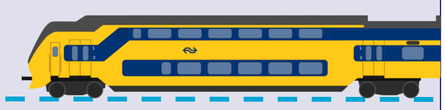
Discovering Generalized Landmarks: Patterns in Abstract States of Real-World Problems

November 2024

i.k.hanou@tudelft.nl

Issa Hanou, Sebastijan Dumančić, Mathijs de Weerdt



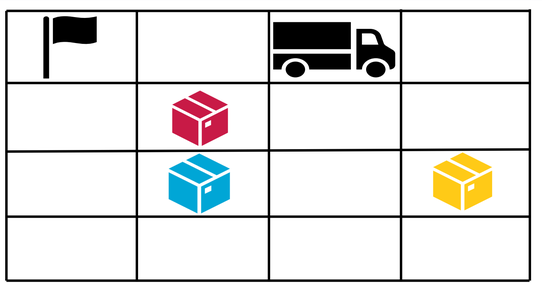
PROBLEM

- **AI Planning: Traditional landmarks (atoms)**
- Limitations
  - Recomputed for every instance
  - Disjunctive landmarks
  - Problem symmetries

Example: Delivery problem

Predicates:
- at(Object,Cell)
- carrying(Truck,Package)
- empty(Truck)
- adjacent(Cell1,Cell2)

Actions:
- pick-package(Truck,Package,Cell)
- drop-package(Truck,Package,Cell)
- move(Truck,Cell1,Cell2)



Traditional landmarks:
- at(truck1,cell-1-1)
- carrying(truck1,packageRed)

LANDMARKS

Generalized landmarks

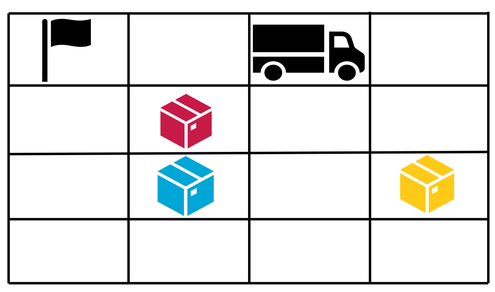
- Landmark is a formula over features as state indicator
- Generalize across instance
- Generalize across objects
- Graph with ordered landmarks
- Conditional looped edges

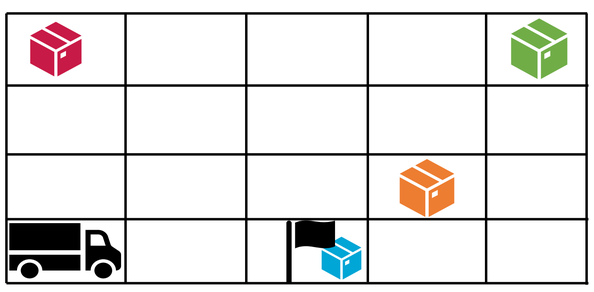
Traditional: at(truck1, cell-1-1)

Generalized: truck at a cell with a package

Traditional: carrying(truck1, package1) → at(package1, c-0-0)

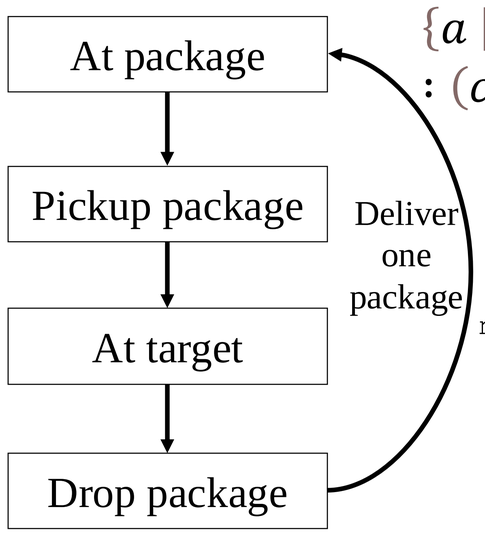
Generalized: carrying a package and not at a goal cell → at a goal cell





DISCOVERY

1. **Find set of training plans**
   - **Create state trajectories**
2. **Generate features and filter to useful ones**
3. **Feature valuations in all training states**
4. **Extract set of feature for next landmark**
5. **Detect loops to previous landmarks**
6. **Create graph**



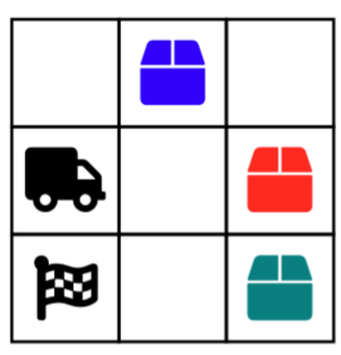
$$\{a \mid \exists b : (a,b) \in [at(a,b)] \sqcap b \in \{c \mid \forall d : (c,d) \in [at(d,c)] \rightarrow e \in [truck(e)]\}\}$$

Truck is empty and there is a package at this cell

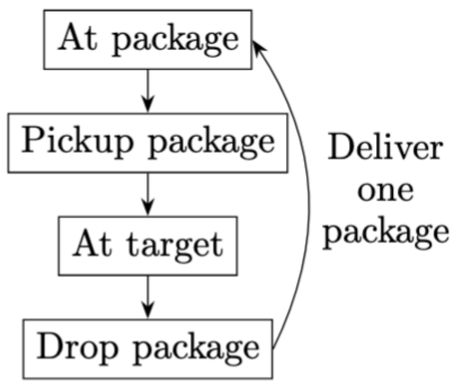
```
n_count(c_some(r_primitive(at_g,0,1),
  c_some(r_inverse(r_primitive(at,0,1)),
  c_primitive(truck,0))))
```

ISLAND PLANNING

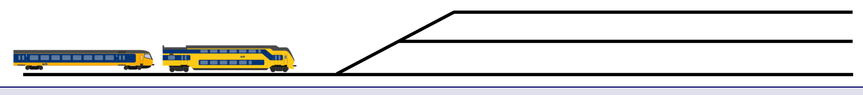
- **Plan between landmarks**
- **Update state**
- **Check for loops**
- **Planner independent**
  - **Example ASP: increasing horizon**
- **Next: Railway application**
  - **Park train 1 at a different track than train 2**



(a) Instance with 3 packages.



(b) Generalized landmarks.



Acknowledgement
This work is part of the NWO LTP-ROBUST RAIL Lab, a collaboration between the Delft University of Technology, Utrecht University, NS and ProRail.

TU Delft Delft University of Technology

Universiteit Utrecht

ProRail