Certified Unsolvability for SAT Planning with Property Directed Reachability

Salomé Eriksson Malte Helmert

University of Basel, Switzerland

ICAPS 2020

Certifying Algorithms

Certifying Algorithm

Emit *certificate* alongside answer, verify *independently*.

in planning:

- solvable: plan
- unsolvable: unsolvability certificate, e.g. [E et al. 2018]

Desired Certificate Properties

- sound & complete
- efficient generation → polynomial in planner runtime
- efficient verification → polynomial in certificate size
- general

Covered So Far

- explicit & symbolic search
- different heuristics
- h^2 preprocessing
- Trapper

SAT-based planning?

- traditionally less suited for detecting unsolvability
- verifying properties of CNF formulas **NP**-complete

Property Directed Reachability [Suda 2014]

reasons about layers L_i:

- overapproximates states with distance $\leq i$ to goal
- iterative refinement
- represented as
  - CNF $\rightarrow$ requires SAT solver
  - dual-Horn (for STRIPS tasks)

$L_u = L_{u-1} \rightarrow$ unsolvable

Unsolvability Proof System [E et al. 2018]

collection of knowledge about *sets of states*

- subset relations
- deadness of state sets

$\{I\}$ or G dead $\rightarrow$ task unsolvable

gaining & verifying knowledge:

- basic statements $A \subseteq B$
  $\rightarrow$ need to be verified *semantically*
- inference rules $A \subseteq B$ and B dead $\rightarrow$ A dead
  $\rightarrow$ need to be verified *syntactically*

PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification

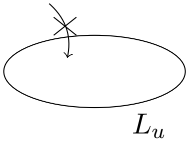
PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification
(1)	$[A]L_u \subseteq L_u$	basic statement



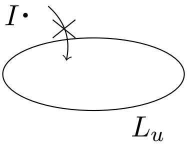
PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification
(1)	$[A]L_u \subseteq L_u$	basic statement
(2)	$\{I\} \subseteq \overline{L_u}$	basic statement



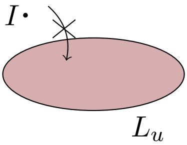
PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification
(1)	$[A]L_u \subseteq L_u$	basic statement
(2)	$\{I\} \subseteq \overline{L_u}$	basic statement
(3)	L_u is dead	from (1) and (2) with rule **RI**



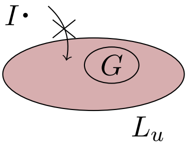
PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification
(1)	$[A]L_u \subseteq L_u$	basic statement
(2)	$\{I\} \subseteq \overline{L_u}$	basic statement
(3)	L_u is dead	from (1) and (2) with rule **RI**
(4)	$G \subseteq L_u$	basic statement



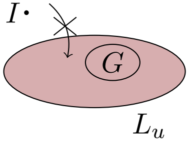
PDR Unsolvability Certificate

PDR Argument

$L_u = L_{u-1} \rightarrow$ unsolvable

certificate translation:

#	statement	justification
(1)	$[A]L_u \subseteq L_u$	basic statement
(2)	$\{I\} \subseteq \overline{L_u}$	basic statement
(3)	L_u is dead	from (1) and (2) with rule **RI**
(4)	$G \subseteq L_u$	basic statement
(5)	G is dead	from (3) and (4) with rule **SD**



Efficient Verification

bottleneck: basic statements $(A \subseteq B)$
$\rightarrow$ depends on representation of A and B

efficient for

- BDDs
- (dual-)Horn formulas
- 2CNF
- explicit enumeration

Not efficient for CNF!

Verifying PDR for positive STRIPS

implemented on top of pdrplan

	base	certifying	verifier
PDR	388	-4	-2
FD-$h^{\text{M\&S}}$	224	-27	-19
FD-h^{max}	203	-47	-14
DFS-CL	394	-8	-1

small generation overhead, efficient verification

Integration of SAT Certificates

Observations

- PDR must have solved related SAT queries already
- SAT solvers are certifying

$\rightarrow$ use SAT certificates from planner's SAT calls*

Example

given: state sets S_φ and S_ψ described by φ and ψ (in CNF)

$\rightarrow S_\varphi \subseteq \overline{S_\psi}$ verified with UNSAT certificate for $\varphi \land \psi$

*SAT calls don't perfectly match basic statements
$\rightarrow$ combine knowledge within proof system

Conclusion & Outlook

Contributions

- certifying version of PDR
- extension of proof system to CNF formalism

outlook:

- traditional SAT solvers with modern upper bound techniques
- problem reformulations (e.g. symmetry, STRIPS duality)
- ...