Classical Planning as QBF without Grounding

Irfansha Shaik, Jaco van de Pol

{irfansha.shaik, jaco}@cs.au.dk

Introduction

- Classical planning is finding a plan from the Initial state to some Goal state.
- SAT based planning is based on encoding bounded reachability of a goal state.
- Translating PDDL representations to propositional logic requires grounding all parameters.
- SAT encodings can be exponentially large for hard to ground domains such as Organic Synthesis.
- Current QBF based planning encodings compress path length, however grounding is still a bottleneck.

Encoding overview and Key Idea

$$\exists\, \mathsf{A}^0, \mathsf{PM}^0, \ldots, \mathsf{A}^{k-1}, \mathsf{PM}^{k-1}$$
$$\forall\, \mathsf{OC}$$
$$\exists\, \mathsf{P}^0, \ldots, \mathsf{P}^k$$
$$\mathsf{I_u}(\mathsf{P}^0, \mathsf{OC}) \wedge \mathsf{G_u}(\mathsf{P}^k, \mathsf{OC}) \wedge \bigwedge_{i=0}^{k-1} \mathsf{T_u}^i(\mathsf{P}^i, \mathsf{P}^{i+1}, \mathsf{OC}, \mathsf{A}^i, \mathsf{PM}^i)$$

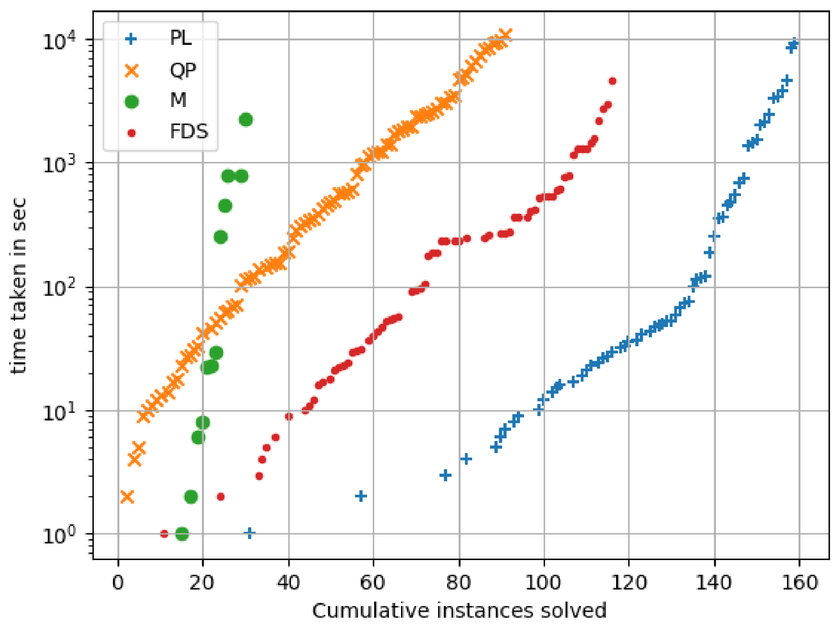
Figure 1: Ungrounded QBF Encoding

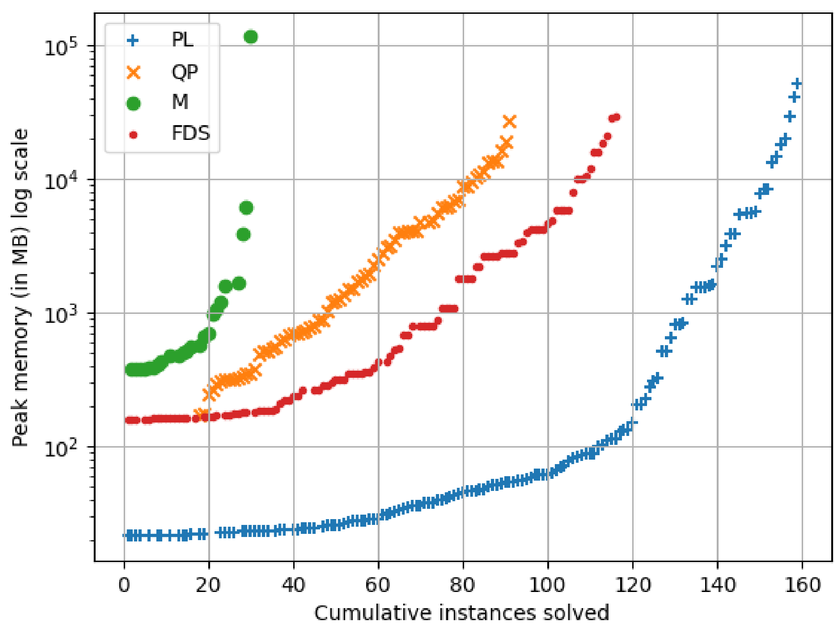
Key Idea

- Representing action and predicate parameters in PDDL specification using universal variables.
- When the universal variables (OC) are expanded, the resulting encoding is equivalent to a SAT encoding.
- In plain words, there exists a plan for all object-combinations there exists predicates for each step such that initial, goal and transition constraints hold.

Results

Comparing our Q-Planner with Madagascar, Fast Downward Stone Soup and Power lifted.





Contribution

- We present a compact QBF encoding which grows logarithmically with objects.
- We provide an open source implementation which takes PDDL specifications and generates QBF encodings.
- Our planner can solve many Organic Synthesis instances, which could not be handled before by any SAT/QBF techniques.

AARHUS UNIVERSITY