Yu-qian JIANG, Shi-qi ZHANG, Piyush KHANDELWAL, Peter STONE, 2019. Task planning in robotics: an empirical comparison of PDDL- and ASP-based systems. *Frontiers of Information Technology & Electronic Engineering*, 20(3):363-373. https://doi.org/10.1631/FITEE.1800514

Task planning in robotics: an empirical comparison of PDDL- and ASP-based systems

Key words: Task planning; Robotics; Planning domain description language (PDDL); Answer set programming (ASP)

Corresponding author: Shi-qi ZHANG
E-mail: szhang@cs.binghamton.edu
ORCID: Yu-qian JIANG, http://orcid.org/0000-0003-0091-6871;
Shi-qi ZHANG, http://orcid.org/0000-0003-4110-8213

Motivations

1. Robots need task planning algorithms to sequence actions toward accomplishing goals that are impossible through individual actions. Off-the-shelf task planners can be used by intelligent robotics practitioners to solve a variety of planning problems.

2. However, many different planners exist, each with different strengths and weaknesses, and there are no general rules for which planner would be best to apply to a given problem.

Problems

1. The goal of this article is to help a robot planning practitioner understand the effects of specific domain properties to aid the choice of language selection for general-purpose planning.

2. We compare two declarative languages: the planning domain definition language (PDDL), the most popular language in the planning community, and answer set programming (ASP), a popular general knowledge representation and reasoning (KRR) language that has been recently used in a variety of task planning problems, including robotics.

Methods

1. We used three benchmark domains to compare the performance of the PDDL planner fast downward with the ASP solver Clingo.

2. The benchmark problems consist of the blocks world and hiking problems from the International Planning Competition (IPC), as well as a variant of the robot navigation problem. Properties of planning problems, such as length of optimal plan and number of objects in the domain, are varied to analyze their effects.

Method

Although planner performance can be sensitive to domain encoding, we take care, to the extent possible, to encode the domains similarly in each language. For example, here are the PDDL and ASP encodings of the opendoor action from the robot navigation domain.

PDDL

```
(:action opendoor
 :parameters (?d - door)
 :precondition (canopen ?d)
 :effect (and (open ?d)
          (forall (?d1 - door)
               (not (canopen ?d1))))))
```

ASP

```
:- opendoor(D,n), not canopen(D,n-1).
open(D,n) :- opendoor(D,n).
```

Main results

PDDL-based planners perform better on problems with longer solutions, and ASP-based planners are better on tasks with a large number of objects or tasks in which complex reasoning is required to reason about action preconditions and effects.

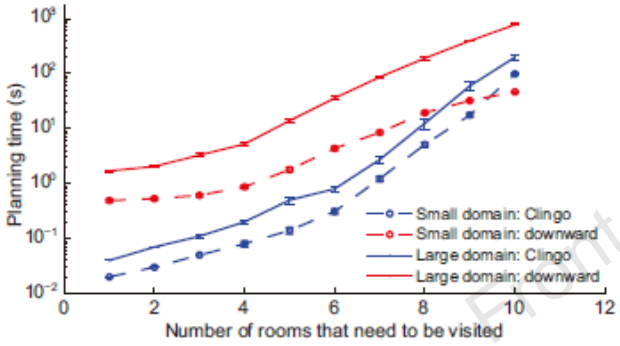

Fig. 1 Robot navigation: small domain, 10 rooms; large domain, 15 rooms. Forty percent of rooms are connected via doors, and the rest are directly accessible from the corridor

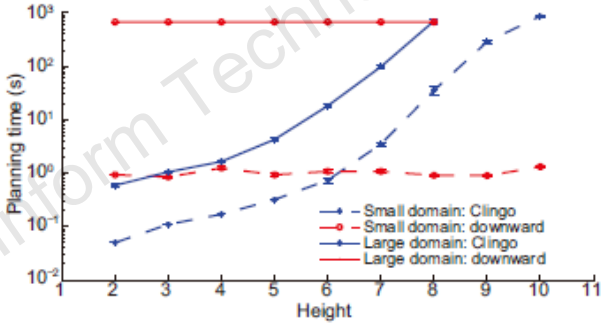

Fig. 2 Blocks world: small domain, 15 blocks; large domain, 60 blocks. We use a timeout of 1800 s (same as IPC). The graph plots only configurations where all trials of both planners finished before timeout (same for all following experiments). References to color refer to the online version of this figure

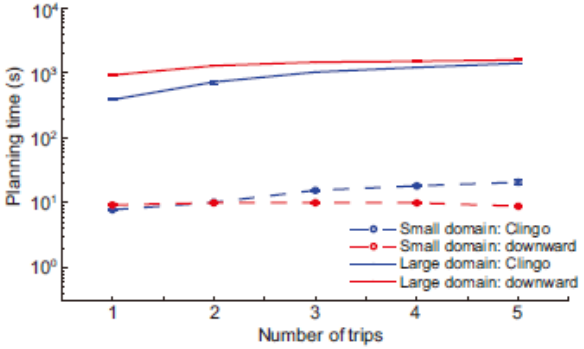

Fig. 5 Hiking: small domain vs. large domain. References to color refer to the online version of this figure

Conclusions

1. We empirically compared ASP- and PDDL-based task planners using three benchmark domains. The evaluations demonstrate that:

   (1) PDDL-based planners perform better when tasks require long solutions;

   (2) However, ASP-based task planners are less susceptible to an increase in the number of domain objects, as long as the number of objects does not explode the number of grounded actions.

   (3) Finally, in domains requiring complex reasoning such as the robot navigation domain, ASP-based planners can be considerably faster than PDDL-based planners for shorter plans.

2. Such observations can serve as a useful reference to task planning practitioners in the process of action language selection.