Hierarchical task and motion planning

Tomas Lozano-Perez
Leslie Pack Kaelbling
MIT CSAIL

Real robot meets real world

Real robot meets real world: large spaces

Configuration space

- joint angles of robot
- base pose
- positions and orientations of all objects in house
- are the dishes clean?
- is the stove on?
- are the leftovers edible?

Real robot meets real world: long horizon

Steps to clean kitchen:

- put away food (10 items)
- wash dishes (40 items)
- put away dishes (40 items)
- clean surfaces (10 items)

Or:

- 1,000 pick, place, or wipe

Or:

- 100,000 linearly interpolated joint motions

Real robot meets real world: uncertainty

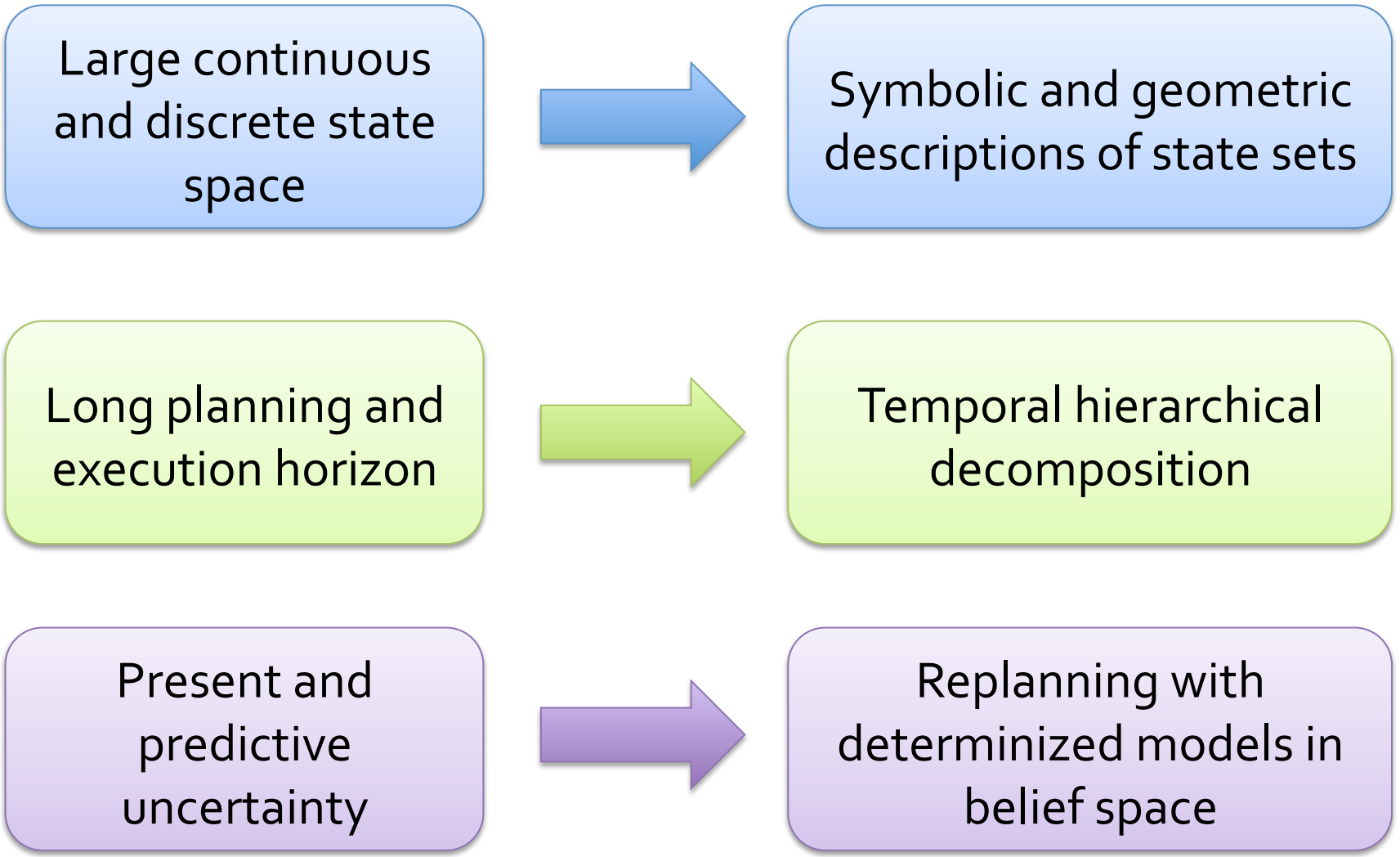
Current-state uncertainty

- What is inside the tupperware?
- Is the dishwasher clean?
- What is the exact pose of the pot?
- What's the friction of a wet dish?

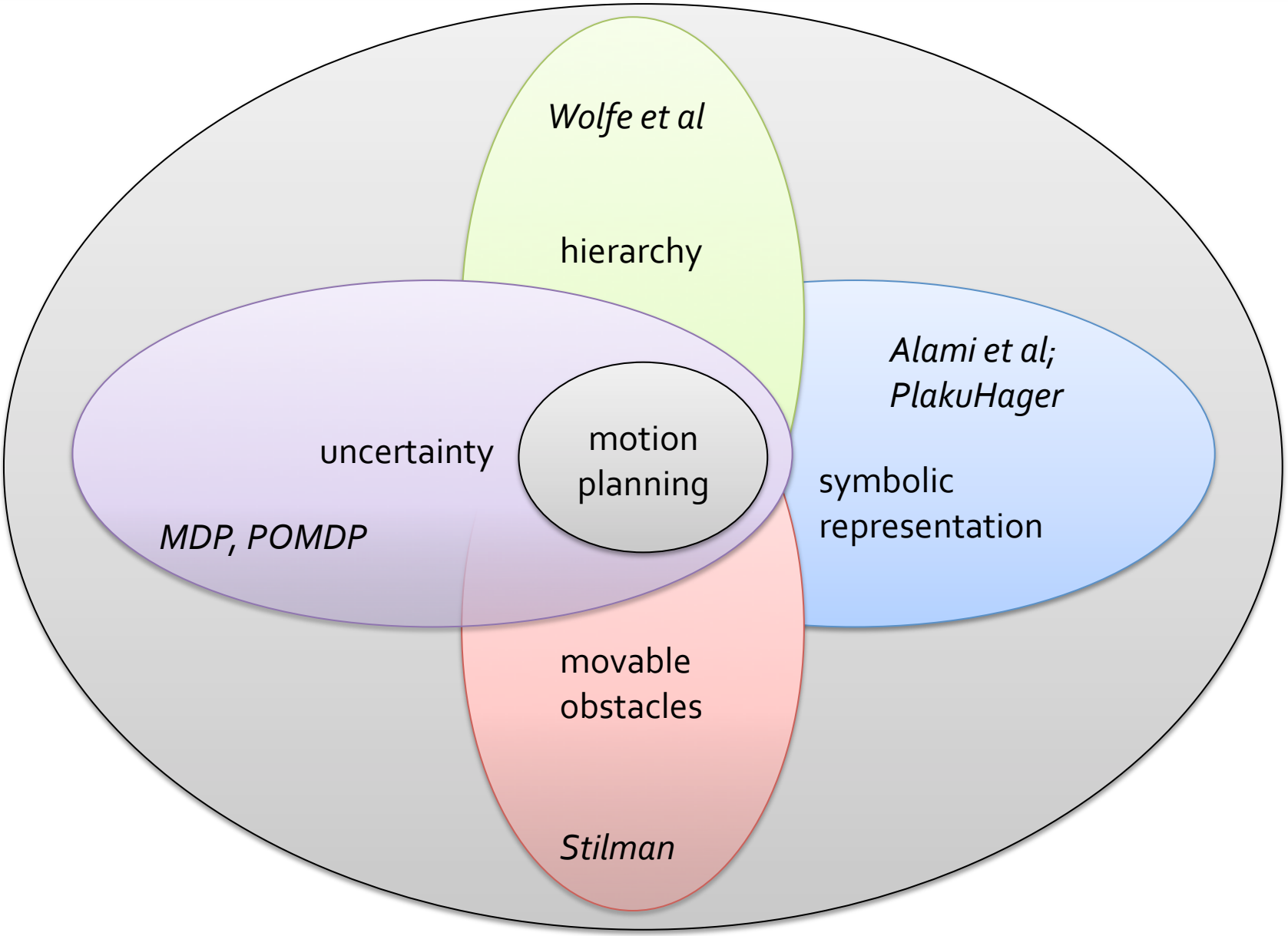
Predictive uncertainty

- What will happen when the robot lifts the cookie sheet?
- What is the error in the motor control?
- When will the inhabitants come home?

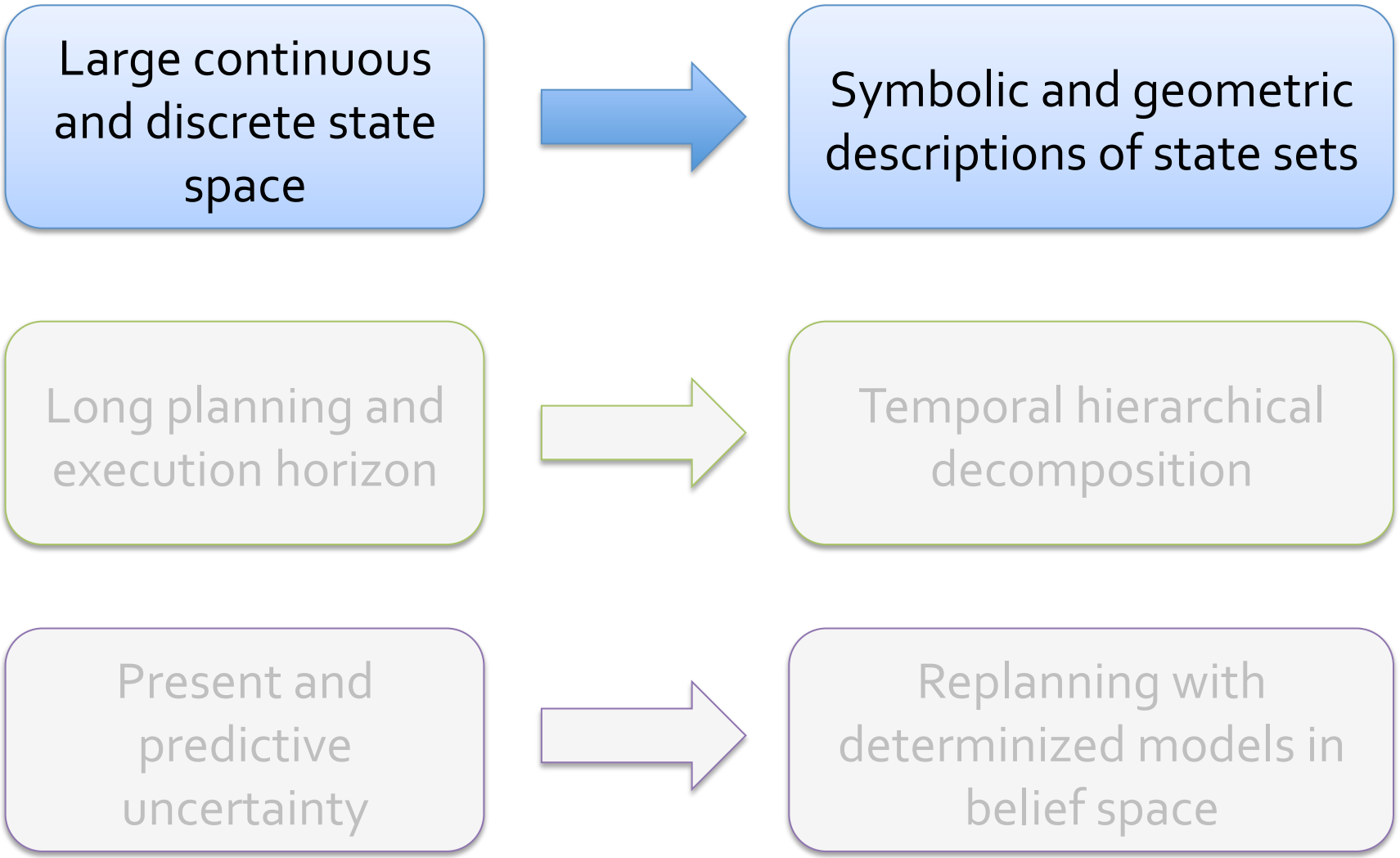
Addressing real world challenges



Some related approaches



Addressing real world challenges



Symbolic and geometric descriptions of state sets

Symbols for non-geometric properties

- are the dishes clean? ***Clean(dish24)***
- is the stove on? ***On(stove)***
- are the leftovers edible? ***Edible(lasagna)***

Symbols for geometric and physical abstractions

- is the beer in the refrigerator? ***In(beer, refrigerator)***
  - region containment
- is the robot holding the pot? ***Holding(pot)***
  - contacts, force closure, …

Short descriptions for large sets

Classical symbolic action descriptions

```
Put(Block, Target):
  exists: Source
  pre: On(Block, Source),
       Clear(Block),
       Clear(Target)
  result:
      On(Block, Target),
         not On
  (Block,Source),
      Clear(Source),
      not Clear(Target)
```

- Requires complete symbolic description of world state
- No explicit reasoning: requires all results to be asserted
- Requires enumeration of all possible values of variables

Symbolic planning in the real world

Classical formulation

- Requires complete symbolic description of world state
- Requires enumeration of all possible values of variables
- No explicit reasoning: requires all results to be asserted

Real-world formulation

- Symbolic description incomplete in infinite domain (regions, …)
- Generators for values of existential variables
- Extra reasoning about consistency and entailment for geometric properties

Alami et al; Cambon et al; Plaku and Hager

Specific start state; abstract goal

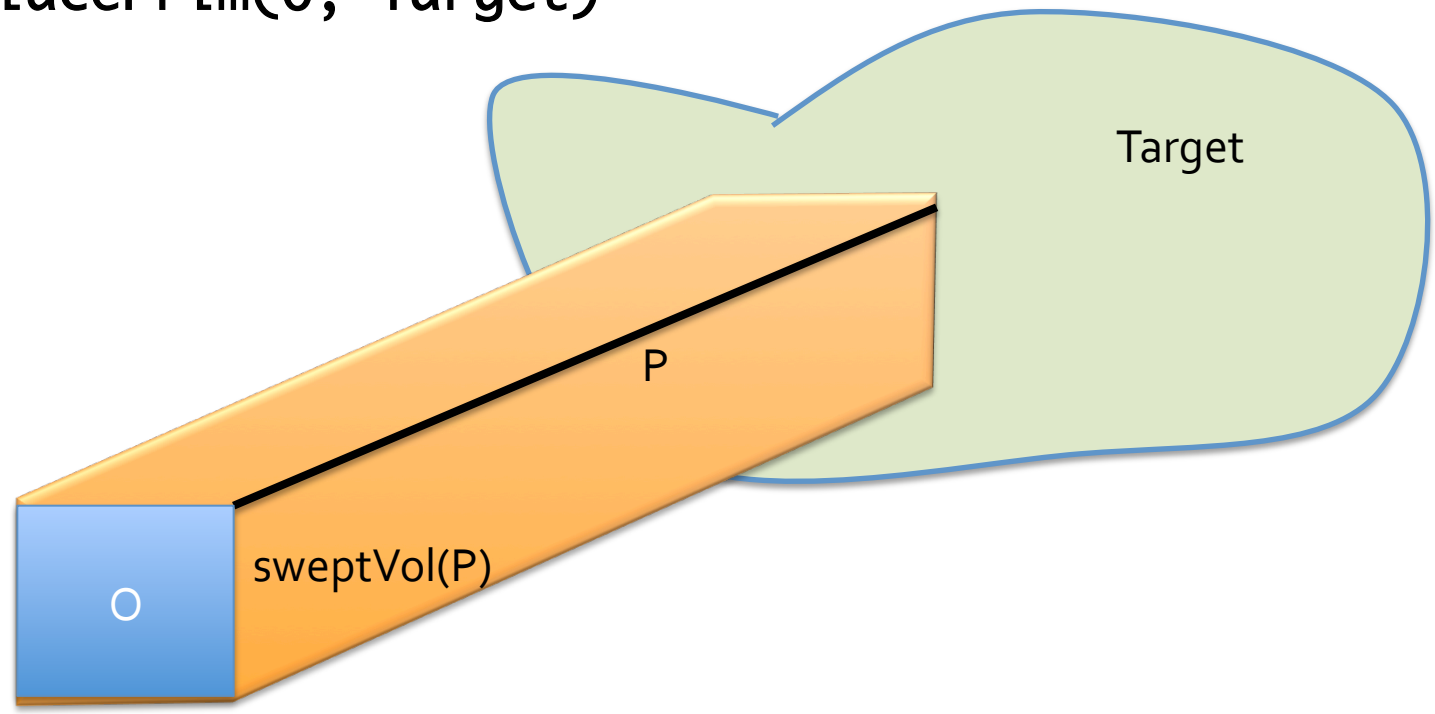
Initial state known in geometric detail

Goal set is abstract, symbolic

tidy(house) ∧ charged(robot)

Planning operator: Place

```
Place(O, Target):
   exists: P ∈ generatePlacePaths(O, Target)
   pre: ClearX(sweptVol(P), O), Holding(O)
   result: In(O, Target)
   prim: PlacePrim(O, Target)
```

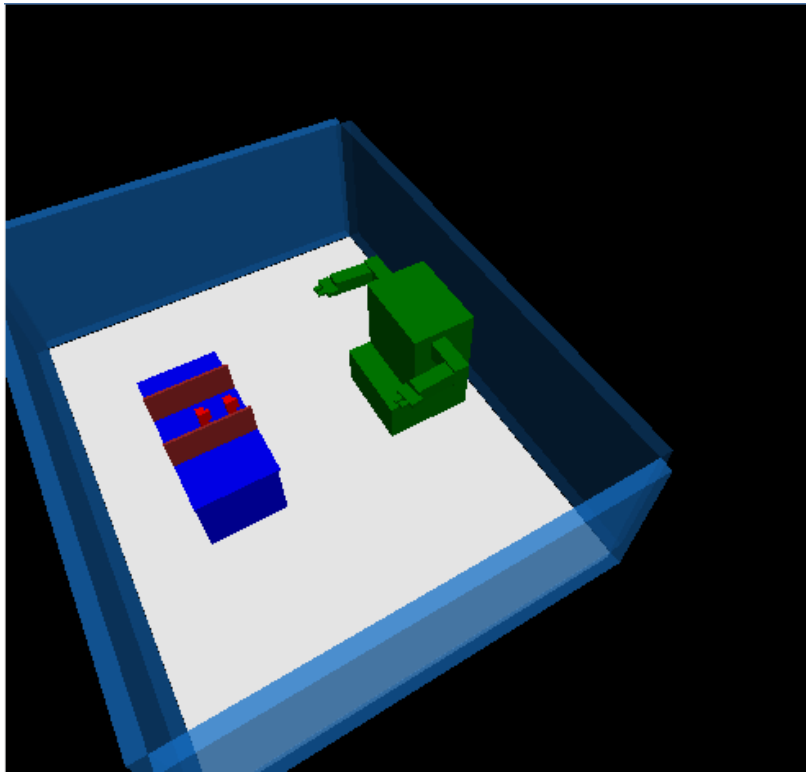


Stilman and Kuffner

Generators

Efficient conservative planner

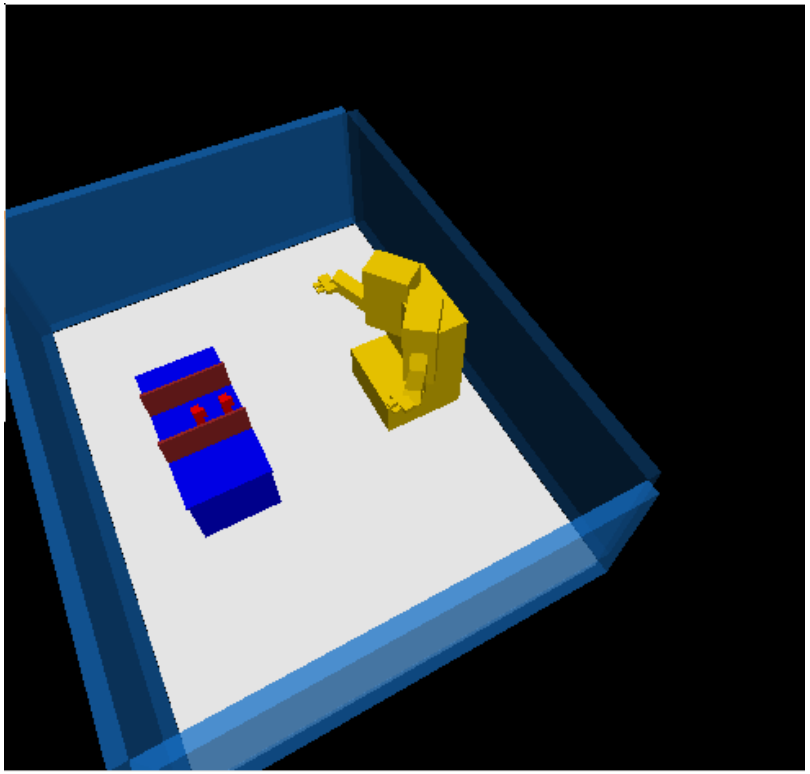
- Simplified robot model
- Objects: x, y, z, th
- Grasp selection



Primitives

Your favorite motion planner

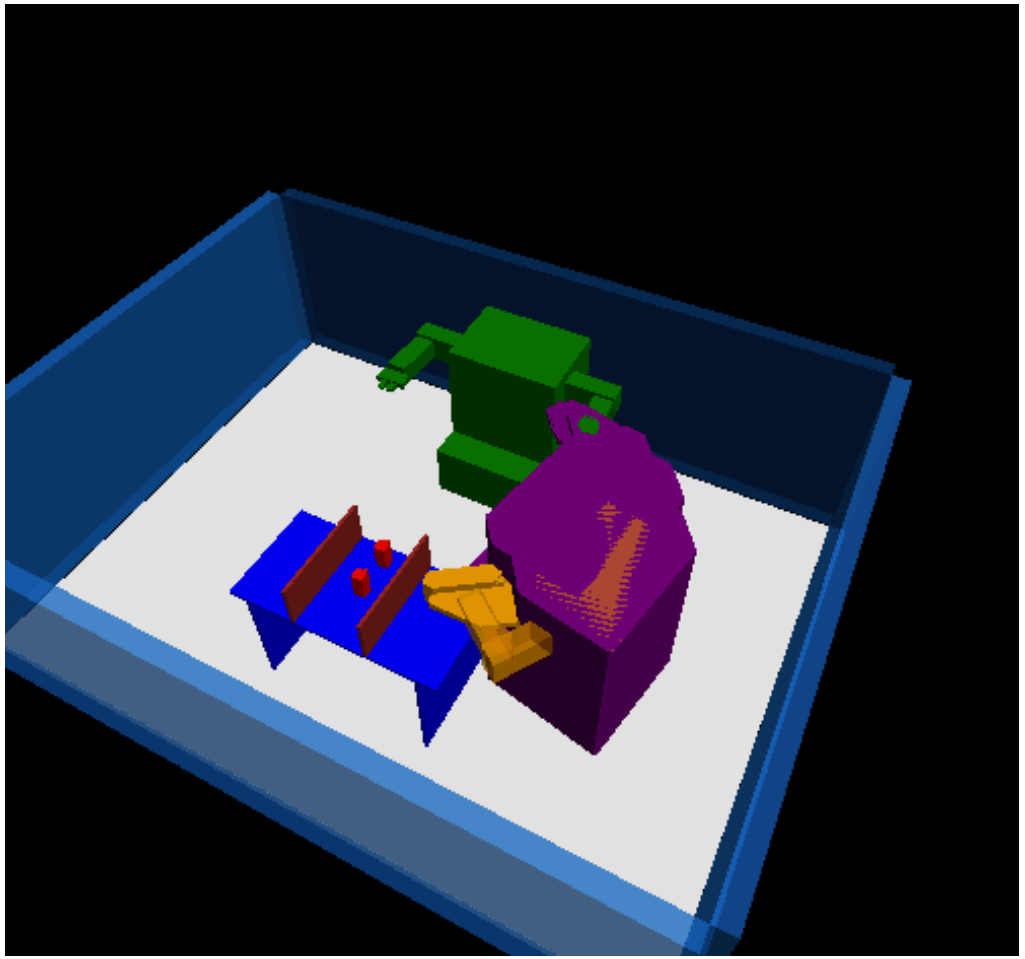
- Accurate robot model
- No reasoning about objects
- Grasps from generator



Generators

```
Place(O, Target):
   exists: P ∈ generatePlacePaths(O, Target)
```

- visibility graph planner
- fail fast
- conservative
- not the path we'll use: certificate that one exists

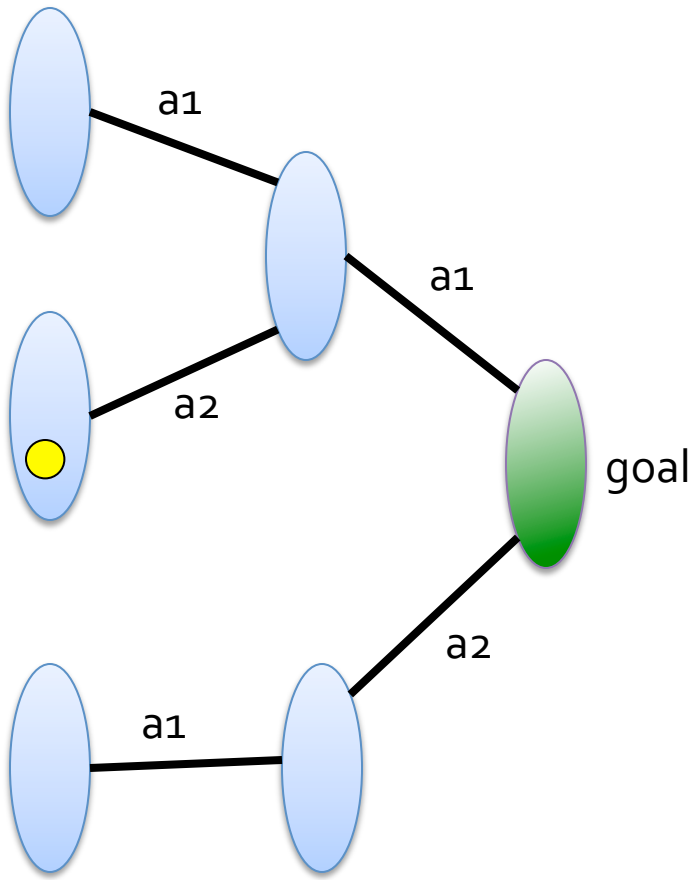


Goal Regression / Pre-image backchaining

Weakest precondition of goal set under each action sequence

Test whether start state is in a pre-image

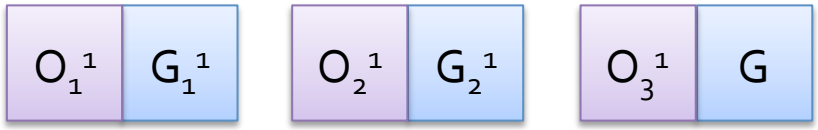
Represent goal and pre-images as conjunctions of fluents



Why regression?

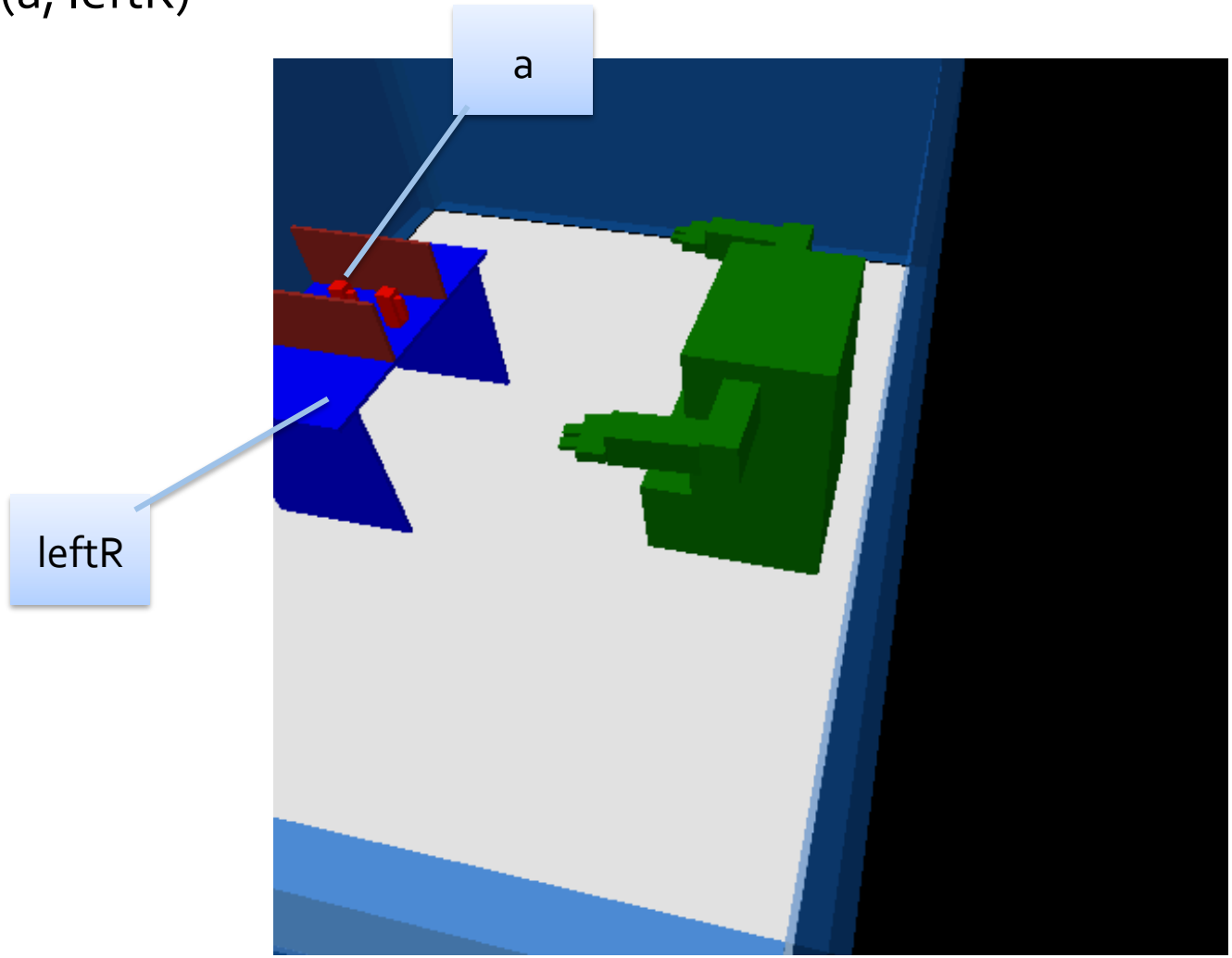
- Compact logical description of infinite pre-image sets
- Just need to test whether starting geometry is in the set
- **Goal-directedness:**
  - In natural settings, the initial state is incredibly detailed
  - Forward branching factor is huge!!
  - Goal has simple symbolic description

- Clear criteria for execution monitoring



Regression and generation

Goal: In(a, leftR)

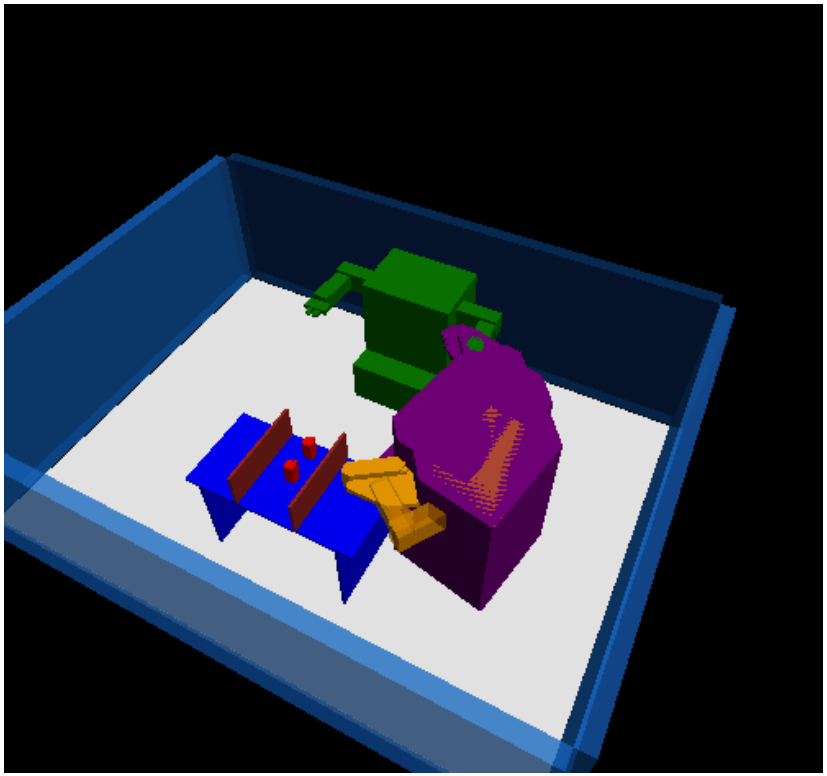


Regression and generation

Goal: In(a, leftR)

O: Place(a, leftR)

G: **Holding(a)**, ClearX(Swept(PlacePath(a, leftR)))



Regression and generation

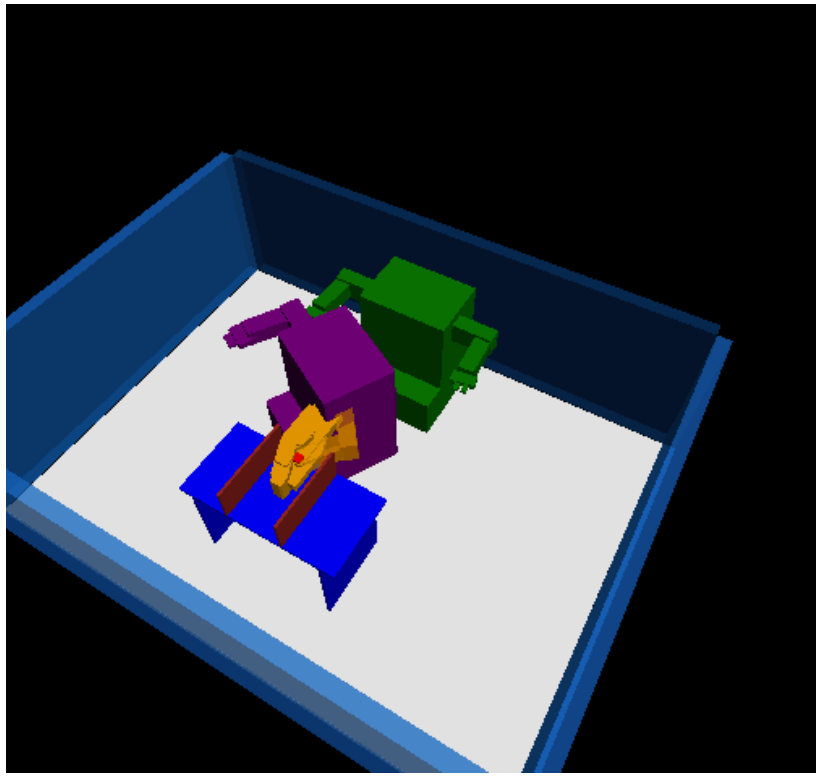
Goal: In(a, leftR)

O: Place(a, leftR)

G: **Holding(a)**, ClearX(Swept(PlacePath(a, leftR)))

O: Pick(a)

G: **ClearX(Swept(PickPath(a))**,
ClearX(Swept(PlacePath(a, leftR))



Regression and generation

Goal: In(a, leftR)

O: Place(a, leftR)

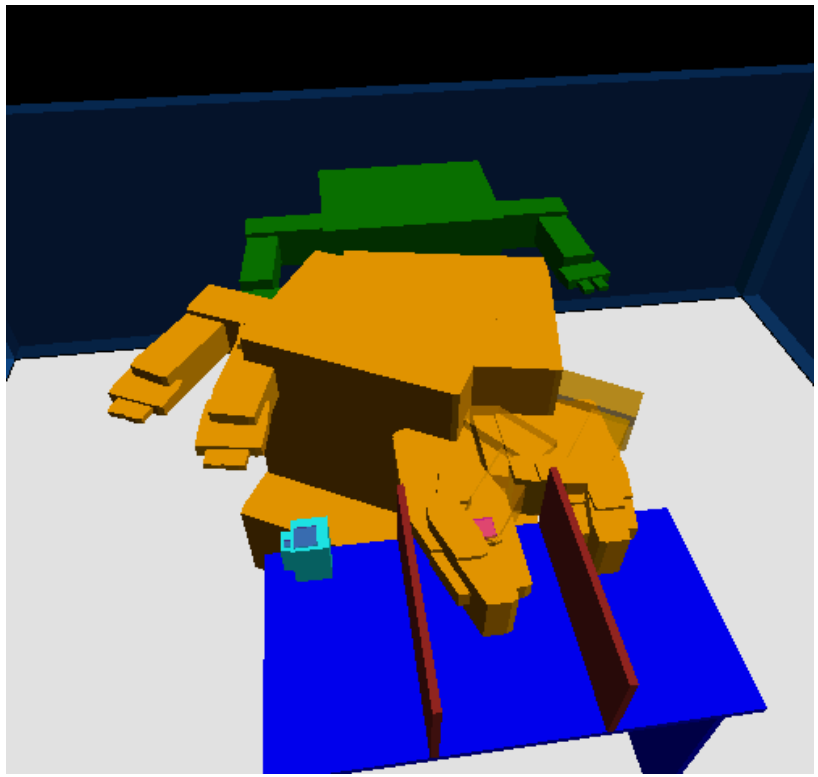
G: **Holding(a)**, ClearX(Swept(PlacePath(a, leftR)))

O: Pick(a)

G: **ClearX(Swept(PickPath(a))**,
ClearX(Swept(PlacePath(a, leftR))

O: Remove(b, Swept(PickPath(a))

G: **In(b, Parking(b))**,
ClearX(Swept(PlacePath(a, leftR))



Regression and generation

Goal: In(a, leftR)

O: **Place(a, leftR)**

G: Holding(a), ClearX(Swept(PlacePath(a, leftR)))

O: **Pick(a)**

G: ClearX(Swept(PickPath(a)), ClearX(Swept(PlacePath(a, leftR))

O: Remove(b, Swept(PickPath(a))

G: In(b, Parking(b)), ClearX(Swept(PlacePath(a, leftR))

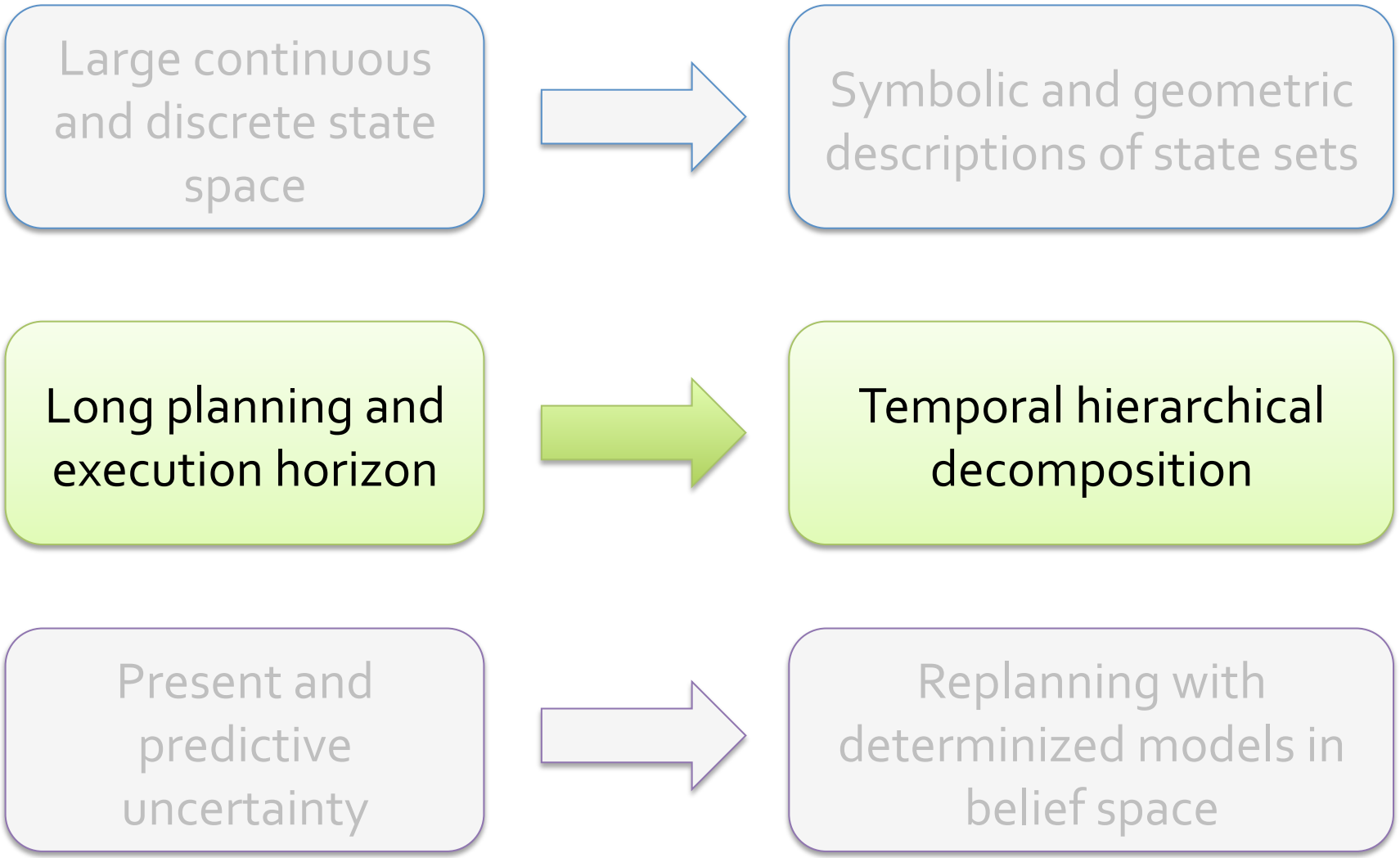
O: **Place(b, Parking(b))**

G: Holding(b), ClearX(Sw(PP(b, Park(b)))), ClearX(Sw(PlaceP(a, leftR))

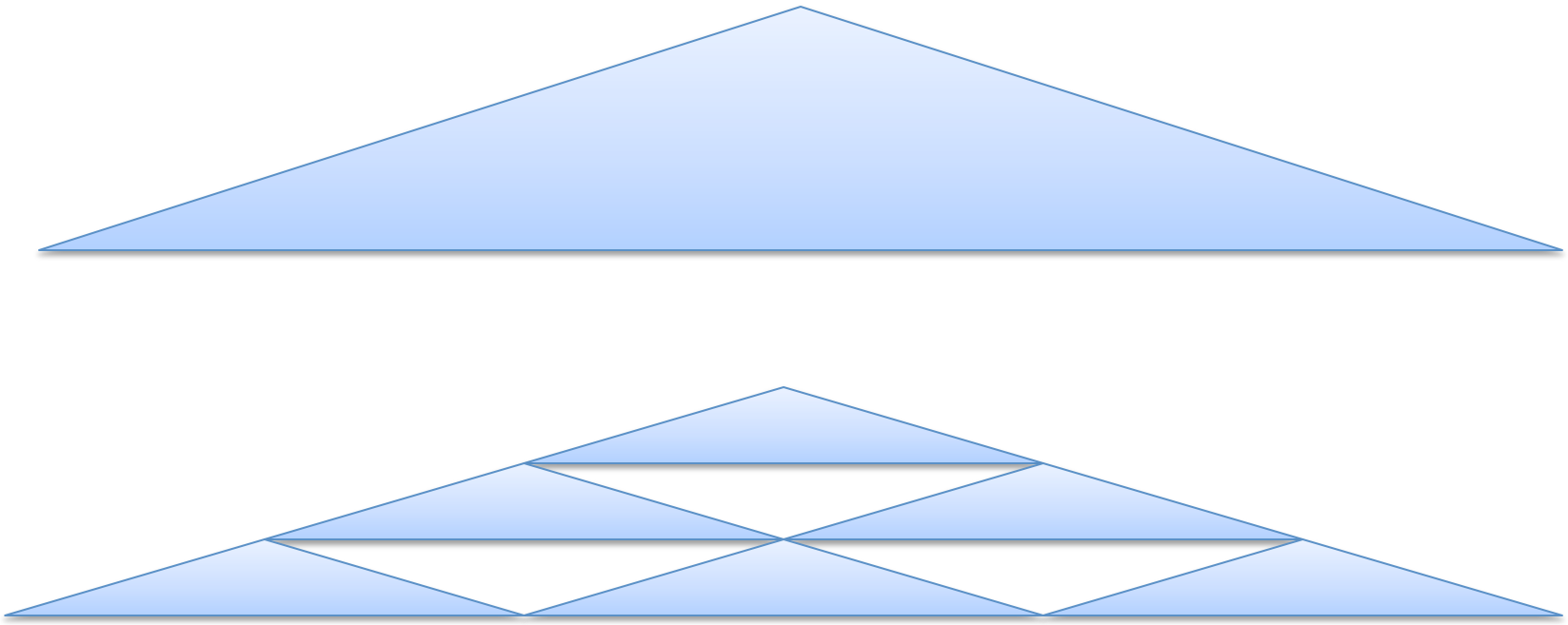
O: **Pick(b)**

G: ClearX(Sw(PickP(b))), ClearX(Sw(PP(b, Park(b)))),
ClearX(Sw(PlaceP(a, leftR))

Addressing real world challenges



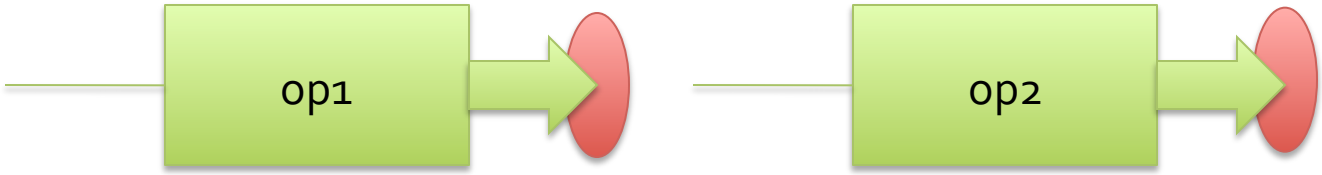
Hierarchy crucial for large problems



Hierarchical semantics

Subgoal is an abstract operator:

What does it mean to sequence two subgoals?



Depends on who gets to choose the outcome:

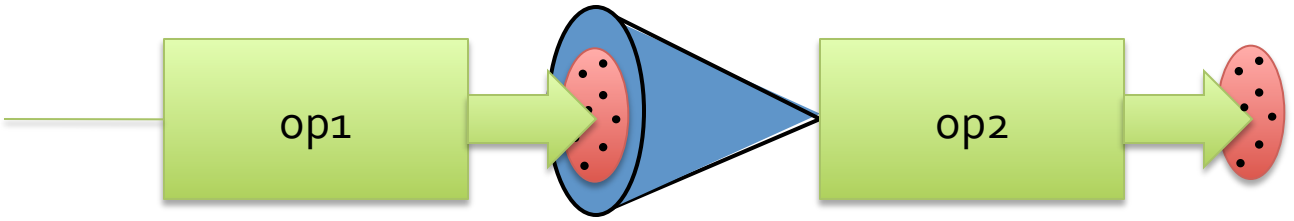
us

nature

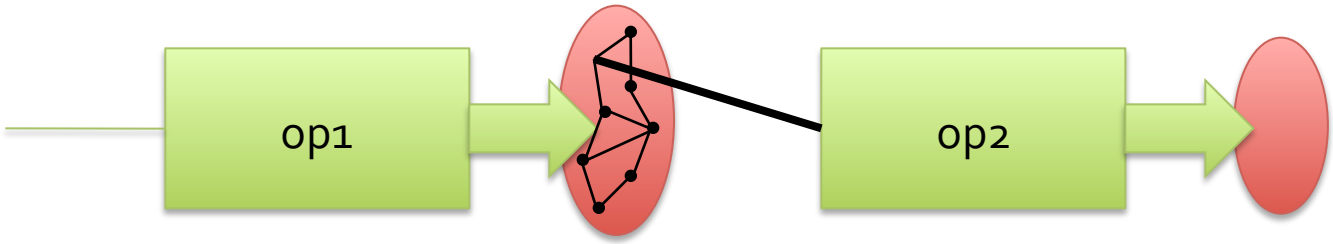
Wolfe, Marthi, Russell

Satanic Semantics

We have to handle any outcome the devil picks



Okay if: Preconditions of op2 can be achieved from any state resulting from op1



Postponing preconditions creates hierarchy

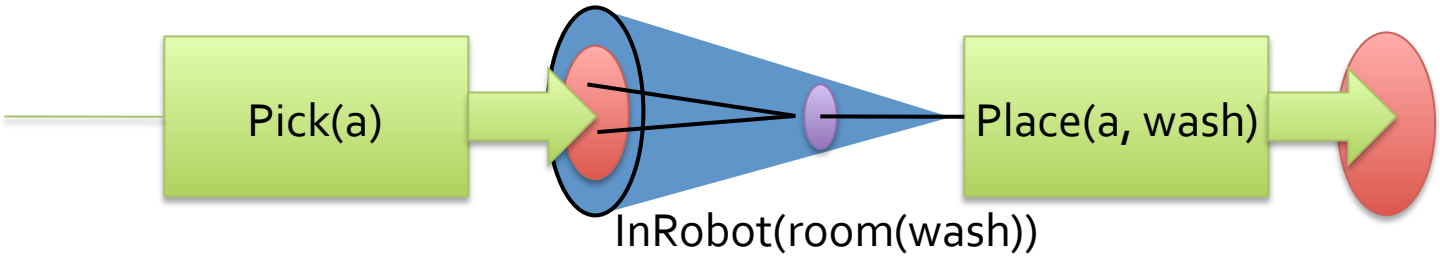
```
Pick(O):
pre: …
res: Holding(O)
```

```
Place(O, R):
pre: 0.Holding(O)
     1.InRobot(room(R))
res: In(O, R)
```

Postponing preconditions creates hierarchy

```
Pick(O):
pre: …
res: Holding(O)
```

```
Place(O, R):
pre: 0.Holding(O)
     1.InRobot(room(R))
res: In(O, R)
```

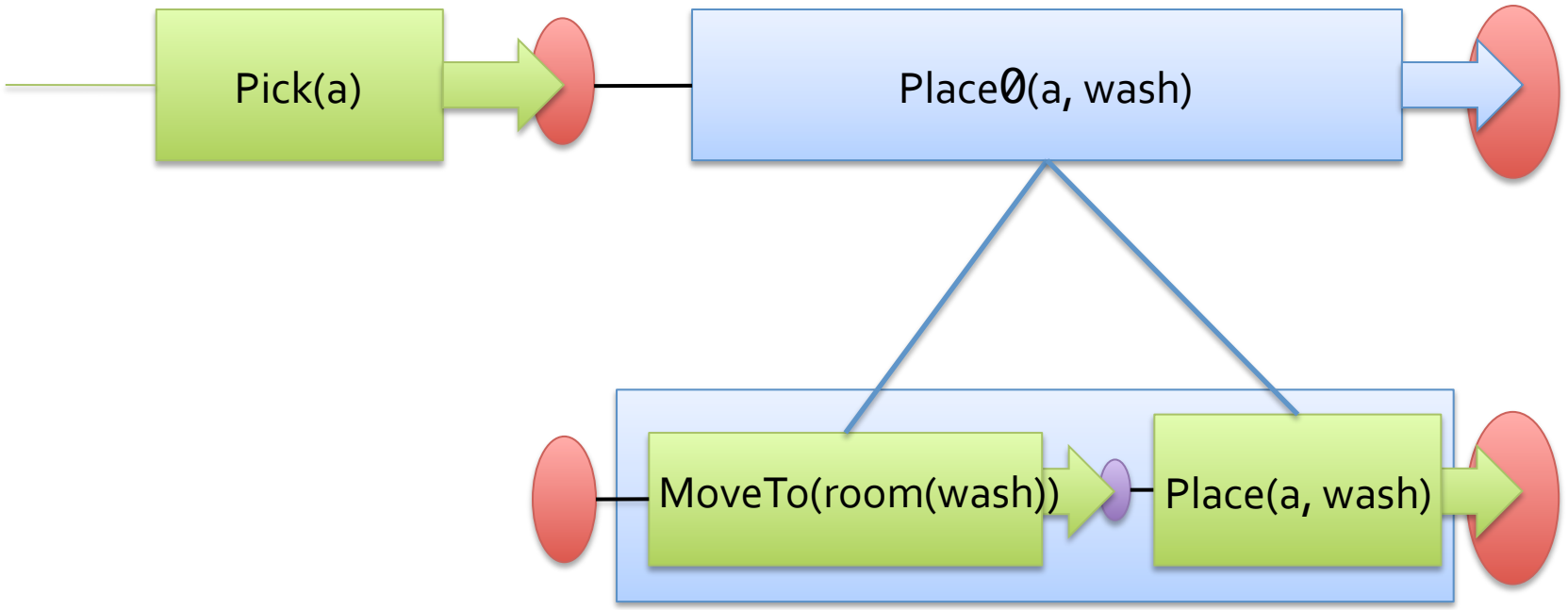


Precondition `InRobot(room(R))` is serialized with the `place` primitive

Hierarchical plan

```
Pick(O):
pre: …
res: Holding(O)
```

```
Place0(O, R):
pre: Holding(O)
res: In(O, R)
```

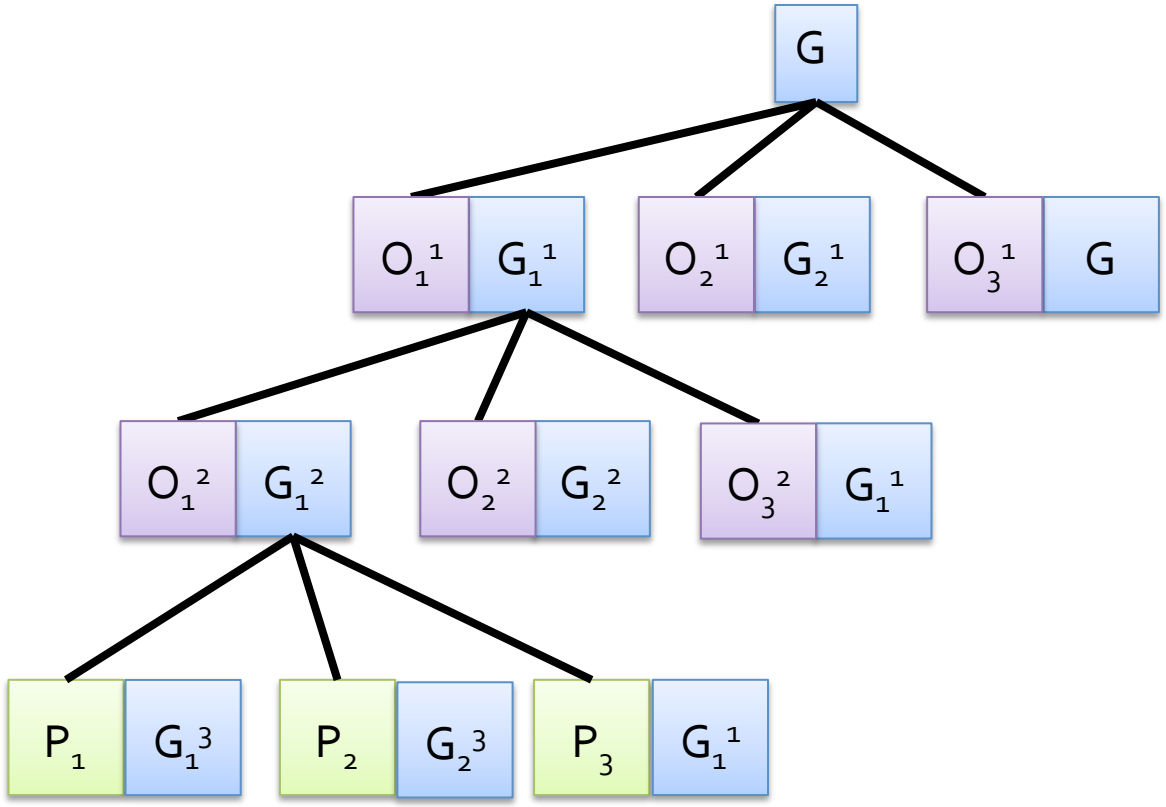


In the now

- Don't consider all the ways pick(a) could terminate: grasp of object, location of robot, ….
- When it is time to plan for Place0(a, wash), actual world state is start state

Planning in the now

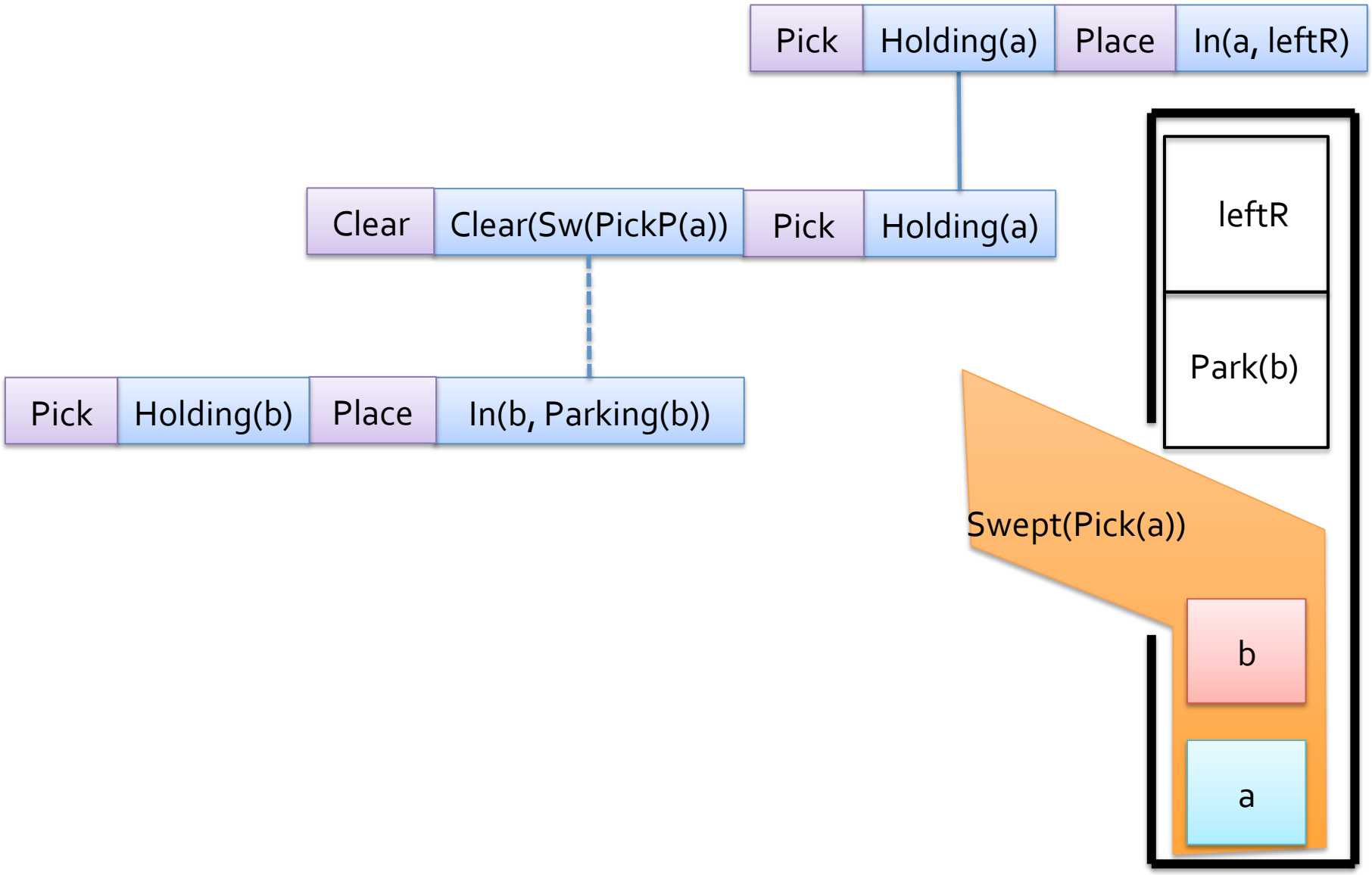
- maintain left expansion of plan tree
- each level uses a higher-fidelity model
- keep track of pre-image for each operation
- recursively plan to achieve those preconditions
- execute primitives



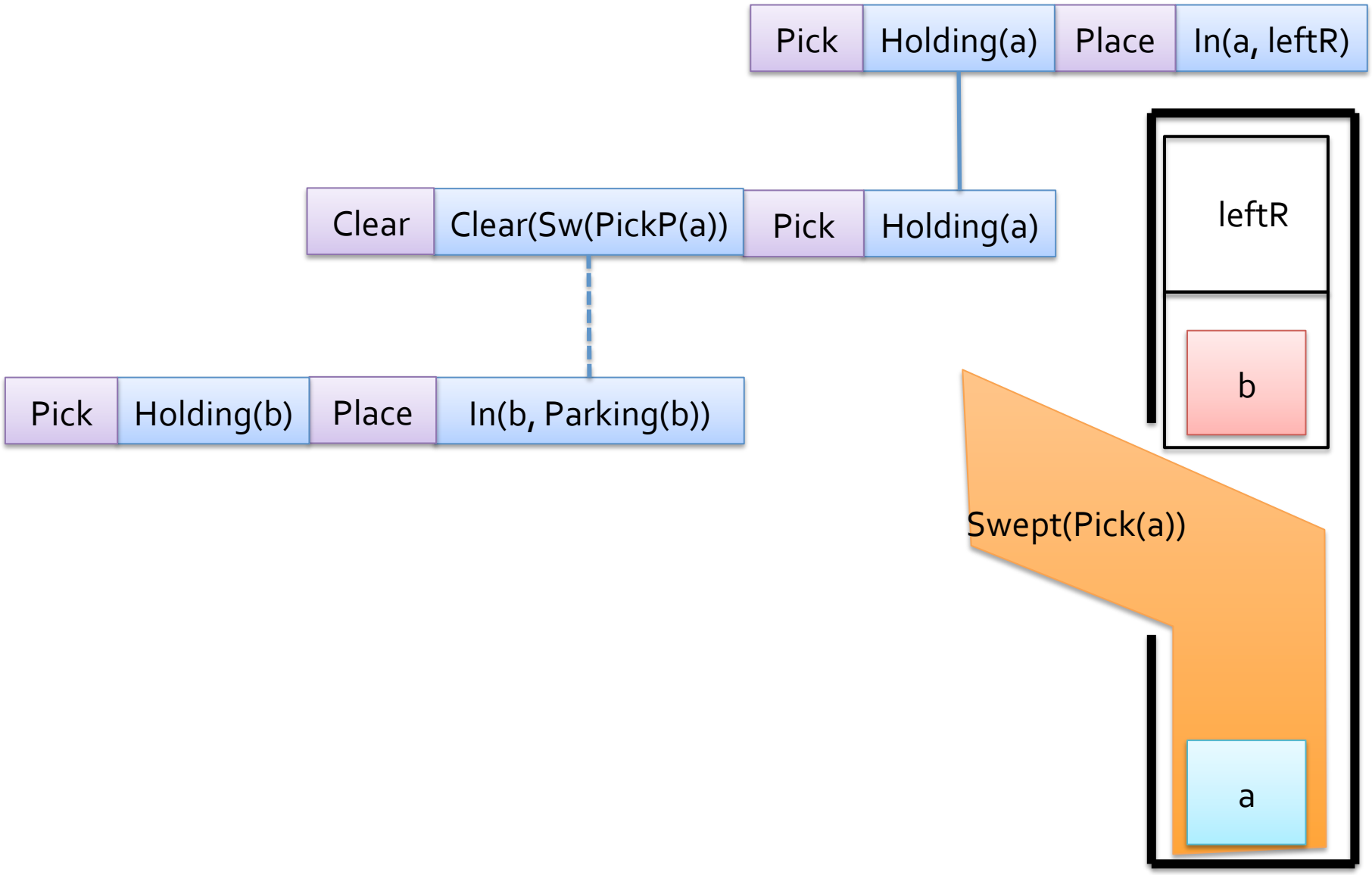
HPN Algorithm

```
HPN(currentState, goal, operators, absLevel, world):
    if holds(goal, currentState):
        return TRUE
    else p = PLAN(currentState, goal, operators, absLevel)
        for (o_i, g_i) in p
            if prim(o_i):
                currentState = world.execute(o_i)
            else HPN(currentState, g_i, operators,
                     NEXTLEVEL(absLevel, o_i), world)
```

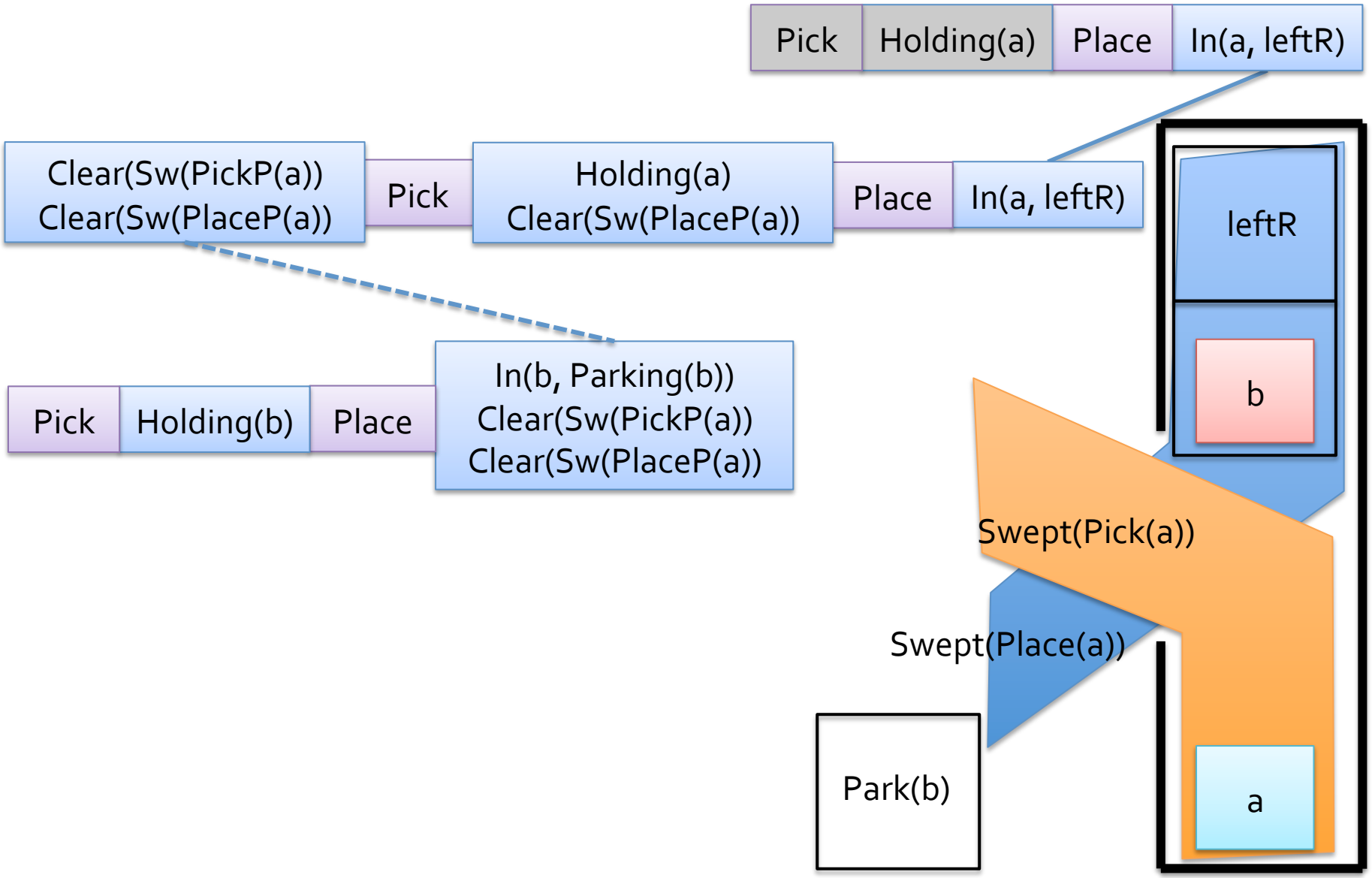
Hierarchical pick and place



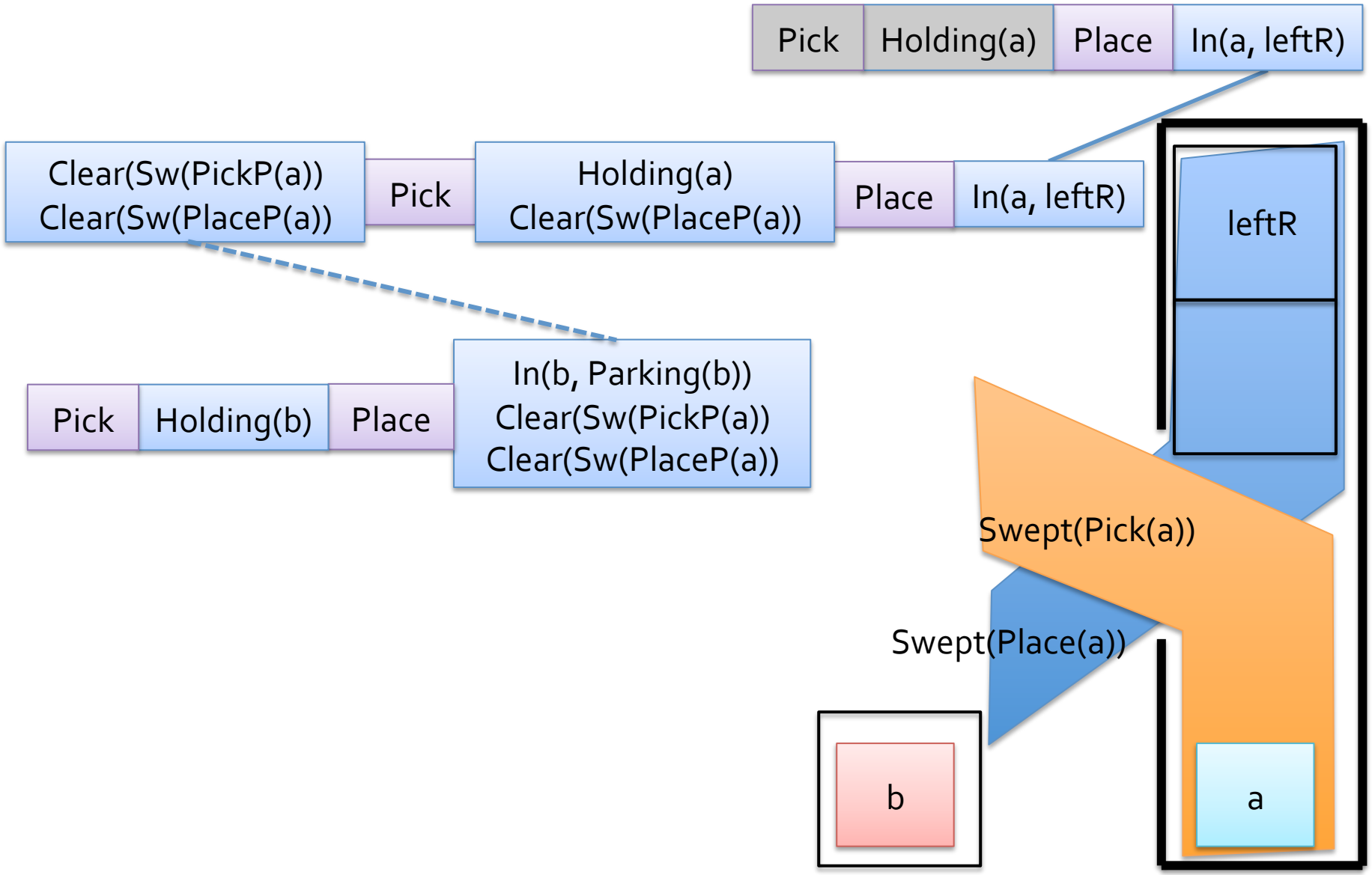
Hierarchical pick and place: too abstract!!



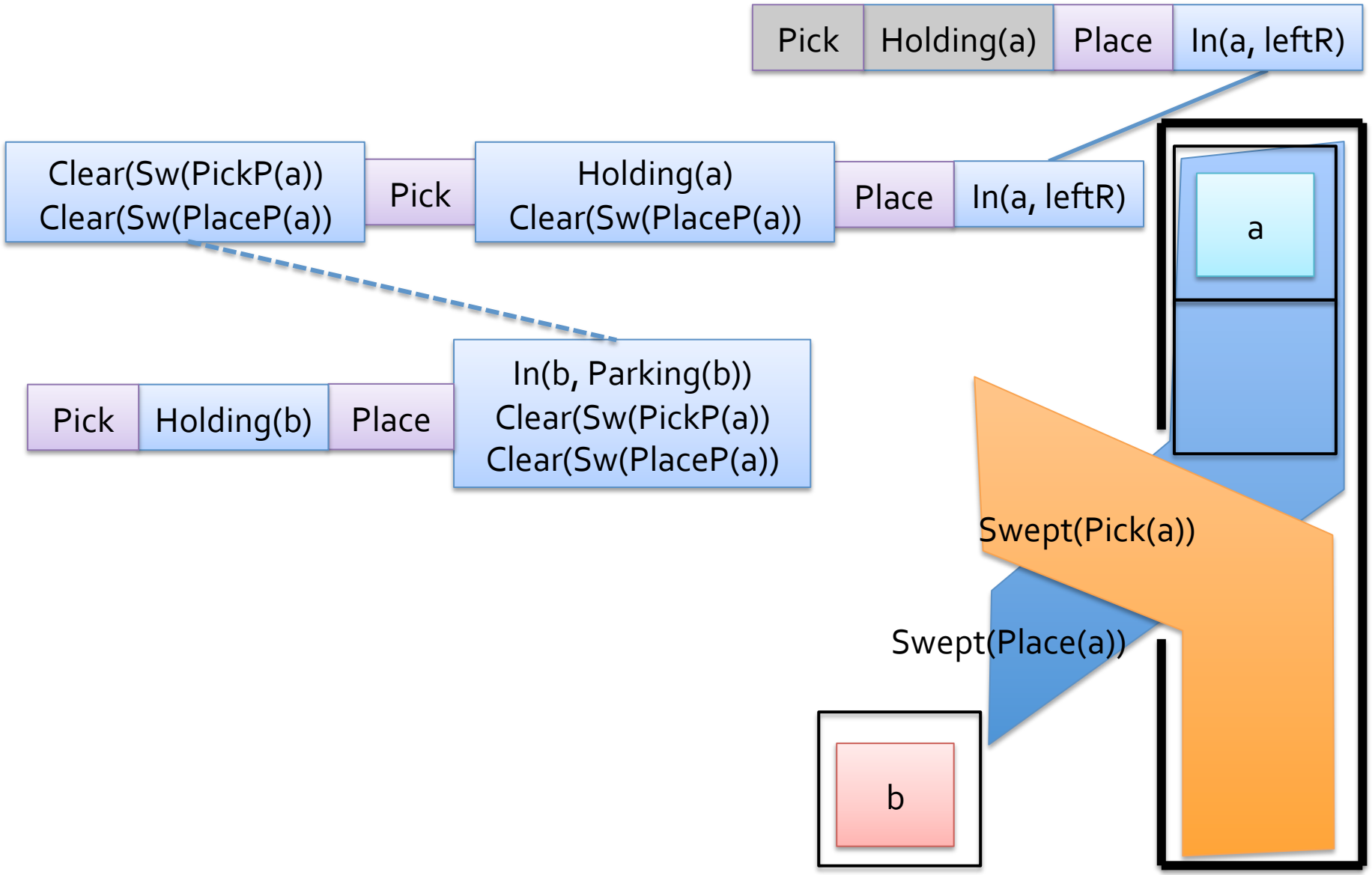
Decrease abstraction level



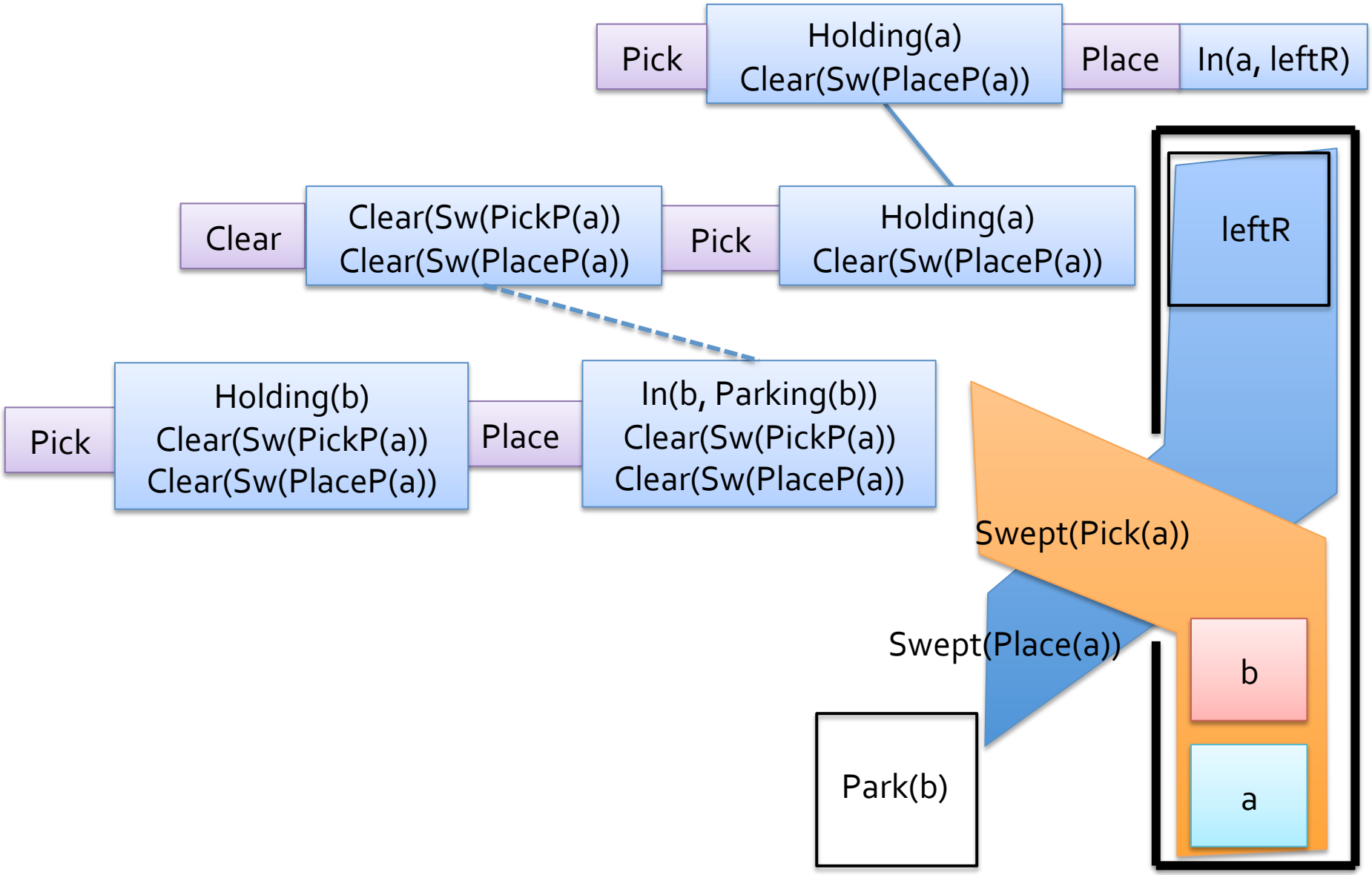
Decrease abstraction level:



Decrease abstraction level:



Hierarchical pick and place: consider clearX



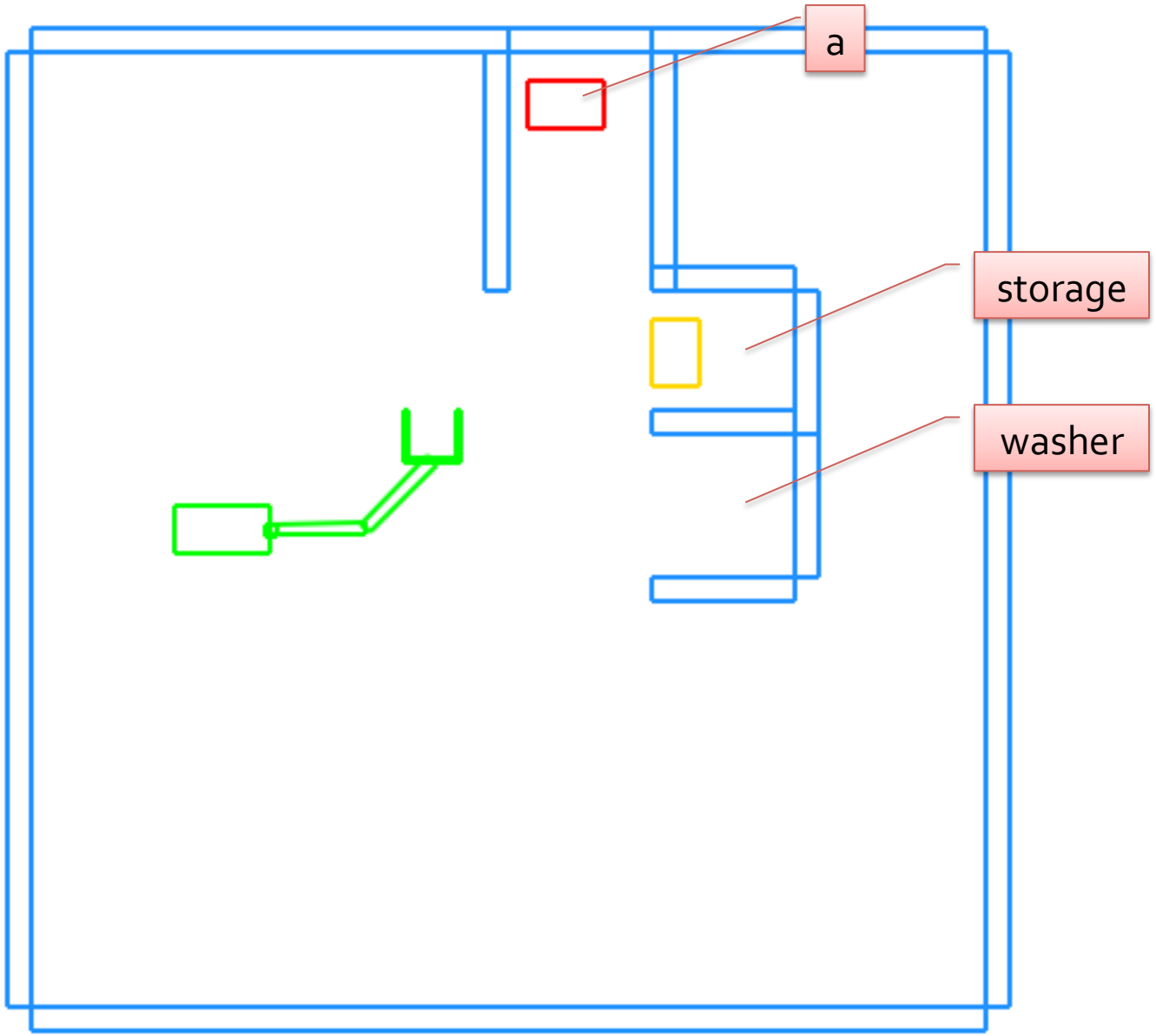
Correctness

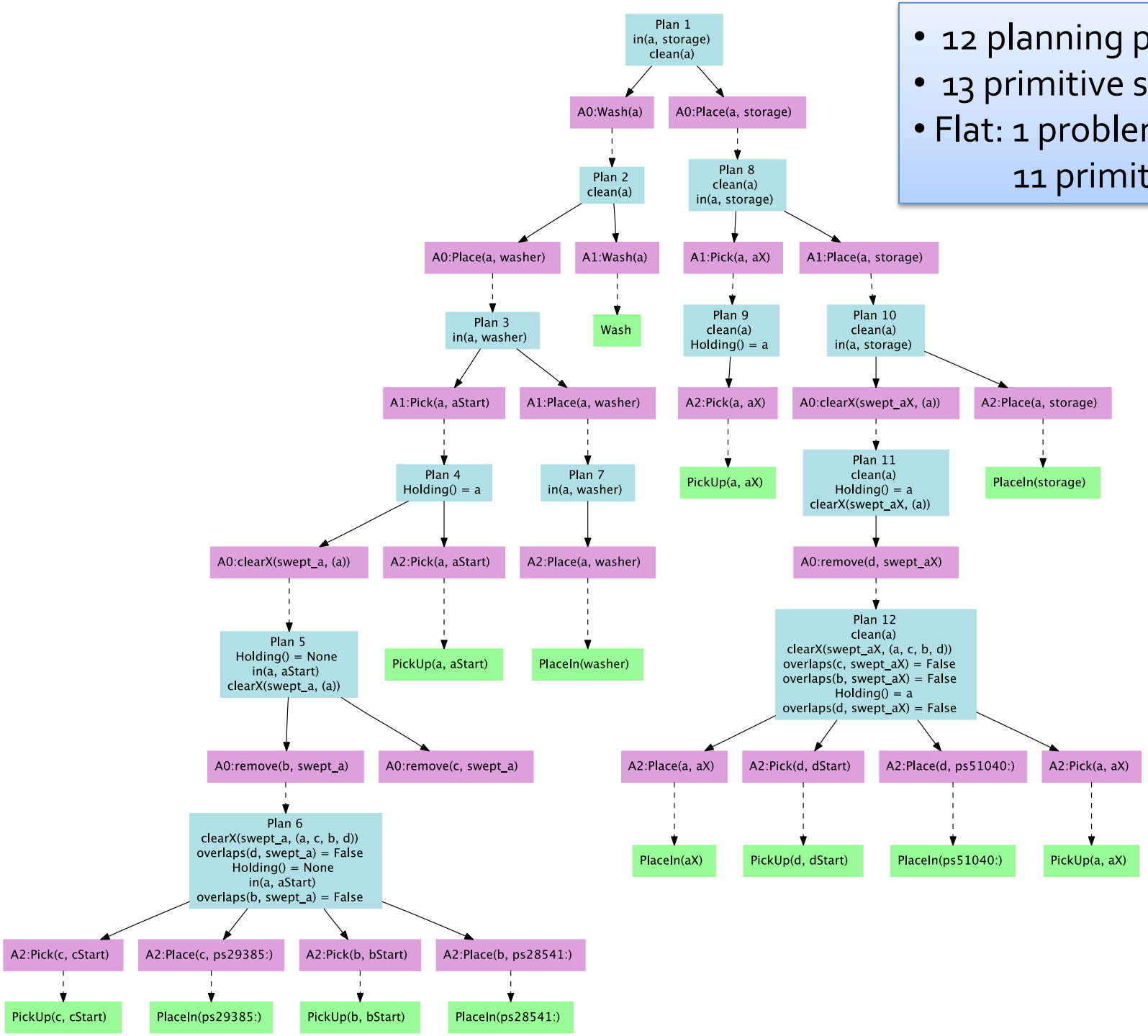
If

- the domain specified by operators ***ops*** at the most concrete abstraction level is a complete and correct formalization of the primitive actions of domain ***w***
- any state reachable from ***start*** is reachable from any other state reachable from ***start***, and
- some state in ***G*** is reachable from ***start***,

then executing ***HPN(start, G, ops, H, w)*** will cause domain ***w*** to be in a state $s \in G$.

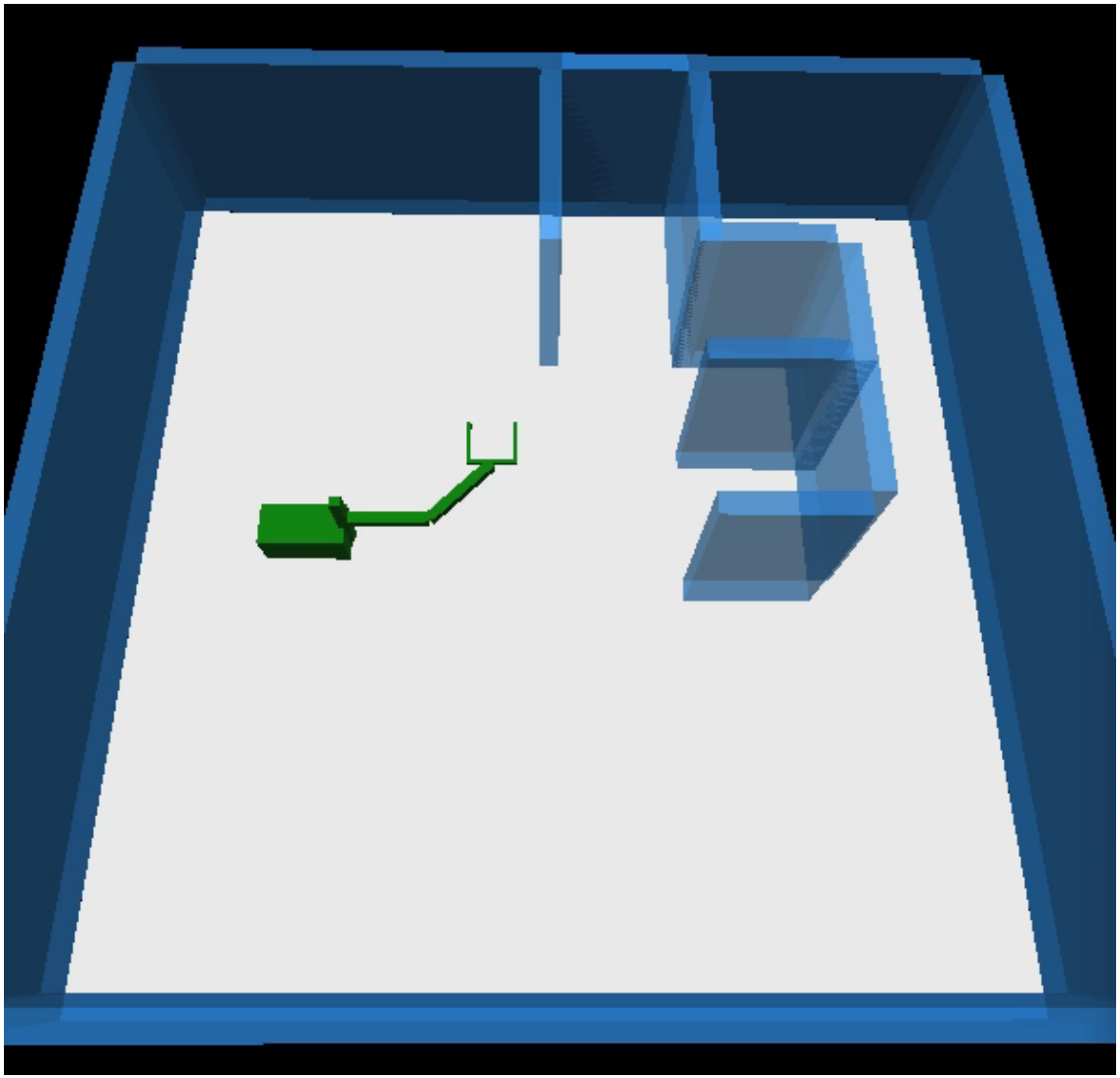
Wash a block and put it away





- 12 planning problems
- 13 primitive steps
- Flat: 1 problem,
  11 primitive steps

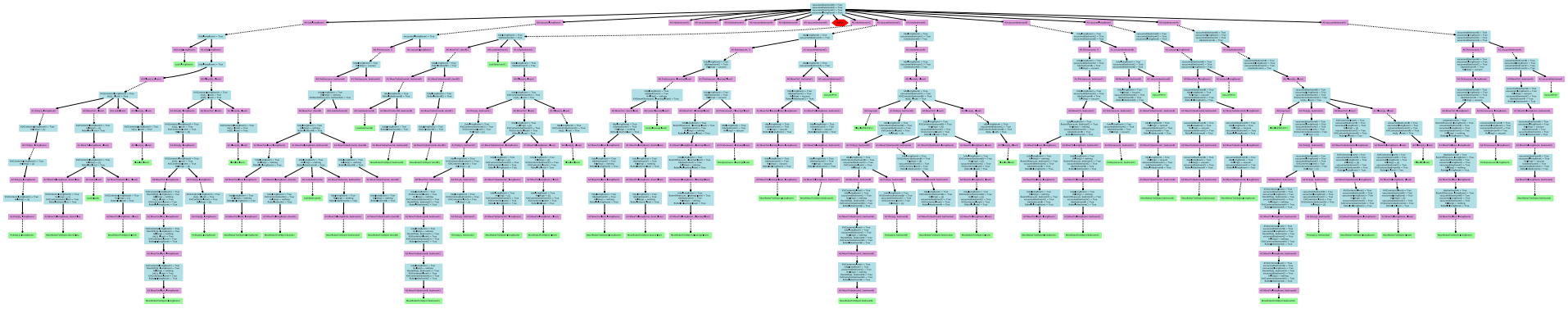
Wash a block and put it away



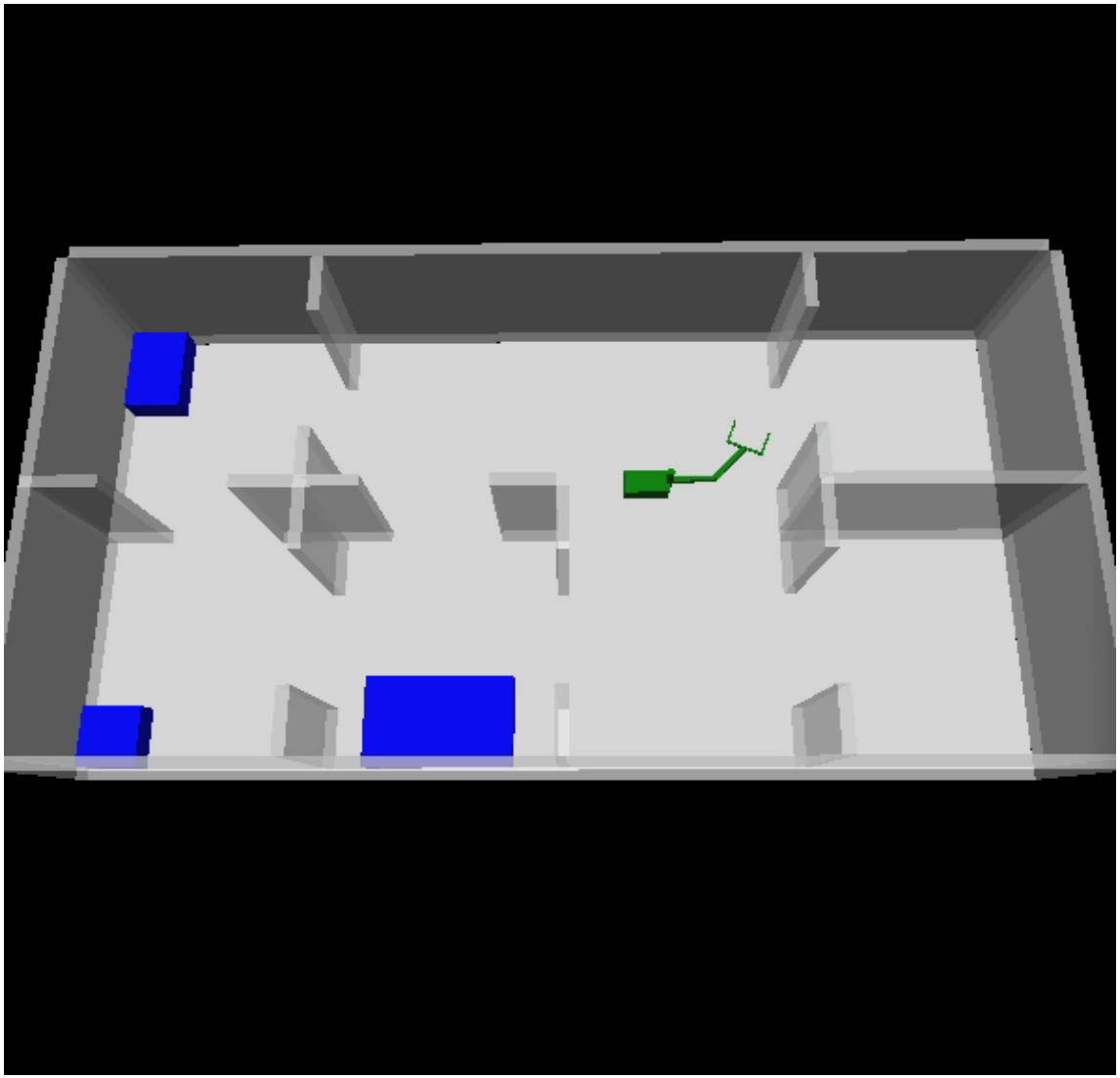
Cleaning house

Goal: mop four of the rooms in the house

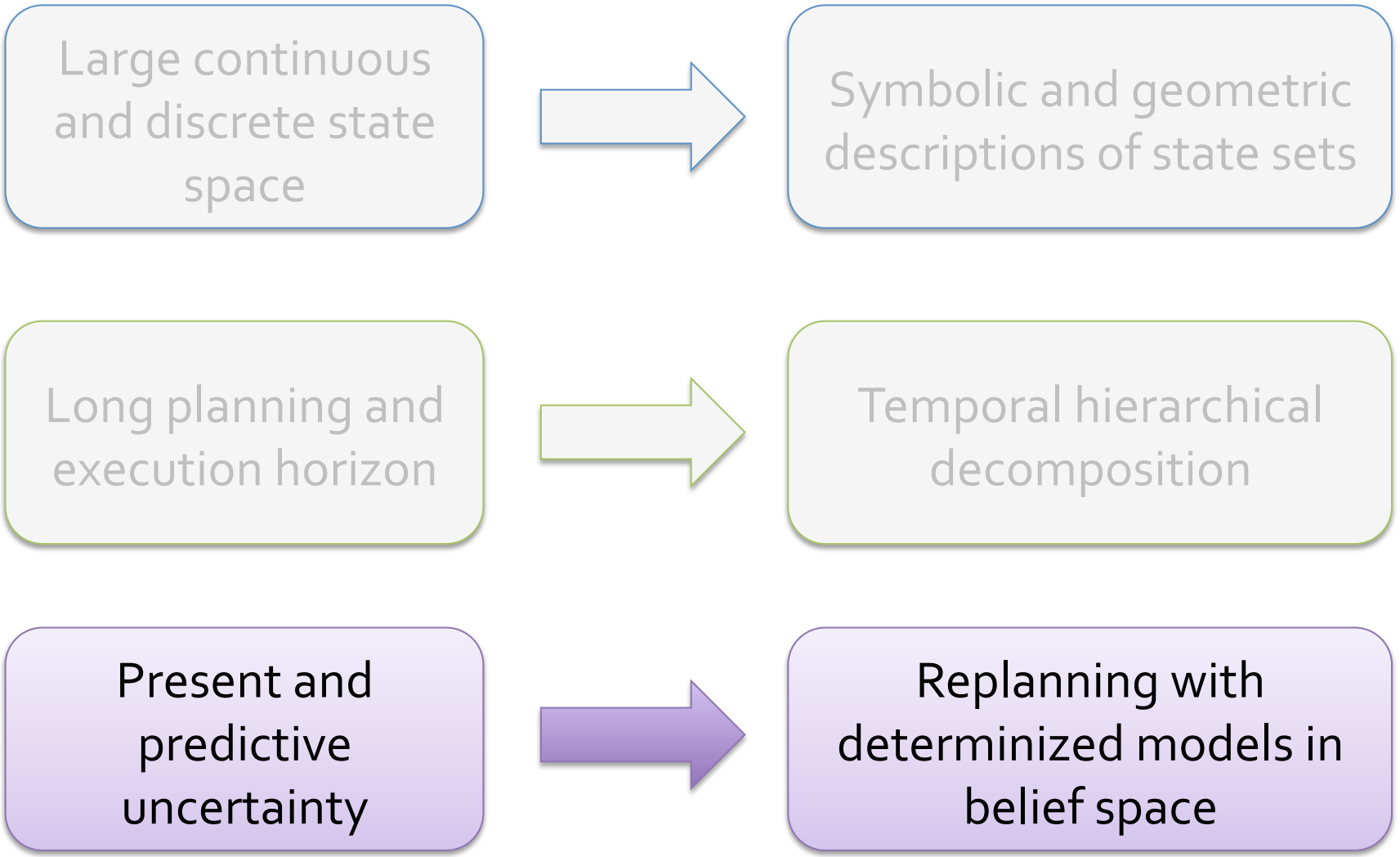
- have to vacuum before mopping
- have to put away junk items before vacuuming



Cleaning house



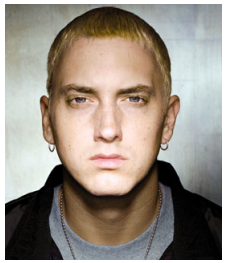
Addressing real world challenges



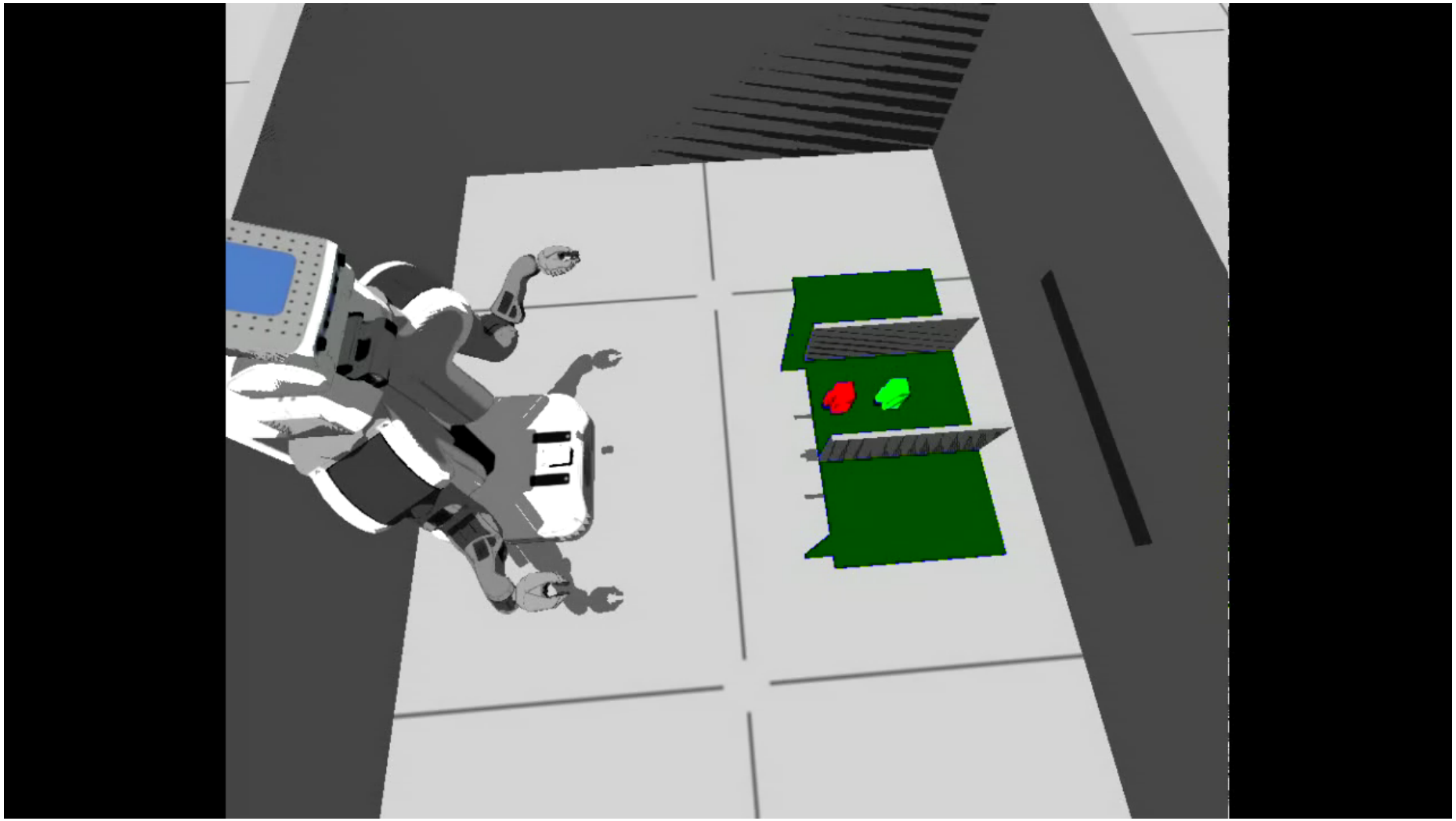
Mens et Manus

M and M

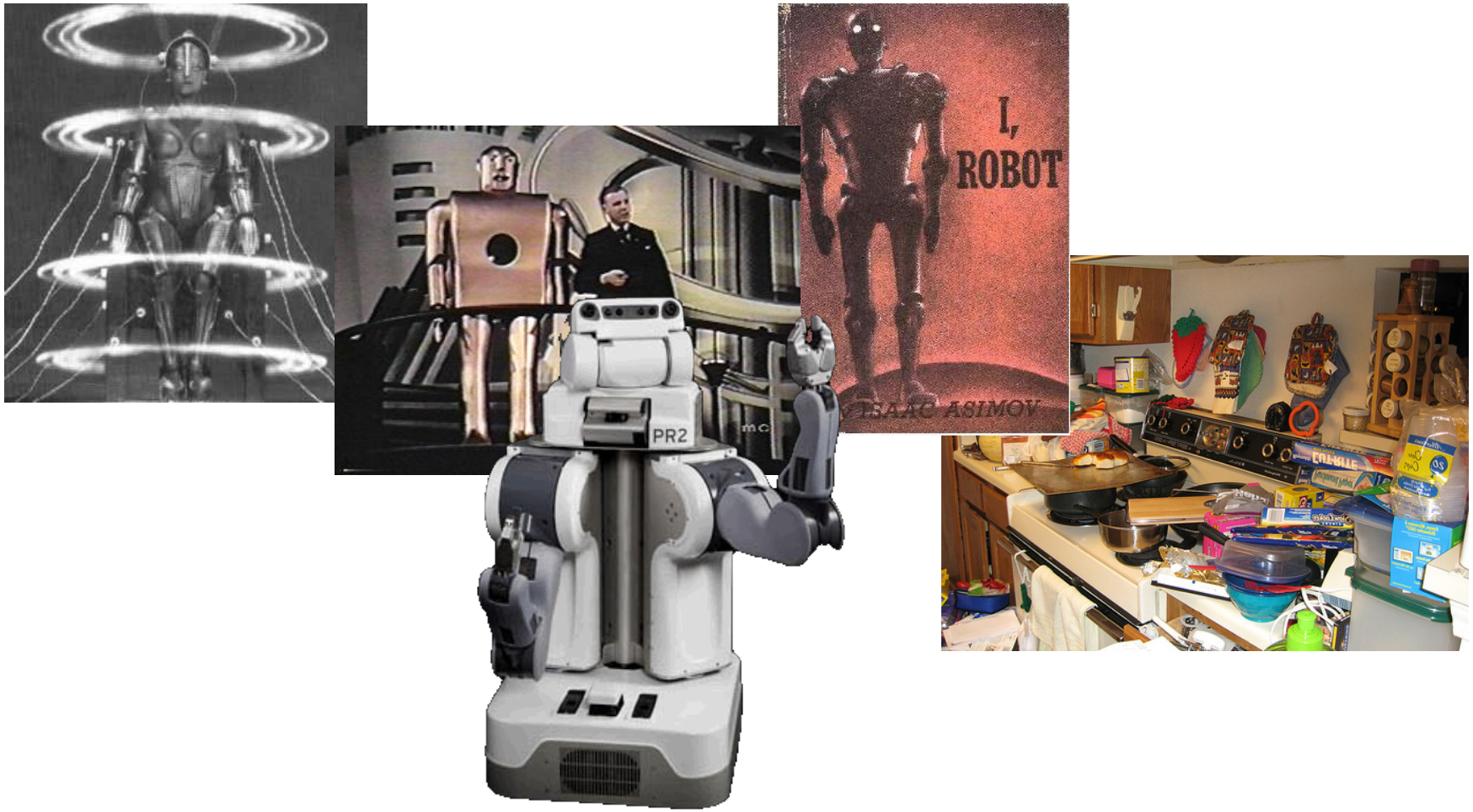
Eminem



M and M dreams of swapping two cups



M and M video


I,
ROBOT
ISAAC ASIMOV
PR2