Unsolvability Certificates for Classical Planning

Salomé Eriksson  Gabriele Röger  Malte Helmert

University of Basel, Switzerland

ICAPS 2017
Validating correctness of planner output:

- **Why?**
  - Software bugs, hardware faults, malicious reasons . . .

- **How?**
  1. Planner outputs a plan: VAL/INVAL
  2. Planner claims unsolvability: ?
Proving Unsolvability

**Goal**

Generate *unsolvability certificate* which can be verified

Desired properties:

- Soundness & Completeness
- Efficient generation
- Efficient verification
- Generality
General Idea

Unsolvable planning problems $\leadsto$ No path from $I$ to goal
General Idea

Unsolvable planning problems \(\rightsquigarrow\) No path from \(I\) to goal

Split graph into \(S_I\) and \(S_G(=\overline{S_I})\) s.t. no outgoing edges from \(S_I\)
General Idea

Unsolvable planning problems $\leadsto$ No path from $I$ to goal

Split graph into $S_I$ and $S_G(=\overline{S_I})$ s.t. no outgoing edges from $S_I$
Inductivity

**Inductive Set**

A state set $S$ is inductive if for all $s \in S$, all operator applications lead to a $s' \in S$.

**Inductive Certificate**

If a state set

1. contains the initial state
2. contains no goal state
3. is inductive

the planning task is unsolvable.
Representation of State Sets

Representation of state sets as **logical formulas**

We focus on the following representations:

- (RO)BDD
- 2CNF
- Horn Formulas
Motivation

Inductive Certificates

Certifying Planning Algorithms

Conclusion

Conjunctive/Disjunctive Certificates

Not all state sets compactly representable

\[ S = \{S_1, \ldots, S_n\} \] is a
- conjunctive certificate: \( \bigcap_{S_i \in S} S_i \) is inductive certificate
- disjunctive certificate: \( \bigcup_{S_i \in S} S_i \) is inductive certificate

Efficient Verification? in general not feasible
\( \Rightarrow \) only consider up to \( r \) sets at once
## Suitable Representations

<table>
<thead>
<tr>
<th>Certificate Type</th>
<th>BDD</th>
<th>2CNF</th>
<th>Horn Formulas</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inductive Certificate</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Conjunctive Certificate</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>r-conjunctive Certificate</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
<tr>
<td>Disjunctive Certificate</td>
<td>No</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>r-disjunctive Certificate</td>
<td>Yes</td>
<td>No</td>
<td>No</td>
</tr>
<tr>
<td>1-disjunctive Certificate</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Blind Search

Progression: expanded = reachable from \( I \) \( \Rightarrow \) inductive certificate

Regression: expanded = backwards-reachable from goal \( \Rightarrow \) complement is inductive certificate

Bidirection: whichever direction shows unsolvability

Suitable representation: BDDs (for symbolic search)
Progression: expanded = reachable from $I$
\[\leadsto\text{inductive certificate}\]
Progression: expanded = reachable from $I$
  ~ inductive certificate

Regression: expanded = backwards-reachable from goal
**Blind Search**

- Progression: expanded $=$ reachable from $I$
  $\leadsto$ inductive certificate

- Regression: expanded $=$ backwards-reachable from goal
  $\leadsto$ complement is inductive certificate

$S_{\text{reg}}$
Blind Search

- Progression: expanded = reachable from $I$
  $\leadsto$ inductive certificate
- Regression: expanded = backwards-reachable from goal
  $\leadsto$ complement is inductive certificate
- Bidirection: whichever direction shows unsolvability
Motivation

Inductive Certificates

Certifying Planning Algorithms

Conclusion

Blind Search

- Progression: expanded = reachable from $I$  
  $\leadsto$ inductive certificate

- Regression: expanded = backwards-reachable from goal  
  $\leadsto$ complement is inductive certificate

- Bidirection: whichever direction shows unsolvability

Suitable representation: BDDs (for symbolic search)
Union of states $s$ where $h^{M&S}(s) = \infty$ is inductive & no goal states

$\Rightarrow$ If $h^{M&S}(I) = \infty$, this union is inductive certificate

For linear merge strategies:

1. Represent cascading tables as ADD
2. Compress to BDD: finite $h$-values lead to $\bot$, infinite to $\top$

\[
\begin{array}{c|cc}
& v_1 & \\
v_2 & 0 & 1 \\
0 & 0 & 1 \\
1 & 1 & 2 \\
v_3 & & \\
0 & 0 & 1 \\
1 & \infty & 1 \\
2 & 2 & 0 \\
v_{1,2} & & \\
0 & & \\
1 & \infty & 1 \\
2 & & \\
\end{array}
\]

Cascading tables

\[
\begin{array}{c}
\text{ADD} \\
\text{BDD}
\end{array}
\]
Delete Relaxation Heuristics

\[ h^+(s) = \infty \] if part of the goal is relaxed unreachable

- \( U^+(s) \): relaxed unreachable variables
- \( \varphi_{U^+(s)} = \bigwedge_{v \in U^+(s)} \neg v \) is inductive & no goal states

\[ \therefore \text{If } h^+(I) = \infty, \varphi_{U^+(s)} \text{ represents inductive certificate} \]

Covers all delete-relaxation heuristics (\( h^{\text{max}}, h^{\text{add}}, h^{\text{FF}}, h^{\text{LM-Cut}}, \ldots \))

Suitable representation: BDDs, Horn Formulas, 2CNF
Similar idea to $h^+$, but with unreachable conjunctions:

$$\bigwedge_{c \in U^m(I)} \bigvee_{v \in c} \neg v$$

Suitable representation: Horn Formulas, 2CNF (for $m \leq 2$), BDDs (as 1-conjunctive Certificate)
Heuristic Search

Heuristic certificates sufficient if $h(I) = \infty$

General heuristic search:

- $S_{\text{exp}} = \{\{s\} | s \in \text{expanded states}\}$
- $S_{\infty}$: family of inductive sets covering all detected dead ends

$\leadsto S_{\text{exp}} \cup S_{\infty}$ is 1-disjunctive certificate

Suitable representation: BDDs, Horn Formulas, 2CNF

Limitation:

- all sets must have same representation
- sets cannot be conjunctive/disjunctive
Trapper [Lipovetzky et al. 2016]:
- only considers states not violating mutexes $M$ (based on $h^2$)
- no escape from $\varphi_{\text{trap}} \leadsto$ inductive
- no goal states (in considered states)

Observations:
- $\varphi_{\text{trap}}$ alone no certificate (goal states)
- states not violating mutexes ($= \varphi_{\neg M}$) inductive

$\leadsto \varphi_{\text{trap}} \land \varphi_{\neg M}$ represents inductive certificate (even 1-disjunctive)

Suitable representation: 2CNF, Horn Formulas
Experiments

Proof of concept implementation of

- $\text{FD}^{\text{cert}}$: generates BDD certificates for $A^* + \frac{h_{\text{max}}}{h_{\text{M&S}}}$
- $\text{Verifier}$: vanilla, $r$-conjunctive, $r$-disjunctive BDD certificates

limits: 30 min generation, 4 hours verification

<table>
<thead>
<tr>
<th>Coverage (702)</th>
<th>$h_{\text{max}}$</th>
<th>Ver.</th>
<th>$h_{\text{M&amp;S}}$</th>
<th>Ver.</th>
</tr>
</thead>
<tbody>
<tr>
<td>FD</td>
<td>212</td>
<td>136</td>
<td>123</td>
<td></td>
</tr>
<tr>
<td>$\text{FD}^{\text{cert}}$</td>
<td>223</td>
<td></td>
<td>191</td>
<td>155</td>
</tr>
</tbody>
</table>

all certificates valid
unsolvability certificates based on inductive sets

- completeness: yes
- efficient generation: yes/no
- efficient verification: mostly yes (if efficient generation)
- generality: yes/no

Future Work

- cover more techniques (heuristics, pruning, ...)
- combined certificate with different formalisms