Hierarchical task and motion planning

Tomas Lozano-Perez
Leslie Pack Kaelbling
MIT CSAIL
Real robot meets real world

WTF? ¹

¹ What To do First?
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

- Large continuous and discrete state space
- Long planning and execution horizon
- Present and predictive uncertainty

→

- Symbolic and geometric descriptions of state sets
- Temporal hierarchical decomposition
- Replanning with determinized models in belief space
Some related approaches

- Alami et al
- Plaku Hager
- Wolfe et al

Topics:
- Motion planning
- Symbolic representation
- MDP, POMDP
- Uncertainty
- Hierarchy
- Movable obstacles
- Stilman
Addressing real world challenges

- Large continuous and discrete state space
- Long planning and execution horizon
- Present and predictive uncertainty

Symbolic and geometric descriptions of state sets
- Temporal hierarchical decomposition
- Replanning with determinized models in belief space
Symbolic and geometric descriptions of state sets

Symbols for non-geometric properties

- are the dishes clean? \( \text{Clean}(\text{dish}24) \)
- is the stove on? \( \text{On}(\text{stove}) \)
- are the leftovers edible? \( \text{Edible}(\text{lasagna}) \)

Symbols for geometric and physical abstractions

- is the beer in the refrigerator? \( \text{In}(\text{beer}, \text{refrigerator}) \)
  - region containment
- is the robot holding the pot? \( \text{Holding}(\text{pot}) \)
  - 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

<table>
<thead>
<tr>
<th>Classical formulation</th>
<th>Real-world formulation</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Requires complete symbolic description of world state</td>
<td>• Symbolic description incomplete in infinite domain (regions, ...)</td>
</tr>
<tr>
<td>• Requires enumeration of all possible values of variables</td>
<td>• Generators for values of existential variables</td>
</tr>
<tr>
<td>• No explicit reasoning: requires all results to be asserted</td>
<td>• Extra reasoning about consistency and entailment for geometric properties</td>
</tr>
</tbody>
</table>

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

\( tidy(\text{house}) \land charged(\text{robot}) \)
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)
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(0, Target):
exists: \( P \in \text{generatePlacePaths}(0, \text{Target}) \)

- 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: $\text{In}(a, \text{leftR})$

O: $\text{Place}(a, \text{leftR})$

G: **Holding**(a), $\text{ClearX}(\text{Swept}(\text{PlacePath}(a, \text{leftR})))$
Regression and generation

Goal: $\text{In}(a, \text{leftR})$

$O$: $\text{Place}(a, \text{leftR})$

$G$: **Holding**($a$), $\text{ClearX}(\text{Swept}(\text{PlacePath}(a, \text{leftR})))$

$O$: $\text{Pick}(a)$

$G$: **ClearX**($\text{Swept}(\text{PickPath}(a))$),

$\quad \text{ClearX}(\text{Swept}(\text{PlacePath}(a, \text{leftR})))$
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

- Large continuous and discrete state space → Symbolic and geometric descriptions of state sets
- Long planning and execution horizon → Temporal hierarchical decomposition
- Present and predictive uncertainty → Replanning with determinized models in belief space
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:
**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(0):
pre: ...
res: Holding(0)

Place(0, R):
pre: 0.Holding(0)
1.InRobot(room(R))
res: In(0, R)
Postponing preconditions creates hierarchy

Pick(0):
pre: ...
res: Holding(0)

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

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

Pick(0):
pre: ...
res: Holding(0)

Place0(0, R):
pre: Holding(0)
res: In(0, R)

Pick(a) → Place0(a, wash) → MoveTo(room(wash)) → Place(a, wash)
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

\[
\text{HPN}(\text{currentState}, \text{goal}, \text{operators}, \text{absLevel}, \text{world}):
\]

\[
\text{if } \text{holds}(\text{goal}, \text{currentState}):
\]

\[
\text{return } \text{TRUE}
\]

\[
\text{else } p = \text{PLAN}(\text{currentState}, \text{goal}, \text{operators}, \text{absLevel})
\]

\[
\text{for } (o_i, g_i) \text{ in } p
\]

\[
\text{if } \text{prim}(o_i):
\]

\[
\text{currentState} = \text{world.execute}(o_i)
\]

\[
\text{else } \text{HPN}(\text{currentState}, g_i, \text{operators}, \text{NEXTLEVEL}(\text{absLevel}, o_i), \text{world})
\]
Hierarchical pick and place

Pick \(\text{Holding}(a)\) Place \(\text{In}(a, \text{leftR})\)

Clear \(\text{Clear(\text{Sw(PickP}(a))))\) Pick \(\text{Holding}(a)\)

Pick \(\text{Holding}(b)\) Place \(\text{In}(b, \text{Parking}(b))\)

\(\text{Swept(Pick}(a))\)

\(\text{leftR}\)

\(\text{Park}(b)\)

\(a\)

\(b\)
Hierarchical pick and place: too abstract!!

In(a, leftR)
Place
Holding(a)
Pick
Holding(a)
Pick
Clear(Sw(PickP(a)))
Clear
Pick
Holding(a)
Place
In(b, Parking(b))
Pick
Holding(b)
Place
In(b, Parking(b))

Swept(Pick(a))
Decrease abstraction level

- Clear(Sw(PickP(a)))
- Clear(Sw(PlaceP(a)))
- Swept(Place(a))
- Decrease abstraction level
- Park(b)
- In(b, Parking(b))
- Clear(Sw(PlaceP(a)))
- In(a, leftR)
- Clear(Sw(PickP(a)))
- Swept(Pick(a))
- Swept(Place(a))
- Park(b)
- b
- a
- leftR
Decrease abstraction level:
Decompose abstraction level:

- **Pick**
- **Holding(a)**
- **Place**
- **In(a, leftR)**

### Activities

- **Clear(Sw(PickP(a)))**
- **Clear(Sw(PlaceP(a)))**

### Inclusions

- **In(b, Parking(b))**
- **Clear(Sw(PickP(a)))**
- **Clear(Sw(PlaceP(a)))**

### Swept

- **Swept(Pick(a))**
- **Swept(Place(a))**
Hierarchical pick and place: consider clearX

- Swept(Pick(a))
- Swept(Place(a))
- Park(b)

Pick

Holding(a) Clear(Sw(PlaceP(a)))

Clear(Sw(PickP(a))) Clear(Sw(PlaceP(a)))

Clear(Sw(PickP(a))) Clear(Sw(PlaceP(a)))

Holding(a) Clear(Sw(PlaceP(a)))

In(a, leftR)

In(b, Parking(b)) Clear(Sw(PickP(a))) Clear(Sw(PlaceP(a)))

Pick

Clear

Place

leftR
Correctness

If

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

then executing $\texttt{HPN(start, G, ops, H, w)}$ 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

- Large continuous and discrete state space
- Symbolic and geometric descriptions of state sets
- Long planning and execution horizon
- Temporal hierarchical decomposition
- Present and predictive uncertainty
- Replanning with determinized models in belief space
Mens et Manus

M and M

Eminem
M and M dreams of swapping two cups
M and M video