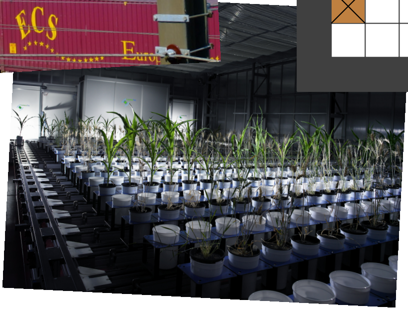
Certifying Planning Systems: Witnesses for Unsolvability

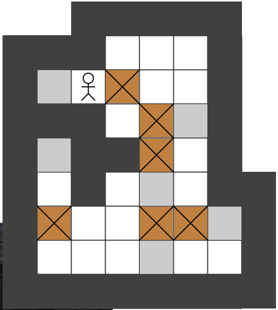
Salomé Eriksson

University of Basel, Switzerland

April 26, 2019

Classical Planning



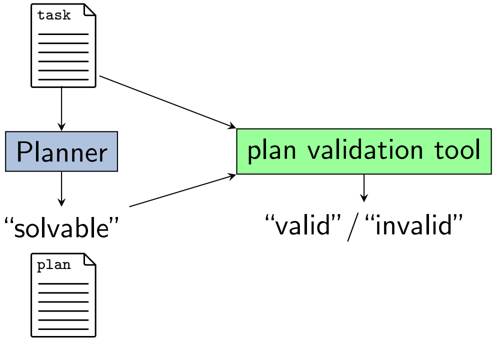


Validating Planner Output

- Why?
  - software bugs
  - hardware faults
  - malicious reasons
  - ...
- How?
  - tests on known instances
  - formal correctness proofs
  - certifying algorithms

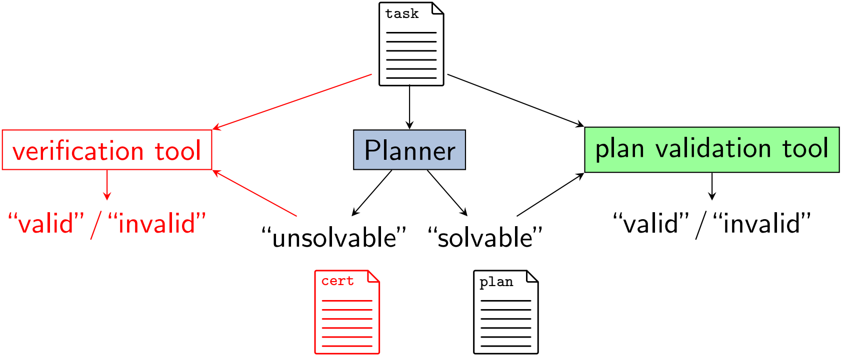
Certifying Algorithms

generate a witness alongside answer:



Certifying Algorithms

generate a witness alongside answer:



Contribution

Main Contributions

two suitable witness types for unsolvable planning tasks:

I Inductive Certificates

II Proof System

theoretical and experimental comparison

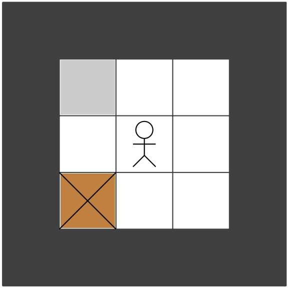
suitability measures:

- soundness & completeness
- efficient generation and verification
- generality

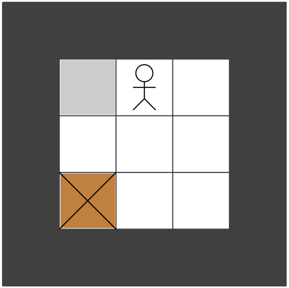
Witness I: Inductive Certificates

[E, Röger, Helmert, ICAPS 2017]

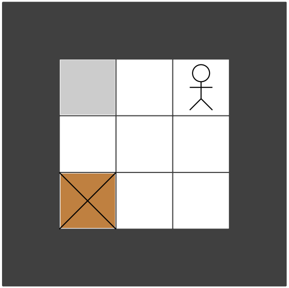
Inductive Sets



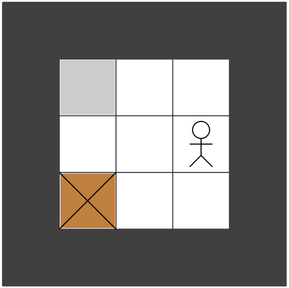
Inductive Sets



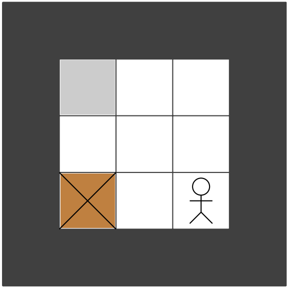
Inductive Sets



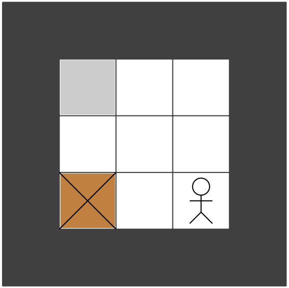
Inductive Sets



Inductive Sets



Inductive Sets



can only reach states with "box in corner"

Inductive Set

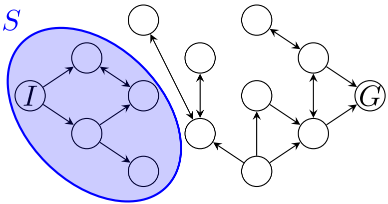
A set of states is inductive if all action applications to a state in S lead to a state which is also in S. ($S[A] \subseteq S$).

Inductive Certificate

Inductive Certificate

set of states S with following properties:

- contains I
- contains no goal
- inductive



Soundness & Completeness

Theorem

Inductive certificates are sound and complete.

states reachable from I:

- contains I
- is inductive
- contains no goal if task solvable

Efficient Verification

depends on how S is represented

- formalisms based on propositional logic
- Which logical operations are needed for efficient verification?

several commonly used formalisms support needed operations

Composite Certificates

not all sets can be compactly described
$\rightsquigarrow$ represent as union or intersection of sets

r-disjunctive Certificates

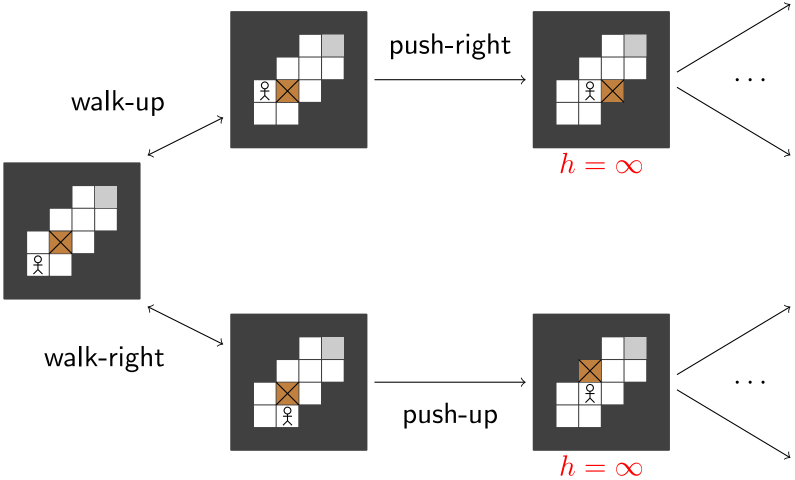
family $\mathcal{F}$ of sets with:

- $I \in S$ for some $S \in \mathcal{F}$
- no goal in any $S \in \mathcal{F}$
- $S[a] \subseteq \bigcup_{S' \in \mathcal{F}'} S'$ for all $a \in A$, $S \in \mathcal{F}$ with $\mathcal{F}' \subseteq \mathcal{F}$ and $|\mathcal{F}'| \leq r$.

Application to Heuristic Search

heuristic can detect dead-ends
$\rightsquigarrow$ set of reachable states not explored fully

Application to Heuristic Search



Application to Heuristic Search

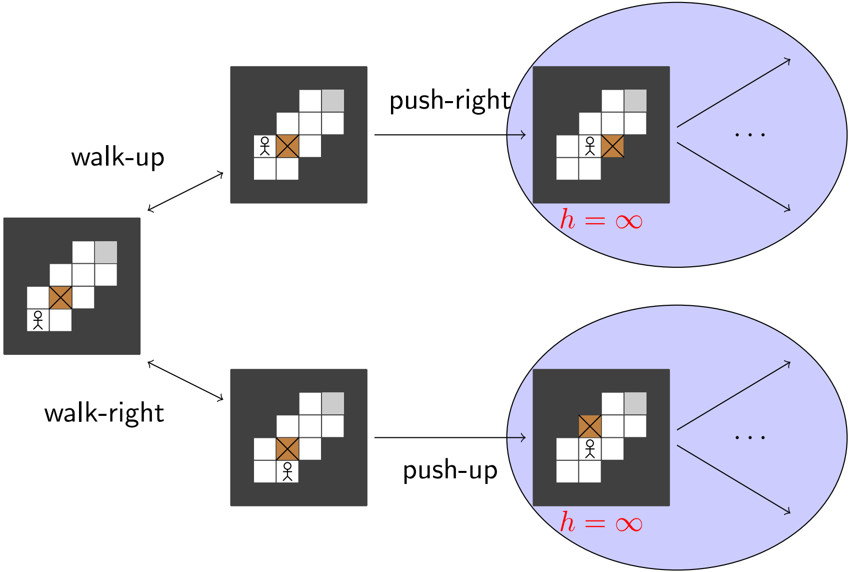
heuristic can detect dead-ends
$\rightsquigarrow$ set of reachable states not explored fully

Heuristic Search Certificate

Union of:

- inductive set for each dead-end
  - for each $a \in A$: leads to itself

Application to Heuristic Search



Application to Heuristic Search

heuristic can detect dead-ends
$\rightsquigarrow$ set of reachable states not explored fully

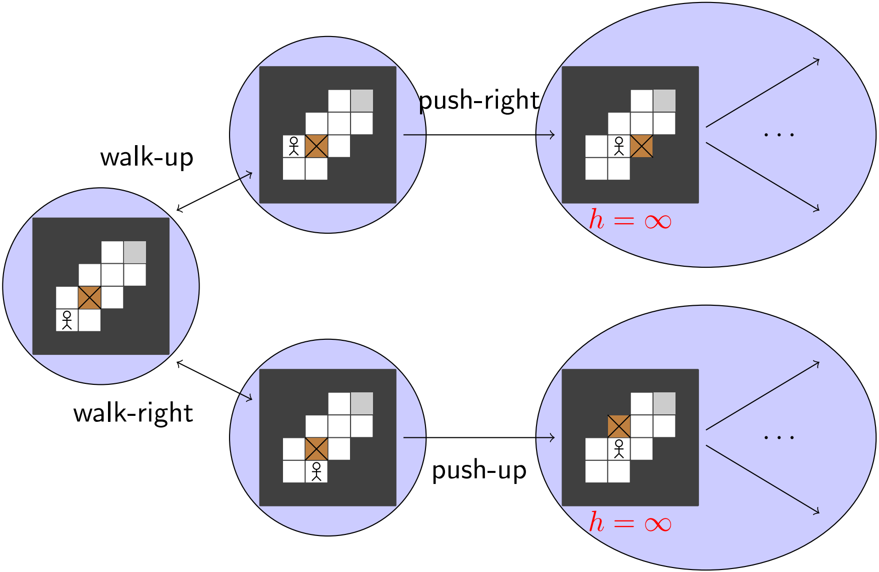
Heuristic Search Certificate

Union of:

- inductive set for each dead-end
  - for each $a \in A$: leads to itself
- one set for each expanded state
  - for each $a \in A$: leads to one expanded or dead-end state

$\rightsquigarrow$ 1-disjunctive

Application to Heuristic Search



Generating Inductive Certificates

	certificates
blind search	yes
heuristic search	
- single heuristic	yes
- several heuristics	**if same formalism**
h^+	yes
h^m	yes
$h^{\text{M\&S}}$	yes
Landmarks	yes
Trapper	yes
Iterative dead pairs	**no**
CLS	yes

Weaknesses

monolithic: find one inductive set

- cannot mix representations
  - several heuristics
- cannot cover techniques not built on inductive sets
  - iterative dead pairs

Witness II: Proof System

[E, Röger, Helmert, ICAPS 2018]

Dead States

incrementally rule out parts of the search space

Definition

A state s is dead if no plan traverses s.
A set of states is dead if all its elements are dead.

initial state / all goal states dead $\rightsquigarrow$ task unsolvable

Proof Systems

based on rules with premises A_i and conclusion B:

$$\frac{A_1 \qquad \dots \qquad A_n}{B}$$

universally true

Rules

- showing that state sets are dead
- end proof
- set theory

Rules

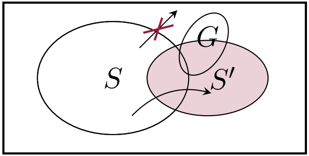
- showing that state sets are dead
- end proof
- set theory

$$\frac{S' \text{ dead} \qquad S \subseteq S'}{S \text{ dead}}$$

Rules

- showing that state sets are dead
- end proof
- set theory

$$\frac{S[A] \subseteq S \cup S' \qquad S' \text{ dead} \qquad S \cap G \text{ dead}}{S \text{ dead}}$$



Rules

- showing that state sets are dead
- end proof
- set theory

$$\frac{I \text{ dead}}{\text{unsolvable}}$$

$$\frac{G \text{ dead}}{\text{unsolvable}}$$

Rules

- showing that state sets are dead
- end proof
- set theory

$$\frac{}{S \subseteq (S \cup S')}$$

$$\frac{S \subseteq S' \qquad S' \subseteq S''}{S \subseteq S''}$$

Basic Statements

show $S \subseteq S'$ holds for concrete sets?
$\rightsquigarrow$ basic statements

- verified for concrete task
- establish "initial" knowledge base

Soundness & Completeness

Theorem

Proofs in the proof system are sound and complete.

inductive certificate S:

- no successor
- containing I
- no goal

(1) $\emptyset$ dead
(2) $S[A] \subseteq S \cup \emptyset$
(3) $S \cap G \subseteq \emptyset$
(4) $S \cap G$ dead
(5) S dead
(6) $I \in S$
(7) I dead
(8) unsolvable

Efficient Verification

rule verification trivial $\rightsquigarrow$ only depends on basic statements

different forms of $S \subseteq S'$:

- S as a intersection of sets
- S' as a union of sets
- S and S' represented in different formalisms

translated inductive certificates require same operations

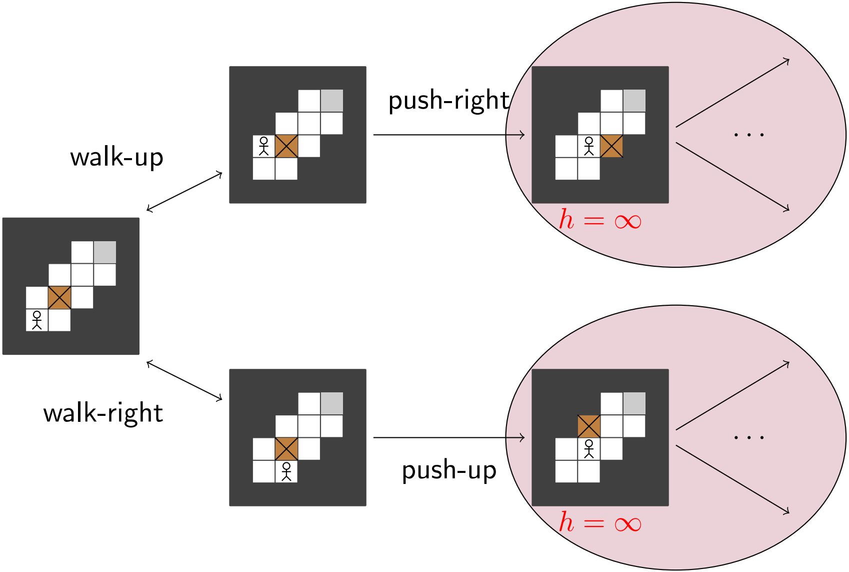
Application to Heuristic Search

Heuristic Search Proof

proof structure:

1. each dead end is dead (inductive set)

Application to Heuristic Search



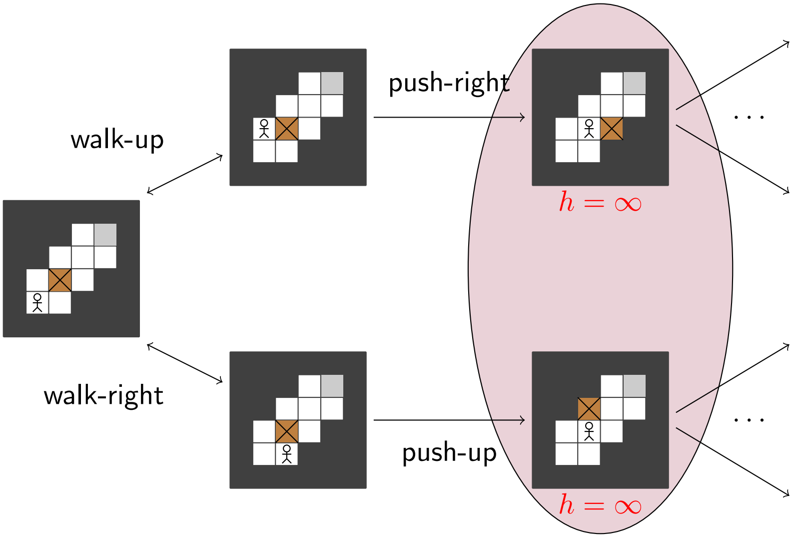
Application to Heuristic Search

Heuristic Search Proof

proof structure:

1. each dead end is dead (inductive set)
2. union of all dead ends is dead

Application to Heuristic Search



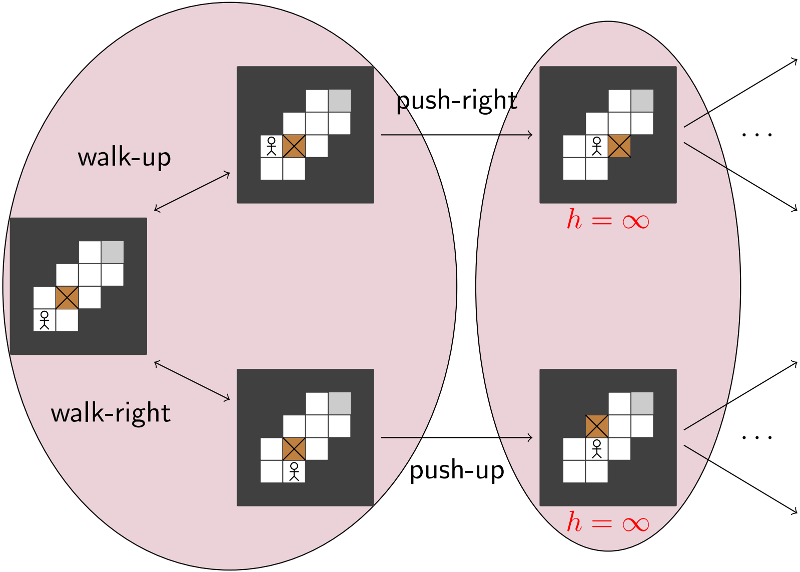
Application to Heuristic Search

Heuristic Search Proof

proof structure:

1. each dead end is dead (inductive set)
2. union of all dead ends is dead
3. expanded$[A]$ $=$ expanded $\cup$ dead $\leadsto$ expanded dead
4. $I \in$ expanded $\leadsto$ I dead.

Application to Heuristic Search



Generating Proofs

	certificates	proofs
blind search	yes	yes
heuristic search		
- single heuristic	yes	yes
- several heuristics	**if same formalism**	yes
h^+	yes	yes
h^m	yes	yes
$h^{\mathsf{M\&S}}$	yes	yes
Landmarks	yes	yes
Trapper	yes	yes
Iterative dead pairs	**no**	yes
CLS	yes	yes

Comparison

Theoretical Comparison

- both witnesses sound & complete
- proof covers more examined techniques
- translation certificate $\rightarrow$ proof possible
    - also for composite certificates, but at cost of size increase

$\rightsquigarrow$ proof system more expressive

Experimental Evaluation

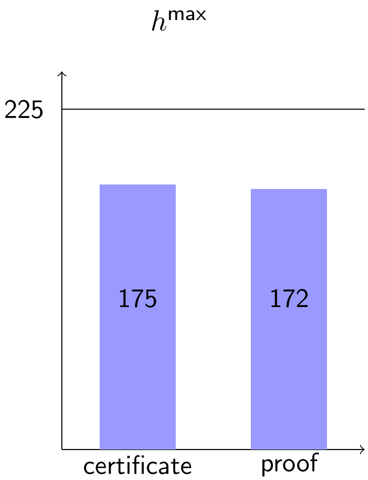
comparison for A* search with

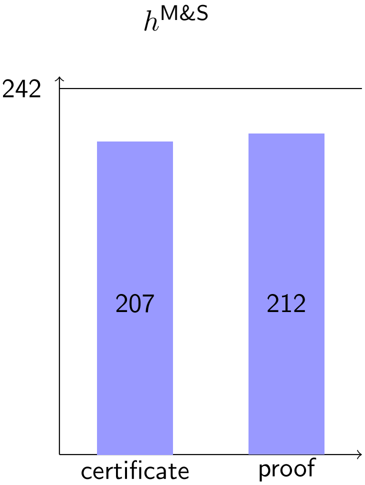
- h^{max}
- $h^{\text{M\&S}}$

limits:

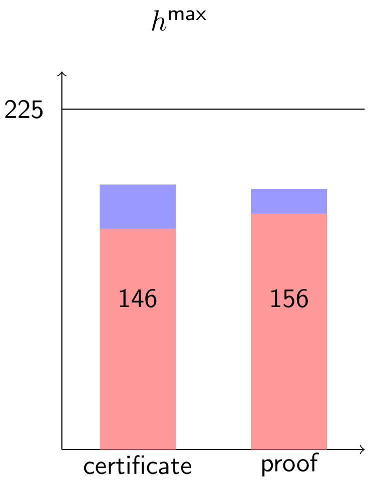
- generate: 30 minutes
- verify: 4 hours

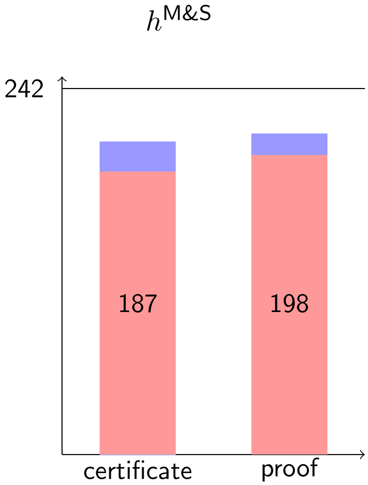
Coverage - Generation



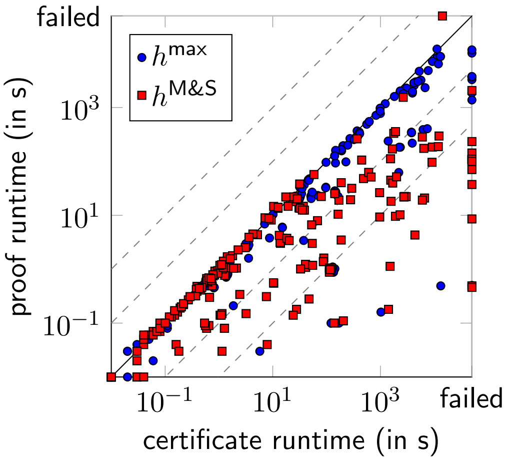


Coverage - Verification



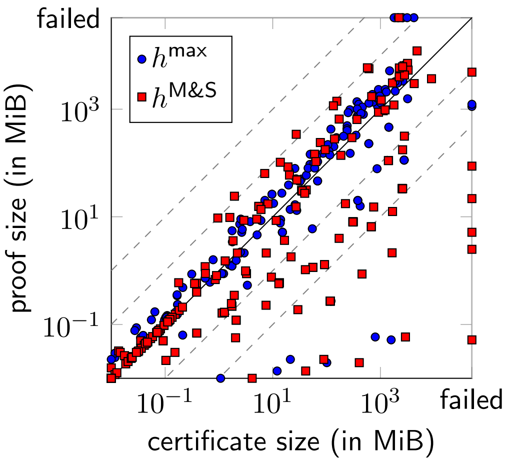


Verification



certificate repeats explicit search

Witness Size



Conclusion

Summary

Inductive Certificates

- describes invariant property which I has but not G
- concise argument for unsolvability
- lacks composability

Proof System

- explicit reasoning with simple rules
- versatile and extensible

Logical Operations

	BDD	Horn	2CNF	MODS
MO	yes	yes	yes	yes
CO	yes	yes	yes	yes
VA	yes	yes	yes	yes
CE	yes	yes	yes	yes
IM	yes	yes	yes	yes
SE	yes	yes	yes	yes
ME	yes	yes	yes	yes
$\wedge$**BC**	yes	yes	yes	yes
$\wedge$**C**	no	yes	yes	no
$\vee$**BC**	yes	no	no	no*
$\vee$**C**	no	no	no	no
$\neg$**C**	yes	no	no	no
CL	yes	yes	yes	yes
RN	no	yes	yes	yes
RN$_{\prec}$	yes	yes	yes	yes
toDNF	no	no	no	yes
toCNF	no	yes	yes	no
CT	yes	(no)	(no)	yes

Transition formula

Traditional:

$$\varphi \wedge \bigwedge_{v_p \in \textit{\textbf{pre}}(a)} v_p \wedge \bigwedge_{v_a \in \textit{\textbf{add}}(a)} v'_a \wedge \bigwedge_{v_d \in (\textit{\textbf{del}}(a) \setminus \textit{\textbf{add}}(a))} \neg v'_d$$

$$\wedge \bigwedge_{v \in (V^{\Pi} \setminus (\textit{\textbf{add}}(a) \cup \textit{\textbf{del}}(a))} (v \leftrightarrow v') \models \varphi[V \rightarrow V']$$

New:

$$\Big((\varphi \wedge \bigwedge_{v_p \in \textit{\textbf{pre}}(a)} v_p)[(\textit{\textbf{add}}(a) \cup \textit{\textbf{del}}(a)) \rightarrow X']\Big)$$

$$\wedge \bigwedge_{v_a \in \textit{\textbf{add}}(a)} v_a \wedge \bigwedge_{v_d \in (\textit{\textbf{del}}(a) \setminus \textit{\textbf{add}}(a))} \neg v_d \models \varphi$$

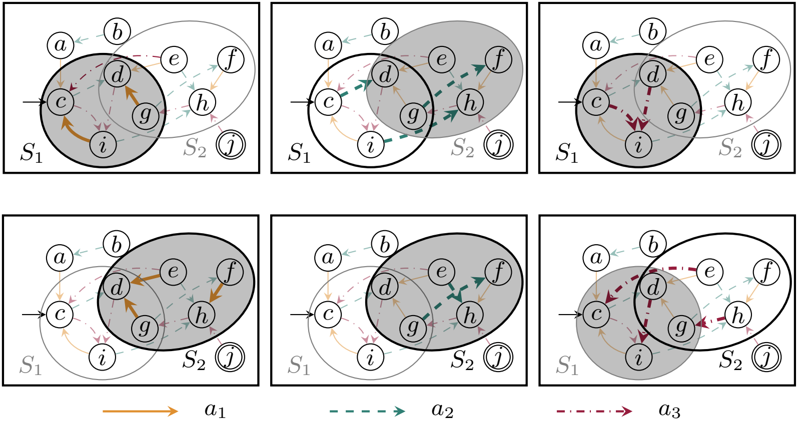
Disjunctive Certificates

r-disjunctive certificate

For $r \in \mathbb{N}_0$, a family $\mathcal{F} \subseteq 2^{S^\Pi}$ of state sets of task $\Pi = \langle V^\Pi, A^\Pi, I^\Pi, G^\Pi \rangle$ is called an *r-disjunctive certificate* if:

1. $I^\Pi \in S$ for some $S \in \mathcal{F}$,
2. $S \cap S_G^\Pi = \emptyset$ for all $S \in \mathcal{F}$, and
3. for all $S \in \mathcal{F}$ and all $a \in A^\Pi$, there is a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ with $|\mathcal{F}'| \leq r$ such that $S[a] \subseteq \bigcup_{S' \in \mathcal{F}'} S'$.

Disjunctive Certificates



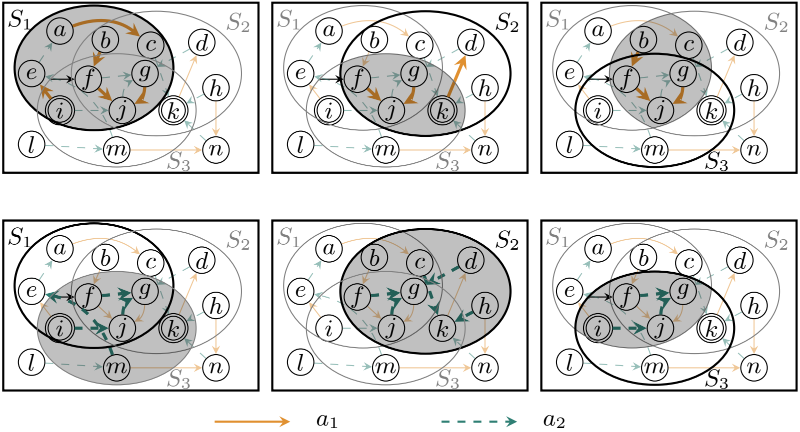
Conjunctive Certificates

r-conjunctive certificate

For $r \in \mathbb{N}_0$, a family $\mathcal{F} \subseteq 2^{S^\Pi}$ of state sets of task $\Pi = \langle V^\Pi, A^\Pi, I^\Pi, G^\Pi \rangle$ is called an *r-conjunctive certificate* if:

1. $I^\Pi \in S$ for all $S \in \mathcal{F}$,
2. there is a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ with $|\mathcal{F}'| \leq r$ such that $(\bigcap_{S \in \mathcal{F}'} S) \cap S_G^\Pi = \emptyset$, and
3. for all $S \in \mathcal{F}$ and all $a \in A^\Pi$, there is a subfamily $\mathcal{F}' \subseteq \mathcal{F}$ with $|\mathcal{F}'| \leq r$
   such that $(\bigcap_{S' \in \mathcal{F}'} S')[a] \subseteq S$.

Conjunctive Certificates



Proof System Rules

Empty set **D**ead

$$\frac{}{\emptyset \text{ dead}}\ \text{ED}$$

Union **D**ead

$$\frac{S \text{ dead} \qquad S' \text{ dead}}{S \cup S' \text{ dead}}\ \text{UD}$$

Subset **D**ead

$$\frac{S' \text{ dead} \qquad S \sqsubseteq S'}{S \text{ dead}}\ \text{SD}$$

Progression **G**oal

$$\frac{S[A^{\Pi}] \sqsubseteq S \cup S' \qquad S' \text{ dead} \qquad S \cap S_G^{\Pi} \text{ dead}}{S \text{ dead}}\ \text{PG}$$

Progression **I**nitial

$$\frac{S[A^{\Pi}] \sqsubseteq S \cup S' \qquad S' \text{ dead} \qquad \{I^{\Pi}\} \sqsubseteq S}{\overline{S} \text{ dead}}\ \text{PI}$$

Regression **G**oal

$$\frac{[A^{\Pi}]S \sqsubseteq S \cup S' \qquad S' \text{ dead} \qquad \overline{S} \cap S_G^{\Pi} \text{ dead}}{\overline{S} \text{ dead}}\ \text{RG}$$

Regression **I**nitial

$$\frac{[A^{\Pi}]S \sqsubseteq S \cup S' \qquad S' \text{ dead} \qquad \{I^{\Pi}\} \sqsubseteq \overline{S}}{S \text{ dead}}\ \text{RI}$$

Proof System Rules

Conclusion **I**nitial $$\frac{\{I^\Pi\}\ \text{dead}}{\text{unsolvable}}\ \text{CI}$$

Conclusion **G**oal $$\frac{S_G^\Pi\ \text{dead}}{\text{unsolvable}}\ \text{CG}$$

Proof System Rules

Union **R**ight	$\dfrac{}{E \sqsubseteq (E \cup E')}$ UR
Union **L**eft	$\dfrac{}{E \sqsubseteq (E' \cup E)}$ UL
Intersection **R**ight	$\dfrac{}{(E \cap E') \sqsubseteq E}$ IR
Intersection **L**eft	$\dfrac{}{(E' \cap E) \sqsubseteq E}$ IL
DIstributivity	$\dfrac{}{((E \cup E') \cap E'') \sqsubseteq ((E \cap E'') \cup (E' \cap E''))}$ DI
Subset **U**nion	$\dfrac{E \sqsubseteq E'' \qquad E' \sqsubseteq E''}{(E \cup E') \sqsubseteq E''}$ SU
Subset **I**ntersection	$\dfrac{E \sqsubseteq E' \qquad E \sqsubseteq E''}{E \sqsubseteq (E' \cap E'')}$ SI
Subset **T**ransitivity	$\dfrac{E \sqsubseteq E' \qquad E' \sqsubseteq E''}{E \sqsubseteq E''}$ ST

Proof System Rules

Action **T**ransitivity

$$\frac{S[A] \sqsubseteq S' \qquad A' \sqsubseteq A}{S[A'] \sqsubseteq S'}\ \text{AT}$$

Action **U**nion

$$\frac{S[A] \sqsubseteq S' \qquad S[A'] \sqsubseteq S'}{S[A \cup A'] \sqsubseteq S'}\ \text{AU}$$

Progression **T**ransitivity

$$\frac{S[A] \sqsubseteq S'' \qquad S' \sqsubseteq S}{S'[A] \sqsubseteq S''}\ \text{PT}$$

Progression **U**nion

$$\frac{S[A] \sqsubseteq S'' \qquad S'[A] \sqsubseteq S''}{(S \cup S')[A] \sqsubseteq S''}\ \text{PU}$$

Progression to **R**egression

$$\frac{S[A] \sqsubseteq S'}{[A]\overline{S'} \sqsubseteq \overline{S}}\ \text{PR}$$

Regression to **P**rogression

$$\frac{[A]\overline{S'} \sqsubseteq \overline{S}}{S[A] \sqsubseteq S'}\ \text{RP}$$

Proof System Basic Statements

1. $\bigcap_{L_{\mathbf{R}} \in \mathcal{L}} L_{\mathbf{R}} \subseteq \bigcup_{L'_{\mathbf{R}} \in \mathcal{L}'} L'_{\mathbf{R}}$
   with $|\mathcal{L}| + |\mathcal{L}'| \leq r$
2. $(\bigcap_{X_{\mathbf{R}} \in \mathcal{X}} X_{\mathbf{R}})[A] \cap \bigcap_{L_{\mathbf{R}} \in \mathcal{L}} L_{\mathbf{R}} \subseteq \bigcup_{L'_{\mathbf{R}} \in \mathcal{L}'} L'_{\mathbf{R}}$
   with $|\mathcal{X}| + |\mathcal{L}| + |\mathcal{L}'| \leq r$
3. $[A](\bigcap_{X_{\mathbf{R}} \in \mathcal{X}} X_{\mathbf{R}}) \cap \bigcap_{L_{\mathbf{R}} \in \mathcal{L}} L_{\mathbf{R}} \subseteq \bigcup_{L'_{\mathbf{R}} \in \mathcal{L}'} L'_{\mathbf{R}}$
   with $|\mathcal{X}| + |\mathcal{L}| + |\mathcal{L}'| \leq r$
4. $L_{\mathbf{R}} \subseteq L'_{\mathbf{R}'}$
5. $A \subseteq A'$

Proof System Basic Statements

$\bigcap_{L_i \in \mathcal{L}} L_i \subseteq \bigcup_{L'_i \in \mathcal{L}'} L'_i$:

	$\mathcal{L}^+ + \mathcal{L}'^- = 0$	$\mathcal{L}^+ + \mathcal{L}'^- = 1$	$\mathcal{L}^+ + \mathcal{L}'^- > 1$
$\mathcal{L}^- + \mathcal{L}'^+ = 0$		**CO**	**CO**, $\wedge$**BC** **toDNF**
$\mathcal{L}^- + \mathcal{L}'^+ = 1$	**VA**	**SE**	**SE**, $\wedge$**BC** **toDNF**, **IM**
$\mathcal{L}^- + \mathcal{L}'^+ > 1$	**VA**, $\vee$**BC** **toCNF**	**SE**, $\vee$**BC** **toCNF**, **CE**	**SE**, $\wedge$**BC**, $\vee$**BC** **toDNF**, **IM**, $\vee$**BC** **toCNF**, **CE**, $\wedge$**BC**

Proof System Basic Statements

$(\bigcap_{X_i \in \mathcal{X}} X_i)[A] \cap \bigcap_{L_i \in \mathcal{L}} \subseteq \bigcup_{L'_i \in \mathcal{L}'} L'_i$ and
$[A](\bigcap_{X_i \in \mathcal{X}} X_i) \cap \bigcap_{L_i \in \mathcal{L}} \subseteq \bigcup_{L'_i \in \mathcal{L}'} L'_i$:

$\mathcal{L}^- + \mathcal{L}'^+ = 0$	**CO**, $\wedge$**BC**, **CL**, **RN**$_\prec$
$\mathcal{L}^- + \mathcal{L}'^+ = 1$	**SE**, $\wedge$**BC**, **CL**, **RN**$_\prec$
$\mathcal{L}^- + \mathcal{L}'^+ > 1$	**SE**, $\vee$**BC**, $\wedge$**BC**, **CL**, **RN**$_\prec$
	toCNF, **CE**, $\wedge$**BC**, **CL**, **RN**$_\prec$

Proof System Basic Statements

$L \subseteq L'$ (mixed):

	$\mathbf{R}$	$\mathbf{R}'$
$\varphi_{\mathbf{R}} \models \psi_{\mathbf{R}'}$ $\neg\psi_{\mathbf{R}'} \models \neg\varphi_{\mathbf{R}}$	**ME**, ns **toDNF** **CE** **ME**	**MO** **IM** **toCNF** **MO**, ns
$\neg\varphi_{\mathbf{R}} \models \psi_{\mathbf{R}'}$ $\neg\psi_{\mathbf{R}'} \models \varphi_{\mathbf{R}}$	**ME**, ns **toCNF** **IM** **MO**, **CT**	**MO**, **CT** **IM** **toCNF** **ME**, ns
$\varphi_{\mathbf{R}} \models \neg\psi_{\mathbf{R}'}$ $\psi_{\mathbf{R}'} \models \neg\varphi_{\mathbf{R}}$	**ME**, ns **toDNF** **CE** **MO**	**MO** **CE** **toDNF** **ME**, ns

M^3	μ^2_0	μ^2_1	μ^2_2	μ^2_3
α^3_0	2	∞	0	∞
α^3_1	1	3	∞	∞

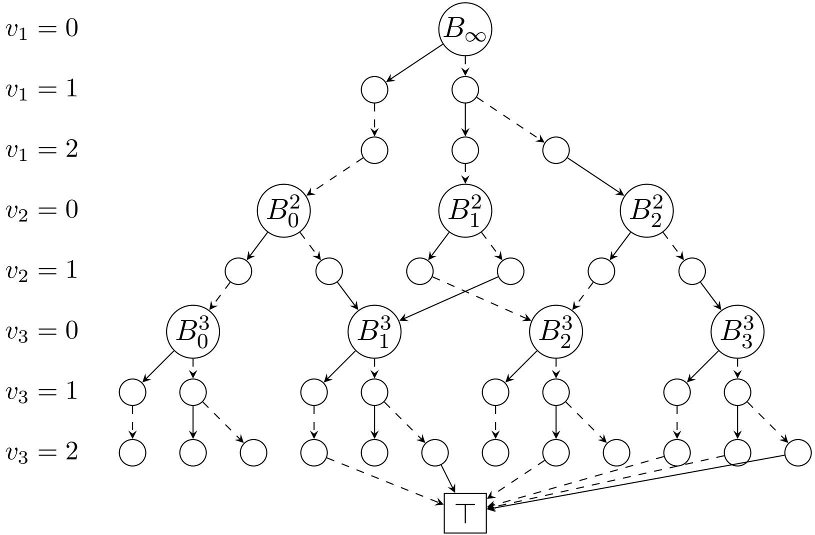
A^3	
$v_3 = 0$	α^3_0
$v_3 = 1$	α^3_1
$v_3 = 2$	α^3_0

M^2	α^1_0	α^1_1	α^1_2
α^2_0	μ^2_0	μ^2_2	μ^2_2
α^2_1	μ^2_1	μ^2_1	μ^2_3

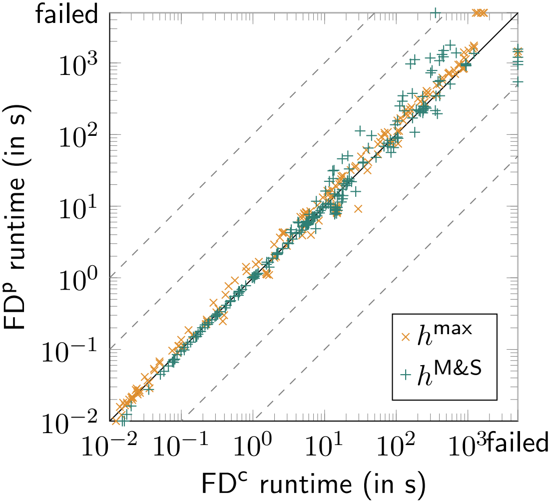
A^2	
$v_2 = 0$	α^2_0
$v_2 = 1$	α^2_1

A^1	
$v_1 = 0$	α^1_0
$v_1 = 1$	α^1_1
$v_1 = 2$	α^1_2

M&S

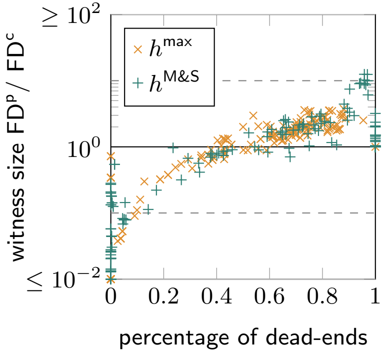

$v_1 = 0$
$v_1 = 1$
$v_1 = 2$
$v_2 = 0$
$v_2 = 1$
$v_3 = 0$
$v_3 = 1$
$v_3 = 2$
B_∞
B^2_0
B^2_1
B^2_2
B^3_0
B^3_1
B^3_2
B^3_3
$\top$

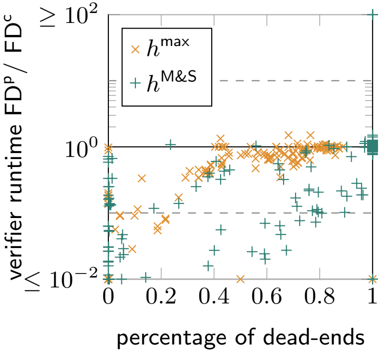
Generation



failed
10^3
10^2
10^1
10^0
10^{-1}
10^{-2}
FD^p runtime (in s)
10^{-2} 10^{-1} 10^0 10^1 10^2 10^3 failed
FD^c runtime (in s)
h^{max}
$h^{M\&S}$

Witness size in relation to dead-ends





Future Work

- cover more planning techniques
  - planning as satisfiability
  - potential heuristics
  - partial order reduction
  - ...
- extend witness definition
  - inductive certificates: more compositions
  - proof system: more rules, more general basic statements