A Proof System for Unsolvable Planning Tasks

Salomé Eriksson Gabriele Röger Malte Helmert

University of Basel, Switzerland

ICAPS 2018

Motivation

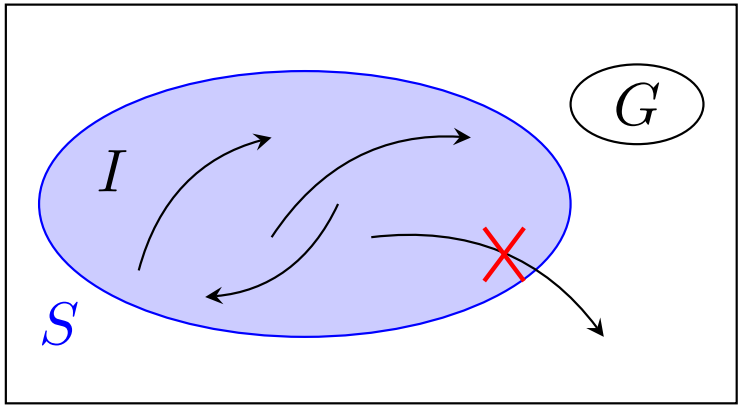
validating correctness of planner output:

- Why?
  $\rightsquigarrow$ software bugs, hardware faults, malicious reasons ...
- How?
  (a) plan output: VAL/INVAL
  (b) unsolvability claim: inductive certificates [Eriksson et al. 2017]

Inductive Certificates

find set S with

- no successors
  (written: $S[A] \subseteq S$)
- containing I
- no goal



weakness: not compositional

$\rightsquigarrow$ new approach: proof system

Proof System

collection of knowledge $(A \cap C) \subset B$, $a \in A$ ...

new knowledge gained through:

- basic statements $A \subset B$
  - state facts about concrete objects
  - need to be verified
- derivation rules $X \subset Y$ and $Y \subset Z \rightarrow X \subset Z$
  - derive new knowledge from existing knowledge
  - universally true (only verify correct application)

Unsolvability Proof System

objects: state sets S in different formalisms

- BDDs
- Horn formulas
- 2CNF formulas
- explicit

types of statements:

- S dead (no plan through any $s \in S$)
- $S \subseteq S'$
- task unsolvable

Rules

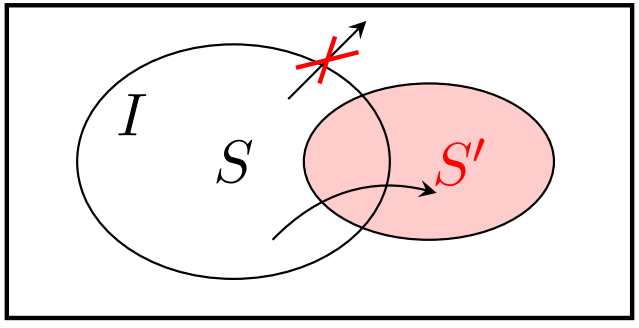
D1			$\emptyset$ is dead
D2	S dead, S' dead	$\rightarrow$	$S \cup S'$ dead
D3	$S' \subseteq S$, S dead	$\rightarrow$	S' dead
D4	$\{I\}$ dead	$\rightarrow$	task unsolvable
D5	G dead	$\rightarrow$	task unsolvable
D6	$S[A] \subseteq S \cup S'$, S' dead, $G \cap S$ dead	$\rightarrow$	S dead
D7	$S[A] \subseteq S \cup S'$, S' dead, $\{I\} \subseteq S$	$\rightarrow$	$\overline{S}$ dead
D8	$[A]S \subseteq S \cup S'$, S' dead, $G \cap \overline{S}$ dead	$\rightarrow$	$\overline{S}$ dead
D9	$[A]S \subseteq S \cup S'$, S' dead, $\{I\} \subseteq \overline{S}$	$\rightarrow$	S dead

Rules

D1			$\emptyset$ is dead
D2	S dead, S' dead	$\rightarrow$	$S \cup S'$ dead
D3	$S' \subseteq S$, S dead	$\rightarrow$	S' dead
D4	$\{I\}$ dead	$\rightarrow$	task unsolvable
D5	G dead	$\rightarrow$	task unsolvable
D6	$S[A] \subseteq S \cup S'$, S' dead, $G \cap S$ dead	$\rightarrow$	S dead
D7	$S[A] \subseteq S \cup S'$, S' dead, $\{I\} \subseteq S$	$\rightarrow$	$\overline{S}$ dead
D8	$[A]S \subseteq S \cup S'$, S' dead, $G \cap \overline{S}$ dead	$\rightarrow$	$\overline{S}$ dead
D9	$[A]S \subseteq S \cup S'$, S' dead, $\{I\} \subseteq \overline{S}$	$\rightarrow$	S dead

Rules

D1			$\emptyset$ is dead
D2	S dead, S' dead	$\rightarrow$	$S \cup S'$ dead
D3	$S' \subseteq S$, S dead	$\rightarrow$	S' dead
D4	$\{I\}$ dead	$\rightarrow$	task unsolvable
D5	G dead	$\rightarrow$	task unsolvable
D6	$S[A] \subseteq S \cup S'$, S' dead, $G \cap S$ dead	$\rightarrow$	S dead
D7	$S[A] \subseteq S \cup S'$, S' dead, $\{I\} \subseteq S$	$\rightarrow$	$\overline{S}$ dead
D8	$[A]S \subseteq S \cup S'$, S' dead, $G \cap \overline{S}$ dead	$\rightarrow$	$\overline{S}$ dead
D9	$[A]S \subseteq S \cup S'$, S' dead, $\{I\} \subseteq \overline{S}$	$\rightarrow$	S dead



Basic Statements

currently restricted to certain subset relations:

B1	$S \subseteq S'$
B2	$S \subseteq S' \cup S''$
B3	$S \cap G \subseteq S'$
B4	$S[A] \subseteq S \cup S'$
B5	$[A]S \subseteq S \cup S'$

S / S': constants ($\{I\}$, G, $\emptyset$), set variables or their complement

Verification in polynomial time:

- B1-B5 if homogeneous (same representation for all S)
- B1 for heterogeneous in some cases

Covered techniques

- blind search (explicit and symbolic)
- heuristic search with one heuristic:
  - delete-relaxation (h^{max}, $h^{\mathsf{LM\text{-}Cut}}$, ...)
  - $h^{\mathsf{M\&S}}$ with linear merge strategy
  - h^m (and h^C)
- trapper [Lipovetzky et al. 2016]

not covered by old approach

- heuristic search with multiple heuristics
- h^2-based preprocessing [Alcázar and Torralba 2015]

Translating Inductive Certificates

inductive certificate S:

- no successor
- containing I
- no goal

(1)	$\emptyset$ dead	D1	
(2)	$S[A] \subseteq S \cup \emptyset$	B4	
(3)	$S \cap G \subseteq \emptyset$	B3	
(4)	$S \cap G$ dead	D3	(3),(1)
(5)	S dead	D6	(2),(1),(4)
(6)	$\{I\} \subseteq S$	B1	
(7)	$\{I\}$ dead	D3	(6),(5)
(8)	unsolvable	D5	(7)

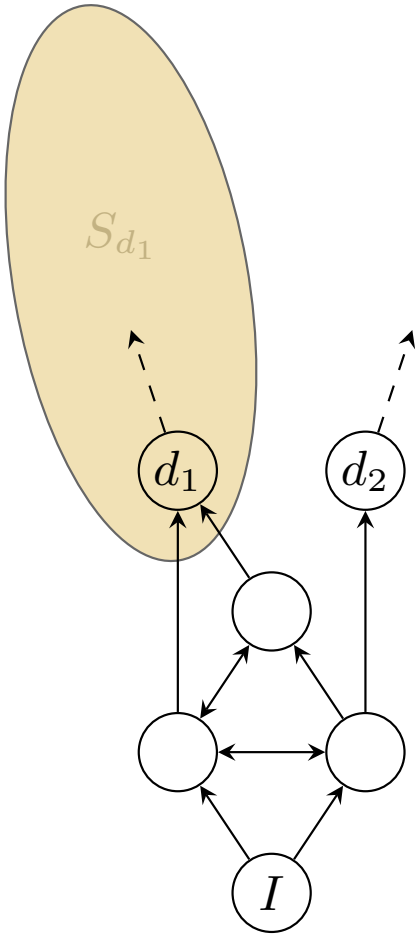
Heuristic Search

How does heuristic search show unsolvability?

- dead-ends are dead
- expanded states lead only to expanded states or dead ends

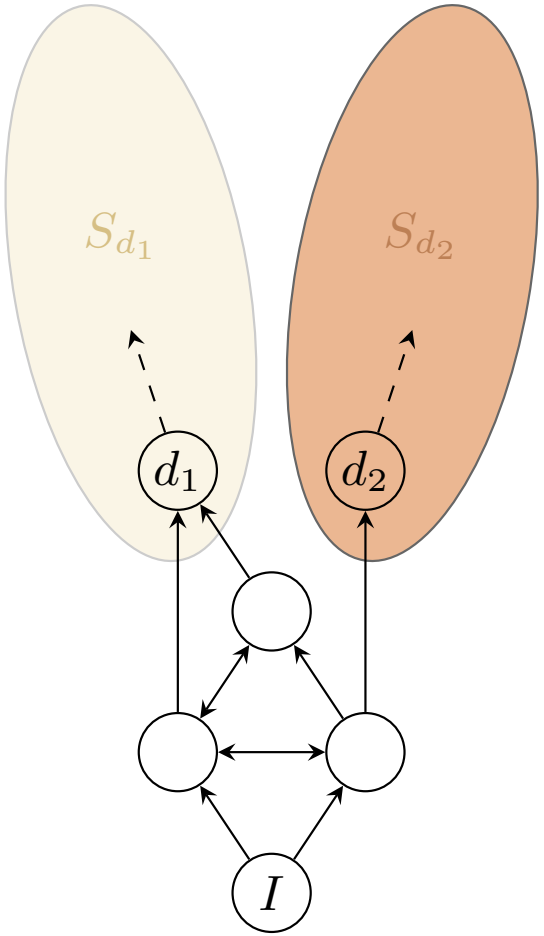
$\rightsquigarrow$ showing deadness of dead states independently

Heuristic Search - Example



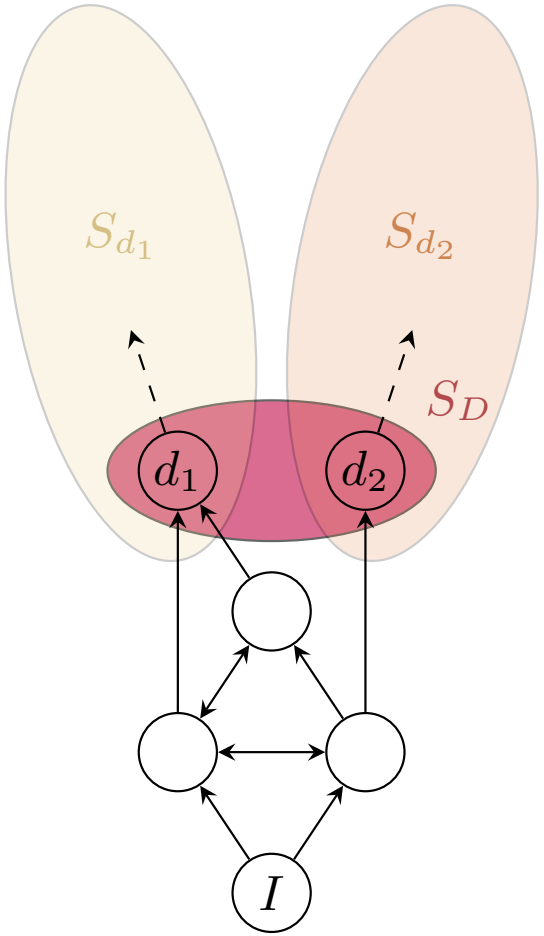
(1)	$\emptyset$ dead	D1	
(2)	$S_{d_1}[A] \subseteq S_{d_1} \cup \emptyset$	B4	
(3)	$S_{d_1} \cap G \subseteq \emptyset$	B3	
(4)	$S_{d_1} \cap G$ dead	D3	(3),(1)
(5)	S_{d_1} dead	D6	(2),(1),(4)
(6)	$\{d_1\} \subseteq S_{d_1}$	B1	
(7)	$\{d_1\}$ dead	D3	(6),(5)

Heuristic Search - Example



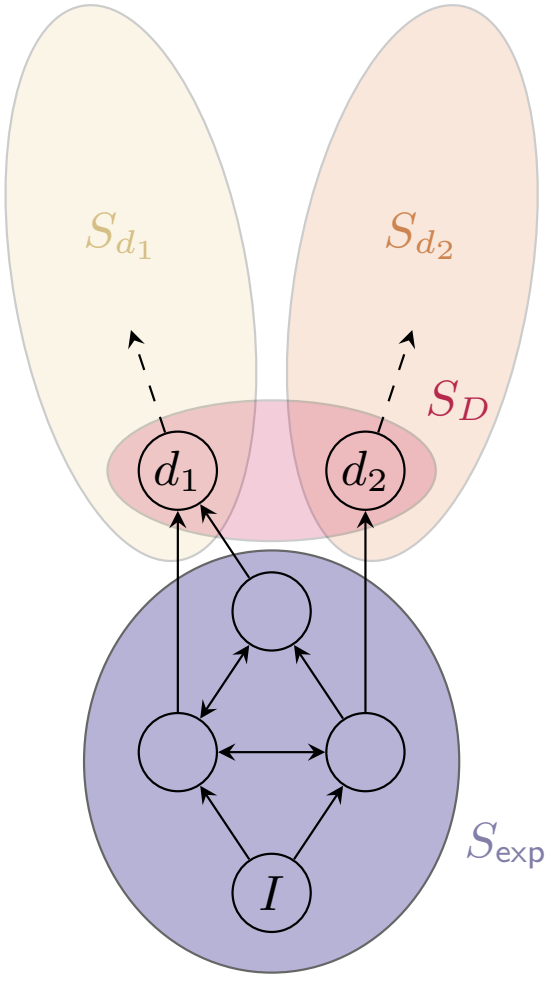
(1)	$\emptyset$ dead	D1	
(8)	$S_{d_2}[A] \subseteq S_{d_2} \cup \emptyset$	B4	
(9)	$S_{d_2} \cap G \subseteq \emptyset$	B3	
(10)	$S_{d_2} \cap G$ dead	D3	(9),(1)
(11)	S_{d_2} dead	D6	(8),(1),(10)
(12)	$\{d_2\} \subseteq S_{d_2}$	B1	
(13)	$\{d_2\}$ dead	D3	(12),(11)

Heuristic Search - Example



(7)	$\{d_1\}$ dead		
(13)	$\{d_2\}$ dead		
(14)	$\{d_1\} \cup \{d_2\}$ dead	D2	(7),(13)
(15)	$S_D \subseteq \{d_1\} \cup \{d_2\}$	B2	
(16)	S_D dead	D3	(15),(14)

Heuristic Search - Example



(1)	$\emptyset$ dead	D1	
(16)	S_D dead		
(17)	$S_{\text{exp}}[A] \subseteq S_{\text{exp}} \cup S_D$	B4	
(18)	$S_{\text{exp}} \cap G \subseteq \emptyset$	B3	
(19)	$S_{\text{exp}} \cap G$ dead	D3	(18),(1)
(20)	S_{exp} dead	D6	(17),(16),(19)
(21)	$\{I\} \subseteq S_{\text{exp}}$	B1	
(22)	$\{I\}$ dead	D3	(21),(20)
(23)	task unsolvable	D4	(22)

Experimental evaluation

implementation of proof generation and independent verifier[1]

algorithms:

- A* search with $h^{\max}$ $h^{\text{M\&S}}$ and h^2
- A* with maximum of $h^{\text{M\&S}}$ and h^2
- clause-learning state space search (DFS-CL) [Steinmetz and Hoffmann 2016]

limits:

- proof generation: 30min, 2GiB
- proof verification: 4h, 2GiB

[1] both available at `https://doi.org/10.5281/zenodo.1196473`

Coverage

	base	certifying	verifier
FD-$h^{\max}$	211	168 (135)*	167 (125)*
FD-$h^{\text{M\&S}}$	230	191 (200)*	184 (163)*
FD-h^2	183	177	177
FD-max$(h^{\text{M\&S}}, h^2)$	204	199	195
DFS-CL	385	386	383

*inductive certificates approach

- generate proofs in 92% within limits
- verify proofs in 99% within limits
- better coverage than certificates

Conclusion

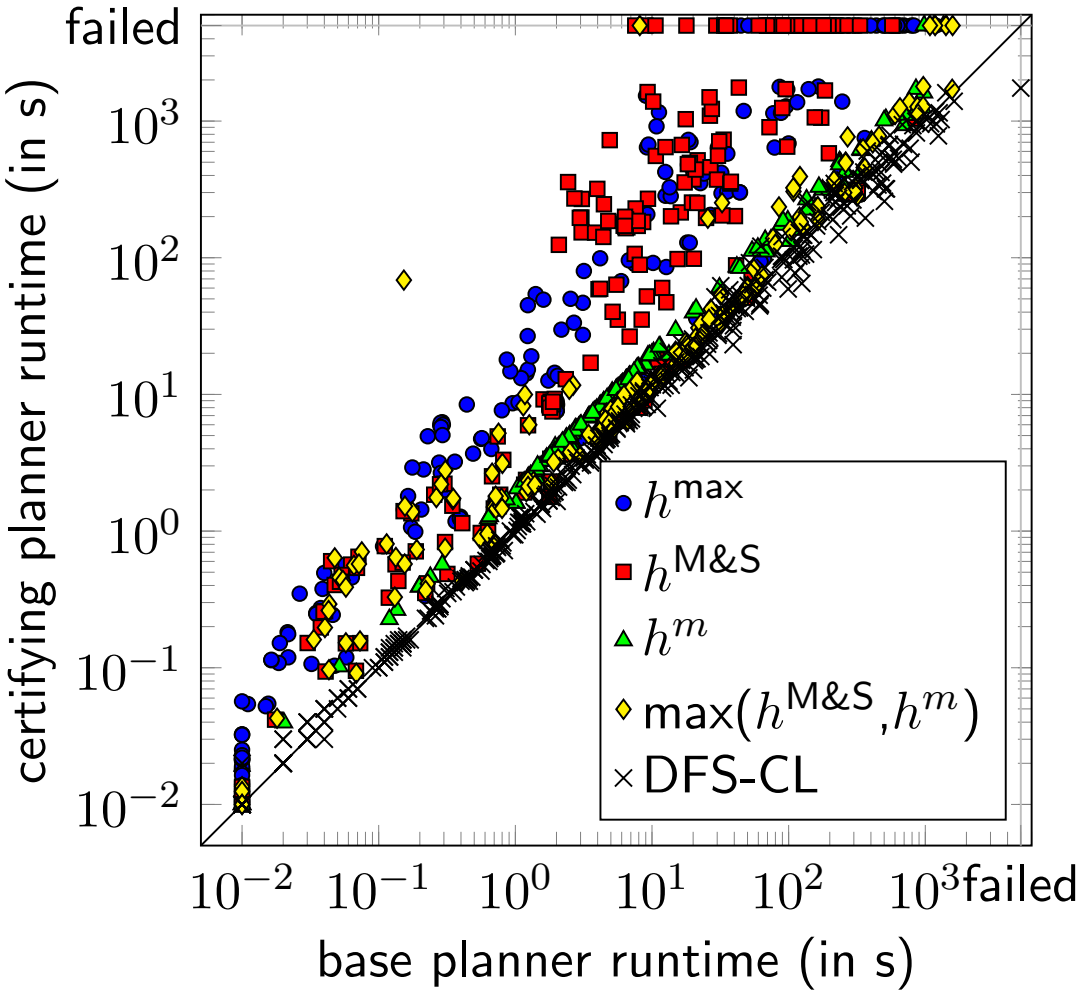
Compositional Proof System

- combination of different approaches possible
- covers wide area of planning techniques
- efficient generation and verification

future work:

- partial order reduction
- flow & potential heuristics

Overhead


failed
10^3
10^2
10^1
10^0
10^{-1}
10^{-2}
certifying planner runtime (in s)
$h^{\max}$
$h^{\text{M\&S}}$
h^m
$\max(h^{\text{M\&S}}, h^m)$
DFS-CL
10^{-2} 10^{-1} 10^0 10^1 10^2 10^3failed
base planner runtime (in s)

Comparison Certificate Size