

COMP90054 Workshop 4

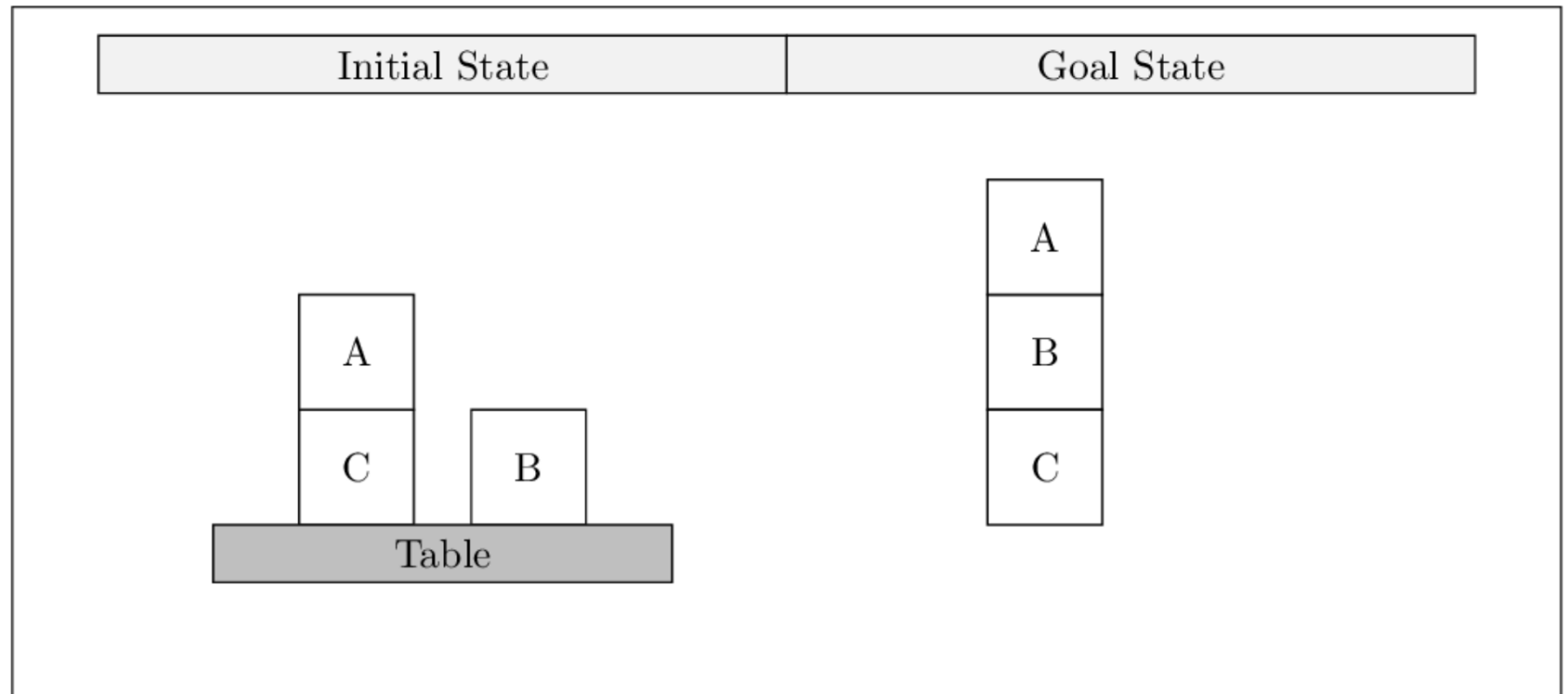
Geye Guo

Problem 1

Model Blocks-World as a STRIPS problem $P=\langle F,O,I,G\rangle$.

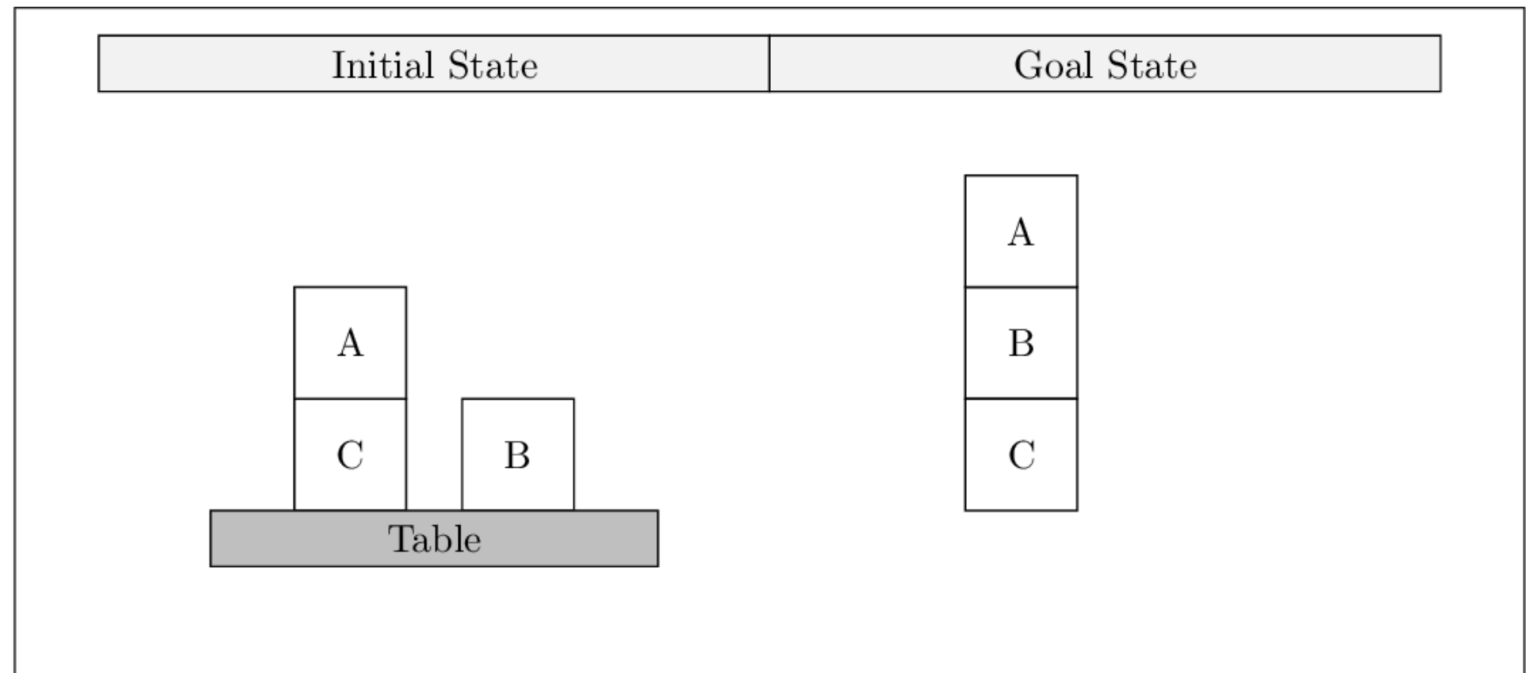
You need to define the set of facts F , the set of operators O , the goal facts G and the initial facts I .

You must also define the pre, add, and del functions.



STRIPS model

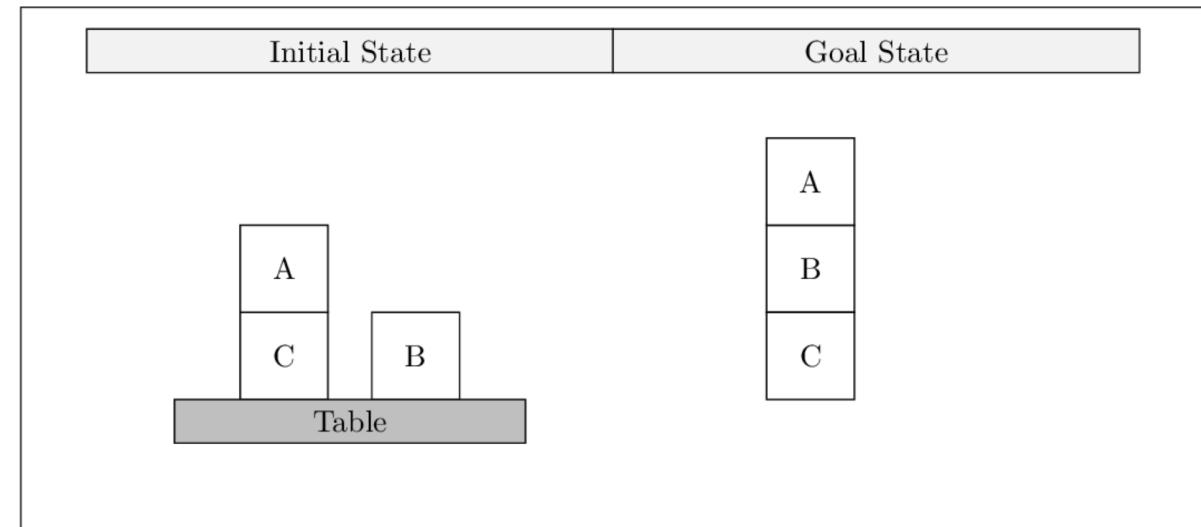
- $F := \{\text{on}(x, y), \text{onTable}(x), \text{clear}(x), \text{holding}(x), \text{armFree}\}$
- $I := \{\text{on}(A, C), \text{onTable}(C), \text{onTable}(B), \text{clear}(A), \text{clear}(B), \text{armFree}\}$
- $G := \{\text{on}(A, B), \text{on}(B, C)\}$



Operators (Stack & Unstack)

O:=

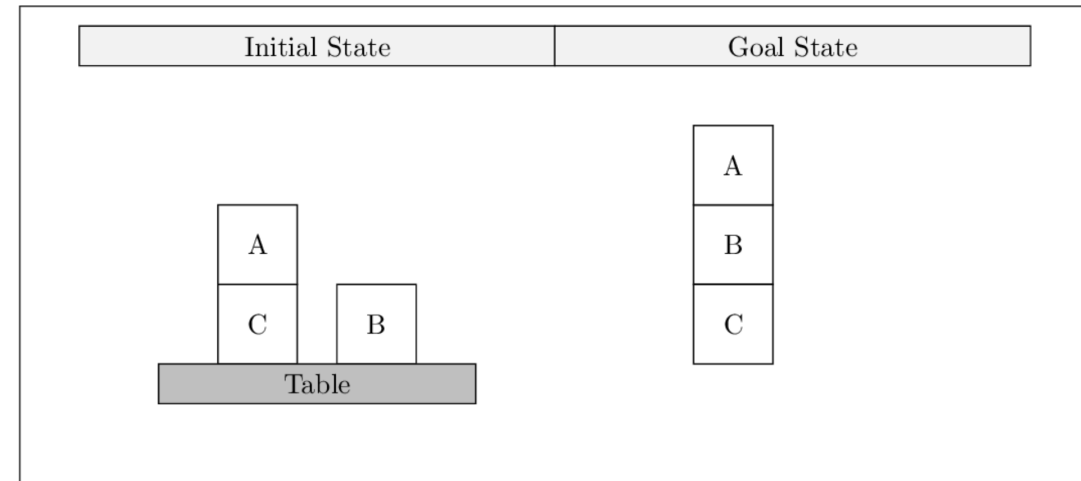
```
{ stack(x,y): =  
    prec:= {holding(x), clear(y)}  
    add:= {clear(x), on(x,y), armFree}  
    del:= {clear(y), holding(x)}  
    | x, y ∈ {A, B, C} and  $x \neq y$   
}  
^ {unstack(x,y): =  
    prec:= {on(x,y), clear(x), armFree}  
    add:= {holding(x), clear(y)}  
    del:= {clear(x), on(x,y), armFree}  
    | x, y ∈ {A, B, C} and  $x \neq y$   
}
```



Operators (putdown & pickup)

\wedge { putdown(x): =
 prec:= {holding(x) }
 add:= {clear(x), onTable(x), armFree}
 del:= {holding(x)}
 | $x, y \in \{A, B, C\}$
}

\wedge {pickup(x): =
 prec:= {onTable(x), clear(x), armFree}
 add:= {holding(x)}
 del:= {clear(x), onTable(x), armFree}
 | $x, y \in \{A, B, C\}$
}



Does $x \neq y$ constraint matter?

How many operators in total?

|O| without $x \neq y$:

Stack(x,y), Unstack(x,y):

Putdown(x), Pickup(x):

In total:

|O| with $x \neq y$:

Stack(x,y), Unstack(x,y):

Putdown(x), Pickup(x):

In total:

How many operators in total

|O| without $x \neq y$:

Stack(x, y), Unstack(x, y): 3*3 each

Putdown(x), Pickup(x): 3 each

In total: $3*3 + 3*3 + 3 + 3 = 24$

|O| with $x \neq y$:

Stack(x, y), Unstack(x, y): 2*3 each

Putdown(x), Pickup(x): 3 each

In total: $2*3 + 2*3 + 3 + 3 = 18$

Stack(x, x)- Prec: holding(x), clear(x)

{putdown(x): =

prec:= {holding(x) }

add:= {**clear(x)**, onTable(x), armFree}

del:= {**holding(x)**}

| x, y ∈ {A, B, C} }

{pickup(x): =

prec:= {onTable(x), clear(x), armFree}

add:= {**holding(x)**}

del:= {**clear(x)**, onTable(x), armFree}

| x, y ∈ {A, B, C} }

Unstack(x, x)- Prec: on(x, x), clear(x)

```
{ stack(x,y): =  
    prec:= {holding(x), clear(y)}  
    add:= {clear(x), on(x,y), armFree}  
    del:= {clear(y), holding(x)}  
    | x, y ∈ {A, B, C} }
```

```
^ {unstack(x,y): =  
    prec:= {on(x,y), clear(x), armFree}  
    add:= {holding(x), clear(y)}  
    del:= {clear(x), on(x,y), armFree}  
    | x, y ∈ {A, B, C} }
```

Problem 2

Implement your STRIPS model in PDDL.

Remember that a PDDL implementation is split between two files: a domain file (also known as an “operator” file) and a problem file (also known as a “fact” file).

PDDL

PDDL is not a propositional language:

- Representation is lifted, using **object variables** to be instantiated from a finite set of **objects**. (Similar to predicate logic)
- **Action schemas** parameterized by objects.
- **Predicates** to be instantiated with objects.

A PDDL planning task comes in two pieces:

- The **domain file** and the **problem file**.
- The problem file gives the objects, the initial state, and the goal state.
- The domain file gives the predicates and the operators; each benchmark domain has *one* domain file.

Platform to run PDDL

- With python and notebook
- Online platform: <http://editor.planning.domains/>
- VS Code, (Sublime) with PDDL extensions

Problem 3

- Blockworld can be modeled with only 2 actions instead of 4.
- The robot can **pick up a block** and **put it down on another block (or the table) in a single action**.
- You've got actions Move(Block, FromTable, ToBlock) and Move(Block, FromBlock, ToTable).
- You now no longer need to keep track of what the robot is holding or if the hand is empty.
- Implement a STRIPS model of this “2-operation” blocks-world in PDDL.

Sample solution

- https://editor.planning.domains/#read_session=1MLeB5c5s8