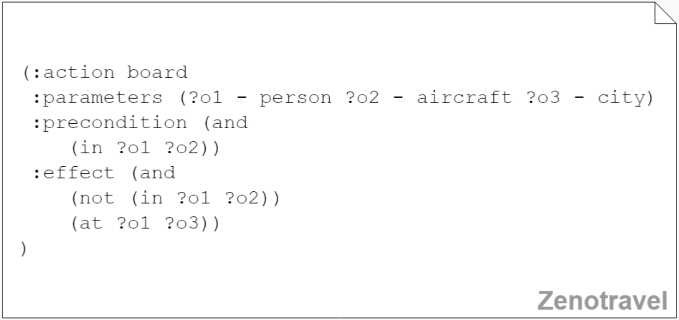
Task 4. Initial/Final States

	Pre		Add		Del		Global	
	P	R	P	R	P	R	P	R
Blocks	0.13	0.11	0.14	0.11	0.14	0.11	0.14	0.11
Driverlog	0.75	0.21	0.2	0.29	0.33	0.14	0.43	0.21
Ferry	0	0	0	0	0	0	0	0
Floortile	0.43	0.27	0.43	0.27	0.33	0.18	0.4	0.24
Grid	-	-	-	-	-	-	-	-
Gripper	0	0	0	0	0	0	0	0
Hanoi	0.5	0.25	0	0	0.5	0.5	0.33	0.25
Miconic	0	0	0	0	0	0	0	0
Satellite	0.5	0.21	0.57	0.8	0.75	0.75	0.61	0.59
Transport	0	0	0.29	0.4	0	0	0.1	0.13
Visitall	-	-	-	-	-	-	-	-
Zenotravel	0.5	0.14	0.29	0.29	0.5	0.29	0.43	0.24
	0.28	0.12	0.19	0.22	0.26	0.2	0.24	0.18

Reformulation

syntactically incorrect *but* *semantically correct*

Actions swapping roles



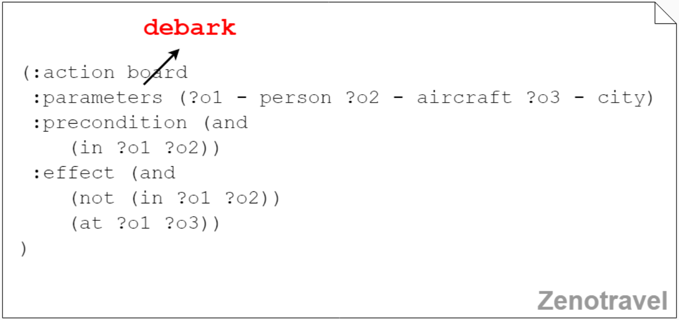
```
(:action board
 :parameters (?o1 - person ?o2 - aircraft ?o3 - city)
 :precondition (and
    (in ?o1 ?o2))
 :effect (and
    (not (in ?o1 ?o2))
    (at ?o1 ?o3))
)
```

Zenotravel

Reformulation

syntactically incorrect *but* *semantically correct*

Actions swapping roles



debark

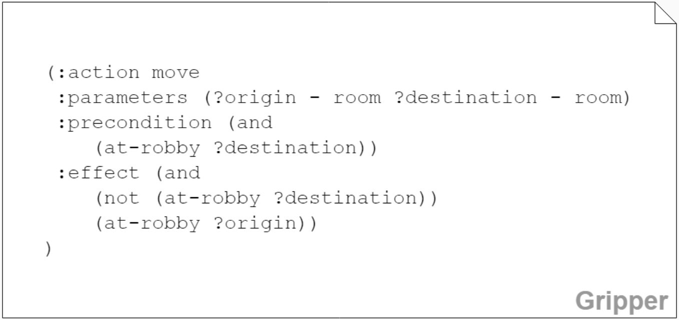
```
(:action board
 :parameters (?o1 - person ?o2 - aircraft ?o3 - city)
 :precondition (and
    (in ?o1 ?o2))
 :effect (and
    (not (in ?o1 ?o2))
    (at ?o1 ?o3))
)
```

Zenotravel

Reformulation

syntactically incorrect   *but*   *semantically correct*

Arguments swapping roles



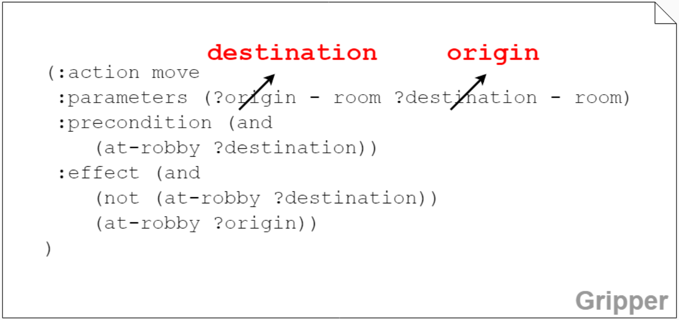
```
(:action move
 :parameters (?origin - room ?destination - room)
 :precondition (and
    (at-robby ?destination))
 :effect (and
    (not (at-robby ?destination))
    (at-robby ?origin))
)
```

Gripper

Reformulation

syntactically incorrect *but* *semantically correct*

Arguments swapping roles



```
(:action move
 :parameters (?origin - room ?destination - room)
 :precondition (and
    (at-robby ?destination))
 :effect (and
    (not (at-robby ?destination))
    (at-robby ?origin))
)
```

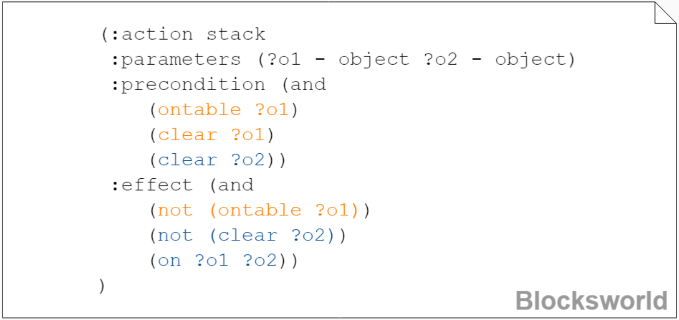
destination
origin
Gripper

Reformulation

syntactically incorrect *but* *semantically correct*

Macro actions

pick-up + stack



```
(:action stack
 :parameters (?o1 - object ?o2 - object)
 :precondition (and
    (ontable ?o1)
    (clear ?o1)
    (clear ?o2))
 :effect (and
    (not (ontable ?o1))
    (not (clear ?o2))
    (on ?o1 ?o2))
)
```

Blocksworld

Task 4. Initial/Final States (swapping roles)

	Pre		Add		Del		Global	
	P	R	P	R	P	R	P	R
Blocks	0.75	0.67	0.86	0.67	0.86	0.67	0.82	0.67
Driverlog	1	0.29	0.5	0.71	0.67	0.29	0.72	0.43
Ferry	1	0.57	1	1	1	1	1	0.86
Floortile	0.57	0.36	1	0.64	0.67	0.36	0.75	0.45
Grid	-	-	-	-	-	-	-	-
Gripper	1	0.67	1	1	1	1	1	0.89
Hanoi	1	0.5	1	1	1	1	1	0.83
Miconic	0.5	0.11	0.67	0.5	0.5	0.33	0.56	0.31
Satellite	0.5	0.21	0.57	0.8	0.75	0.75	0.61	0.59
Transport	1	0.3	0.71	1	1	0.6	0.9	0.63
Visitall	-	-	-	-	-	-	-	-
Zenotravel	1	0.29	0.57	0.57	1	0.57	0.86	0.48
	0.83	0.4	0.79	0.79	0.85	0.66	0.82	0.61

Future work

Metrics robust to reformulation

- Semantics-based evaluation
- Evaluating wrt a testing set

Questions?

Programmable Actions

```
(:action pick-up
 :parameters (?o1 - object ?i1 - step ?i2 - step)
 :precondition
   …
 :effect
   …
   (when (and (del_ontable_pick-up_var1 ))(not (ontable ?o1)))
   (when (and (del_clear_pick-up_var1 ))(not (clear ?o1)))
   (when (and (del_holding_pick-up_var1 ))(not (holding ?o1)))
   (when (and (del_handempty_pick-up ))(not (handempty )))
   (when (and (add_ontable_pick-up_var1 ))(ontable ?o1))
   (when (and (add_clear_pick-up_var1 ))(clear ?o1))
   (when (and (add_holding_pick-up_var1 ))(holding ?o1))
   (when (and (add_handempty_pick-up ))(handempty )))
)
```

Programmable Actions

```
(:action pick-up
 :parameters (?o1 - object ?i1 - step ?i2 - step)
 :precondition
   …
   (or (not (pre_ontable_pick-up_var1 ))(ontable ?o1))
   (or (not (pre_clear_pick-up_var1 ))(clear ?o1))
   (or (not (pre_holding_pick-up_var1 ))(holding ?o1))
   (or (not (pre_handempty_pick-up ))(handempty )))
 :effect
   …
)
```

Programming Actions

```
(:action program_eff_ontable_pick-up_var1
 :parameters ()
 :precondition (and
   (modeProg )
   (not (del_ontable_pick-up_var1))
   (not (add_ontable_pick-up_var1)))
 :effect (and
   (when (pre_ontable_pick-up_var1)
          (del_ontable_pick-up_var1))
   (when (not (pre_ontable_pick-up_var1))
          (add_ontable_pick-up_var1))))
```

Programming Actions

```
(:action program_pre_ontable_pick-up_var1
 :parameters ()
 :precondition
   (and (modeProg )
   (pre_ontable_pick-up_var1 )
   (not (del_ontable_pick-up_var1 ))
   (not (add_ontable_pick-up_var1 )))
 :effect
   (and (not (pre_ontable_pick-up_var1 )))
)
```

Validate actions in the input traces

```
(:action pick-up
 :parameters (?o1 - object ?i1 - step ?i2 - step)
 :precondition
   (not (modeProg ))
   (plan-pick-up ?i1 ?o1)
   (current ?i1)
   (inext ?i1 ?i2)
   …
 :effect (and
   (not (current ?i1))(current ?i2)
   …
)
```

Validate states in the input traces

```
(:action validate_1
 :parameters ()
 :precondition (and
   (not (modeProg ))
   (test0 )(not (test1 ))(not (test2 )) …
   (current i8)
   (holding d) (ontable a) (on b a) …
 :effect (and
   (test1 )
   (not (current i8))(current i1)
   (not (plan-unstack i1 A B))(not (plan-put-down i2 A)) …
   (plan-put-down i1 D)(plan-unstack i2 C B) …
))
```

Task 1. Labeled plans

	Pre		Add		Del			
	P	R	P	R	P	R	P	R
Blocks	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Driverlog	1.0	0.36	0.75	0.86	1.0	0.71	0.92	0.64
Ferry	1.0	0.57	1.0	1.0	1.0	1.0	1.0	0.86
Floortile	0.52	0.68	0.64	0.82	0.83	0.91	0.66	0.80
Grid	0.62	0.47	0.75	0.86	0.78	1.0	0.71	0.78
Gripper	1.0	0.67	1.0	1.0	1.0	1.0	1.0	0.89
Hanoi	1.0	0.50	1.0	1.0	1.0	1.0	1.0	0.83
Miconic	0.75	0.33	0.50	0.50	0.75	1.0	0.67	0.61
Satellite	0.60	0.21	1.0	1.0	1.0	0.75	0.87	0.65
Transport	1.0	0.40	1.0	1.0	1.0	0.80	1.0	0.73
Visitall	1.0	0.50	1.0	1.0	1.0	1.0	1.0	0.83
Zenotravel	1.0	0.36	1.0	1.0	1.0	0.71	1.0	0.69
	0.88	0.50	0.88	0.92	0.95	0.91	0.90	0.78

Task 2. Labeled plans + static predicates

	Pre		Add		Del			
	P	R	P	R	P	R	P	R
Blocks	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Driverlog	0.9	0.64	0.56	0.71	0.86	0.86	0.78	0.73
Ferry	1.0	0.57	1.0	1.0	1.0	1.0	1.0	0.86
Floortile	0.68	0.68	0.89	0.73	1.0	0.82	0.86	0.74
Grid	0.79	0.65	1.0	0.86	0.88	1.0	0.89	0.83
Gripper	1.0	0.67	1.0	1.0	1.0	1.0	1.0	0.89
Hanoi	0.75	0.75	1.0	1.0	1.0	1.0	0.92	0.92
Miconic	0.89	0.89	1.0	0.75	0.75	1.0	0.88	0.88
Satellite	0.82	0.64	1.0	1.0	1.0	0.75	0.94	0.80
Transport	1.0	0.70	0.83	1.0	1.0	0.80	0.94	0.83
Visitall	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Zenotravel	1.0	0.64	0.88	1.0	1.0	0.71	0.96	0.79
	0.90	0.74	0.93	0.92	0.96	0.91	0.93	0.86

Task 3. Labeled plans + static predicates + partial action models

	Pre		Add		Del			
	P	R	P	R	P	R	P	R
Blocks	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Driverlog	1.0	0.71	1.0	1.0	1.0	1.0	1.0	0.90
Ferry	1.0	0.67	1.0	1.0	1.0	1.0	1.0	0.89
Floortile	0.75	0.60	1.0	0.80	1.0	0.80	0.92	0.73
Grid	1.0	0.67	1.0	1.0	1.0	1.0	1.0	0.78
Gripper	1.0	0.50	1.0	1.0	1.0	1.0	1.0	0.83
Miconic	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0
Satellite	1.0	0.57	1.0	1.0	1.0	1.0	1.0	0.86
Transport	1.0	0.75	1.0	1.0	1.0	1.0	1.0	0.92
Zenotravel	1.0	0.67	1.0	1.0	1.0	0.67	1.0	0.78
	0.98	0.71	1.0	0.98	1.0	0.95	0.99	0.87

Relation to Plan/Goal Recognition

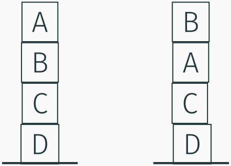
Plan/Goal recognition can be seen as $\Lambda = \langle \mathcal{M}', \Psi, \mathcal{T} \rangle$

Hence

~~Programming~~ & Validation

Relation to Plan/Goal Recognition

$\mathcal{T} = \langle\langle s_0, putdown\ B, s_1\rangle\rangle$



Solution plan

```
~~01: program_pre_holding_putdown_var1~~
~~02: program_del_holding_putdown_var1~~
~~03: program_add_clear_putdown_var1~~
~~04: program_add_ontable_putdown_var1~~
~~05: program_add_handempty_putdown~~

06: unstack A B
07: putdown A
08: unstack B C
09: putdown B
10: pickup A
11: stack A C
12: pickup B
13: stack B A
14: validate_1
```