

Incremental Search for Counterexample-Guided Cartesian Abstraction Refinement

Jendrik Seipp Samuel von Allmen Malte Helmert
University of Basel

In a Nutshell

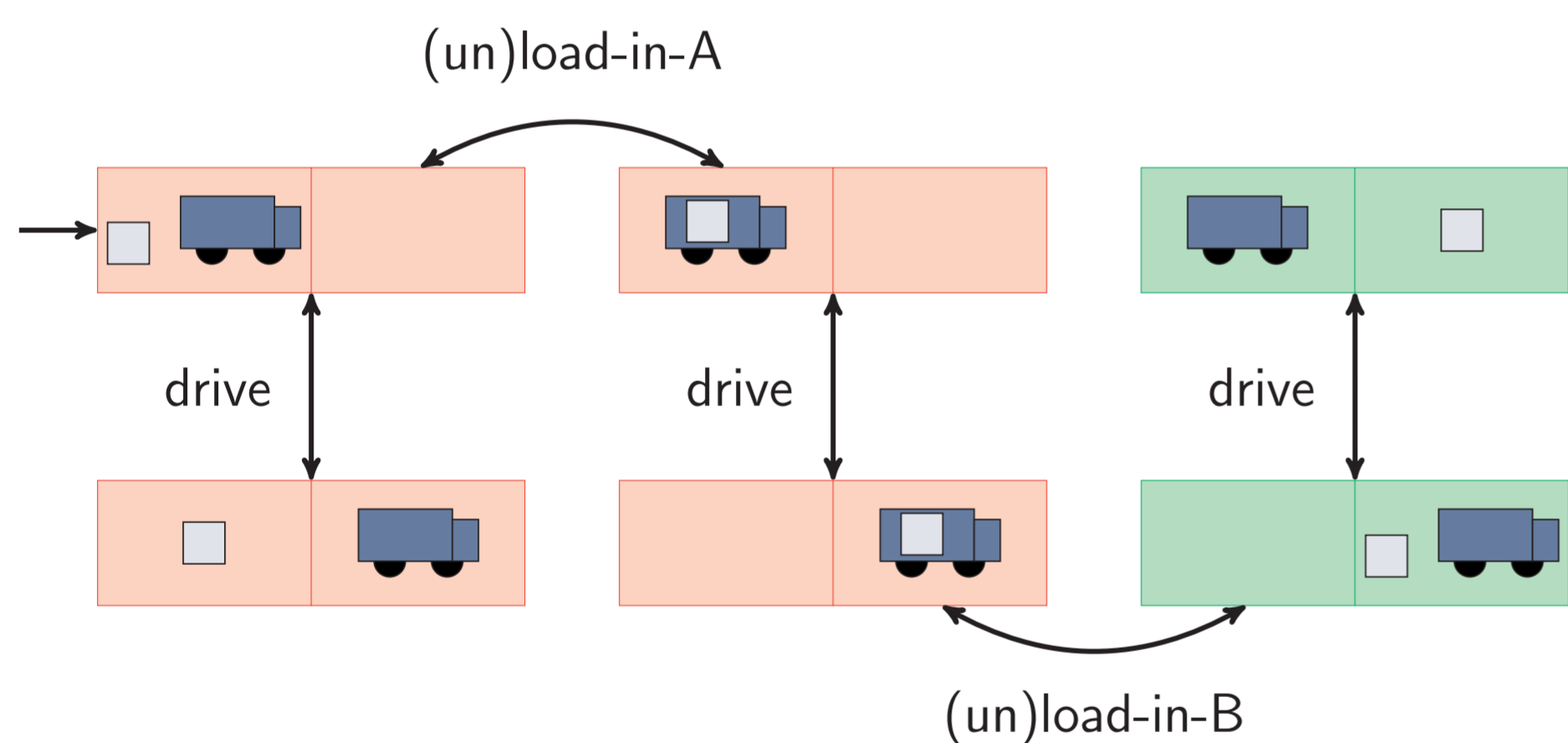
- ▶ optimal classical planning
- ▶ A* search + abstraction heuristic
- ▶ counterexample-guided Cartesian abstraction refinement
- ▶ bottleneck: find shortest path
- ▶ incremental search: 1000x speedup

CEGAR

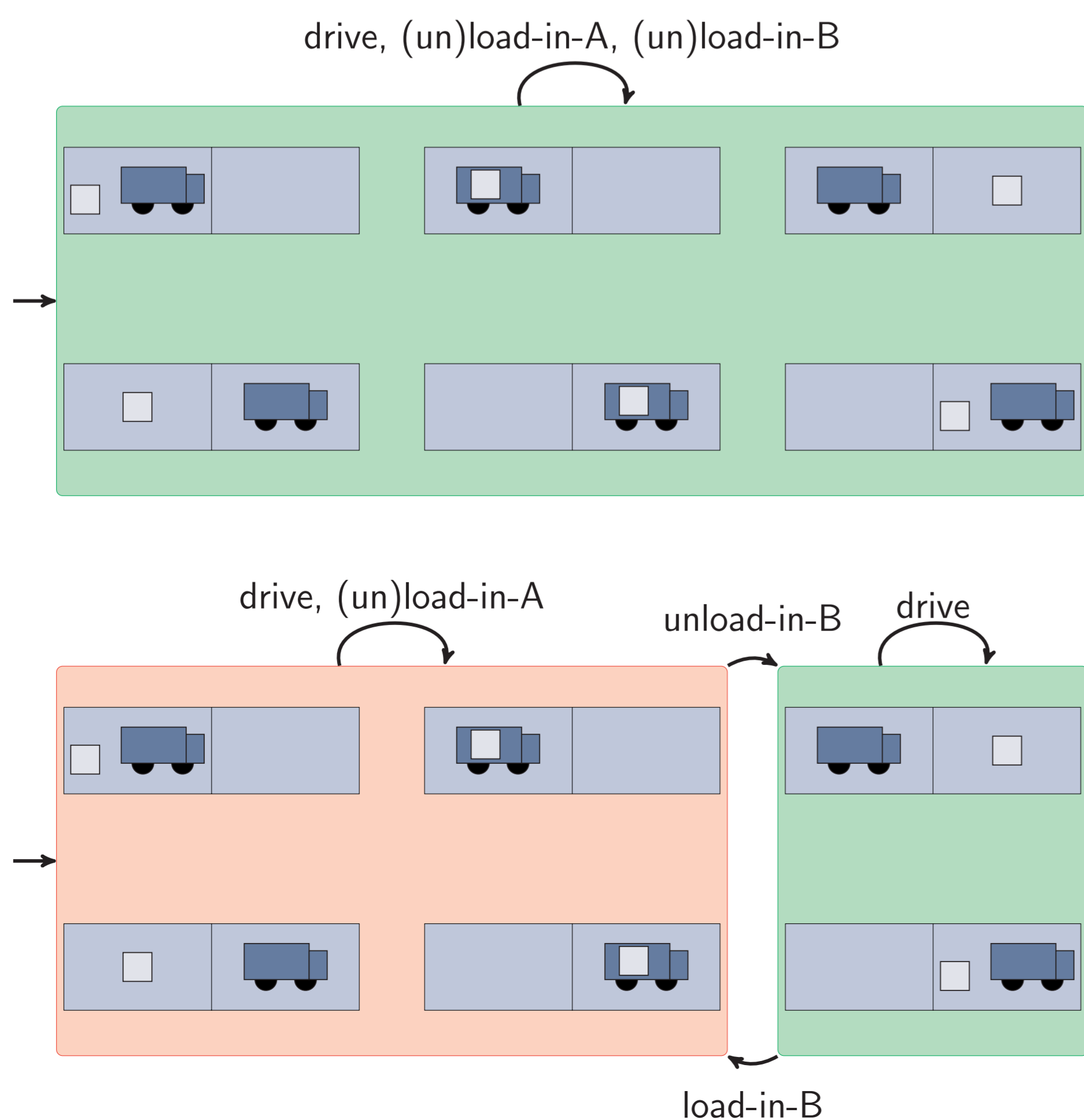
```

compute initial abstraction
until TERMINATE():
  find shortest path in abstraction
  if there is no path:
    return unsolvable
  find flaw in path
  if there is no flaw:
    return plan
  refine abstraction for flaw
return abstraction
    
```

Example Task



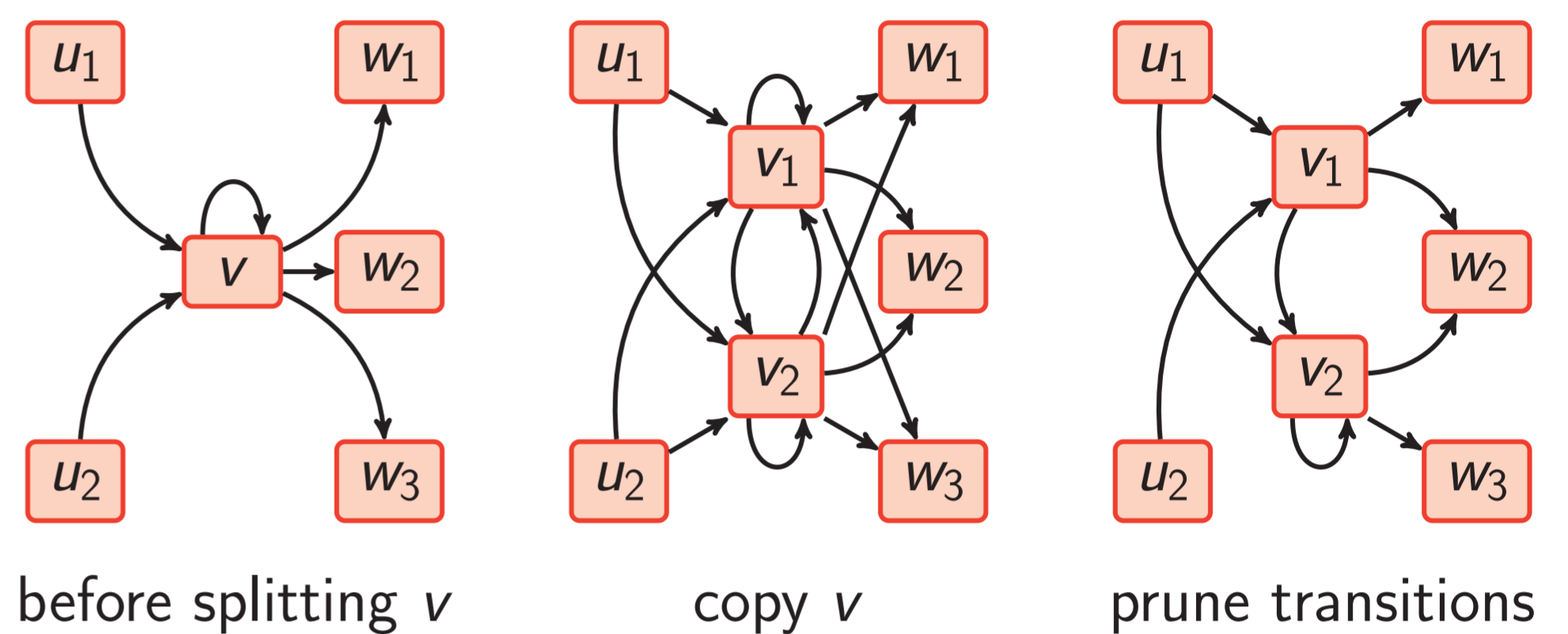
Abstraction Refinement



Incremental Search

- ▶ add/remove transitions
- ▶ increase/decrease weights
- ▶ fixed set of states

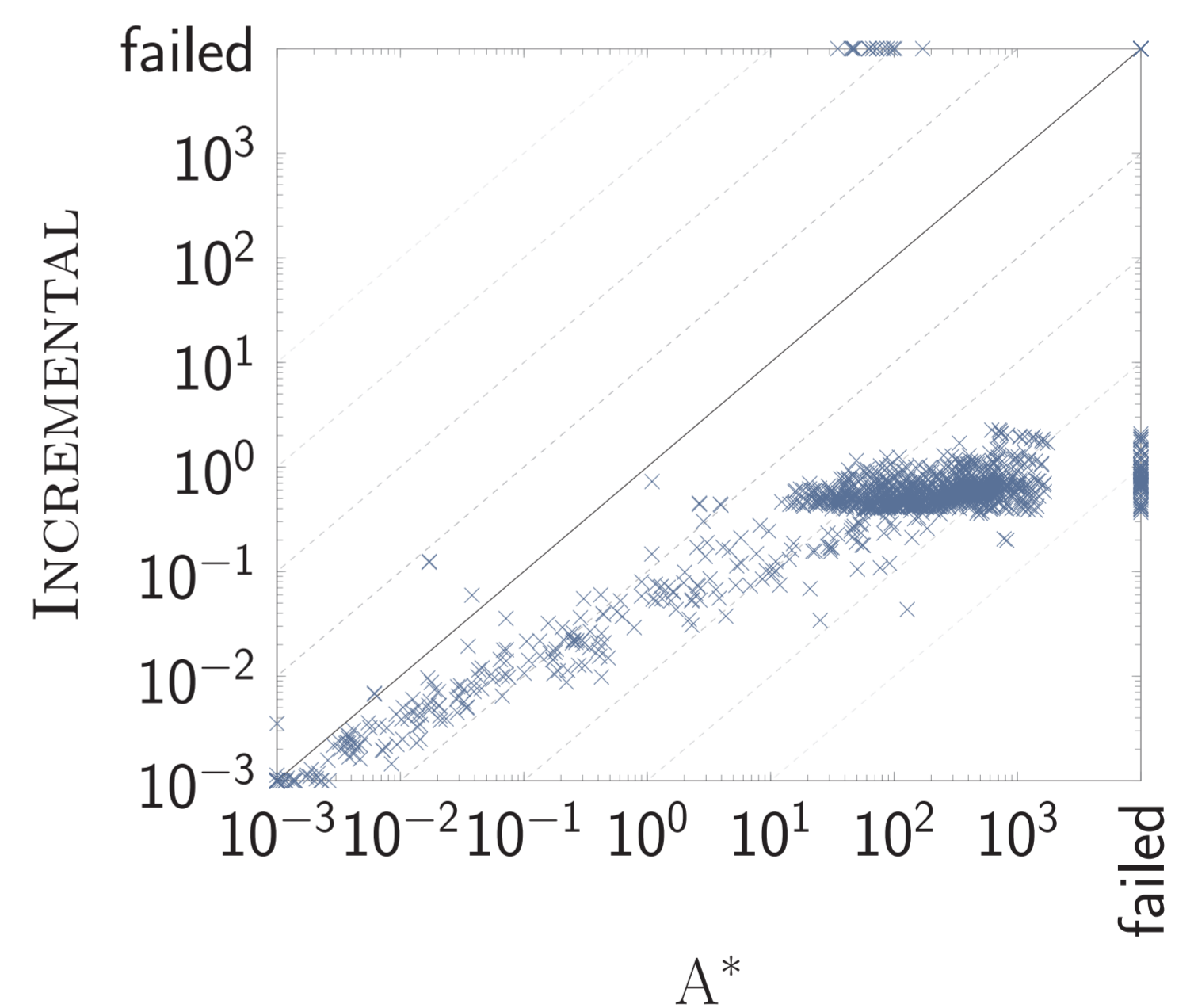
Two-Step Refinement



Increase (Frigioni et al., 2000)

- ▶ increasing weights, removing transitions
- ▶ shortest path tree
- ▶ reconnect ancestor states, mark rest dirty
- ▶ run Dijkstra on dirty states

Time for Finding Shortest Paths



Solved Tasks Over Time

