

Certified Unsolvability for SAT Planning with Property Directed Reachability

Salomé Eriksson Malte Helmert
University of Basel

Certifying Algorithms

Algorithm emits *certificate* alongside its output, which is verified independently:

	SAT solvers	Planners
solvable	satisfying assignment	plan
unsolvable	DRAT proof	unsolvability certificate

Desired Properties

- ▶ sound & complete
- ▶ efficient generation (polynomial in planner runtime)
- ▶ efficient verification (polynomial in certificate size)
- ▶ generality

Unsolvability Certificates for Planning [E et al. 2018]

The certificate incrementally builds a *knowledge base* of proven statements:

- ▶ **objects:** state sets S_i (represented by propositional logic formulas φ_i)
- ▶ **types of statements:**
 - ▶ $S_1 \subseteq S_2$
 - ▶ S_1 dead (no state in S_1 can be part of a plan)
- ▶ **basic statements:**
 - ▶ state facts about concrete objects
 - ▶ need to be verified *semantically*
- ▶ **inference rules:**
 - ▶ derive new knowledge from existing knowledge
 - ▶ universally true \rightarrow only need to be verified *syntactically*

A Task is proven unsolvable if $\{I\}$ or G have been proven to be dead.

Basic Statements Examples

B1	$\bigcap S_i \subseteq \bigcup S_j$	
B2	$(\bigcap S_i)[A] \subseteq \bigcup S_j$	$S[A] = \{s' \mid s \in S, s[a] = s' \text{ for some } a \in A\}$
B3	$[A](\bigcap S_i) \subseteq \bigcup S_j$	$[A]S = \{s \mid s' \in S, s[a] = s' \text{ for some } a \in A\}$

Inference Rules Examples

Rules for showing deadness:

SD	S_1 dead, $S_2 \subseteq S_1$	$\rightarrow S_2$ dead
PG	$S_1[A] \subseteq S_1 \cup S_2$, S_2 dead, $S_1 \cap G$ dead	$\rightarrow S_1$ dead
RI	$[A]S_1 \subseteq S_1 \cup S_2$, S_2 dead, $\{I\} \subseteq \overline{S_1}$	$\rightarrow S_1$ dead
ED		$\rightarrow \emptyset$ dead

Rules from Set Theory:

SI	$S_1 \subseteq S_2$, $S_1 \subseteq S_3$	$\rightarrow S_1 \subseteq S_2 \cap S_3$
ST	$S_1 \subseteq S_2$, $S_2 \subseteq S_3$	$\rightarrow S_1 \subseteq S_3$
UR		$\rightarrow S_1 \subseteq S_1 \cup S_2$

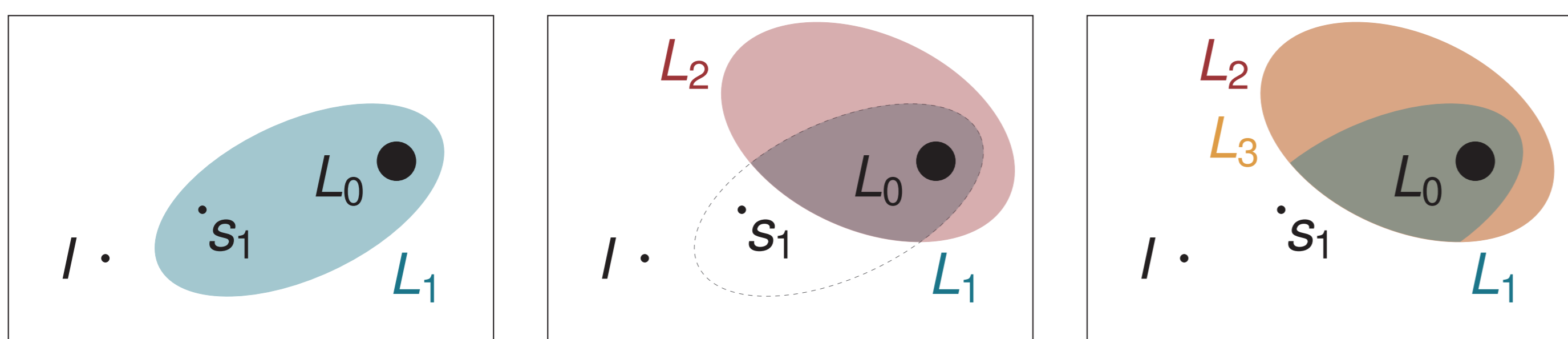
Property Directed Reachability [Suda 2014]

Property Directed Reachability (PDR) reasons about *layers* L_i which

- ▶ overapproximate states with distance $< i$ to goal,
- ▶ are iteratively refined, and
- ▶ are represented as CNF formulas, or dual-Horn formulas for STRIPS tasks.

```

for  $i = 0, \dots$  do
  while  $I \in L_i$  do
    if exists path of length  $i$  from  $I$  to  $G$  then
      return found plan
    else
      strengthen layers where path cannot be extended
    end
  end
  if  $L_u = L_{u-1}$  for some  $u < i$  then
    return unsolvable
  end
end
    
```

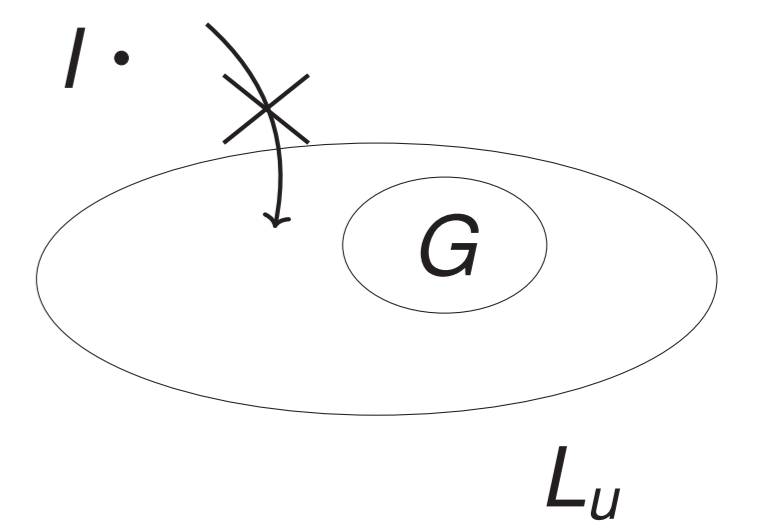


Certificate Structure

PDR's unsolvability argument:

- ▶ we cannot (backwards) reach new states from L_u
- ▶ L_u contains all goal states
- ▶ L_u does not contain the initial state

#	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

Basic statements need to be verified semantically. If this can be done efficiently depends on the state set representation:

$$S_1 \subseteq S_2 \Leftrightarrow \varphi_1 \models \varphi_2$$

- ▶ efficient for BDDs, explicit enumeration, (dual-)Horn and 2CNF formulas
- ▶ not efficient for CNF formulas

Basic Statements for CNF

Planner calls SAT solver, which is a certifying algorithm.

\rightarrow Integrate UNSAT certificates into proof

	statement	required UNSAT certificate(s)
C1a	$S_1 \subseteq S_2$	$\varphi_1 \wedge \neg \gamma$ for each γ in φ_2
C1b	$S_1 \subseteq \overline{S_2}$	$\varphi_1 \wedge \varphi_2$
C2a	$S_1[A] \subseteq S_2$	$\varphi_1 \wedge T_A \wedge \neg \gamma'$ for each γ' in φ_2
C2b	$S_1[A] \subseteq \overline{S_2}$	$\varphi_1 \wedge T_A \wedge \varphi_2'$
C3a	$[A]S_1 \subseteq S_2$	$\varphi_1' \wedge T_A \wedge \neg \gamma$ for each γ in φ_2
C3b	$[A]S_1 \subseteq \overline{S_2}$	$\varphi_1' \wedge T_A \wedge \varphi_2'$

- ▶ state sets S_i represented by CNF formulas $\varphi_i = \bigwedge \gamma_i$
- ▶ transition formula T_A encodes pairs of states (s, s') with $s[a] = s'$ for $a \in A$

Modified Certificate for PDR with SAT

The SAT calls performed by PDR don't match the required certificates.

\rightarrow modify basic statements and use additional inference rules:

#	statement	justification
(1a)	$[A]L_u \subseteq \text{states}(\gamma)$ for all γ in φ_{L_u}	SAT certificates provided by planner from (1a) with rule SI
(1b)	$[A]L_u \subseteq \overline{L_u}$	build UNSAT certificate by hand*
(2)	$\{I\} \subseteq \overline{L_u}$	from (1b) and (2) with rule RI
(3)	L_u is dead	build UNSAT certificates by hand*
(4a)	$G \subseteq \text{states}(\gamma)$ for all γ in φ_{L_u}	from (4a) with rule SI
(4b)	$G \subseteq L_u$	from (3) and (4b) with rule SD
(5)	G is dead	

*formula can be proven unsolvable solely by unit propagation

Experimental Evaluation (PDR without SAT)

	base	certifying	verifier
PDR	388	384	382
FD- $h^{M\&S}$	224	197	178
FD- h^{\max}	203	156	140
DFS-CL	394	386	385

