Heuristics for Planning with SAT and Expressive Action Definitions

Jussi Rintanen

June 14, 2011

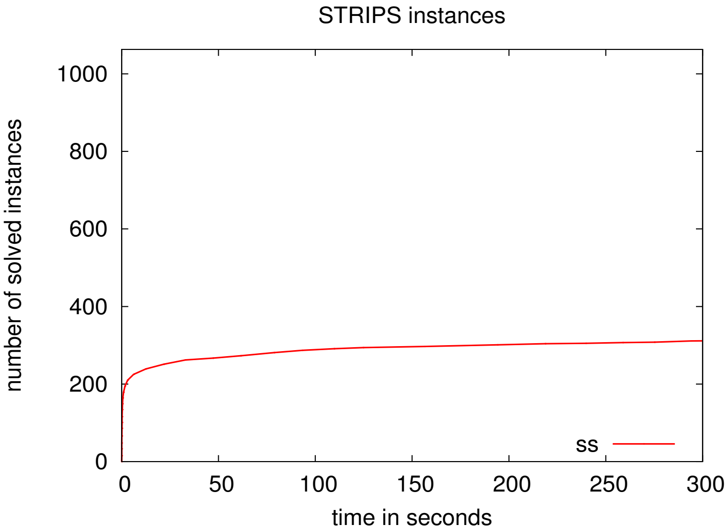
Contribution of the Work

- Earlier work: heuristics for SAT-based planning (classical, non-optimizing), replacing VSIDS et al. in CDCL (Rintanen CP'10, AI'10)
  - A form of backward chaining with CDCL
  - Substantial speed-up for finding plans for most problem types
  - Applicable to almost all notions of plans used with SAT
- This work extends the heuristic.
  - conditional effects: simple change in the encoding scheme
  - disjunctions: requires bigger changes
- Experimental results: outperforms other planners

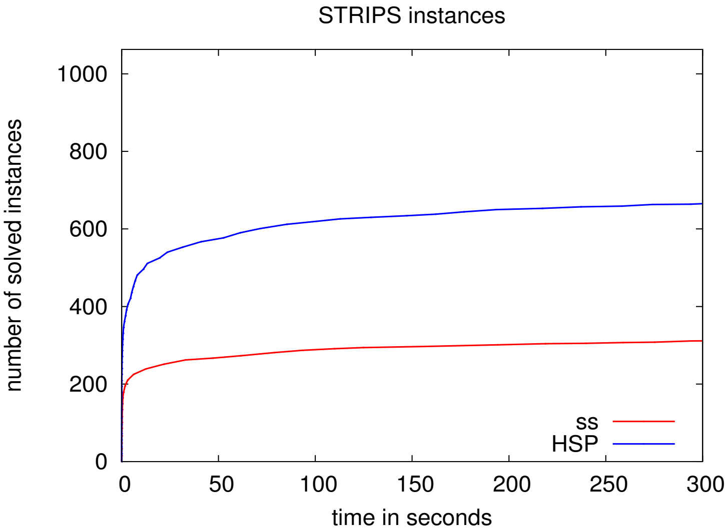
Development of Planning as SAT

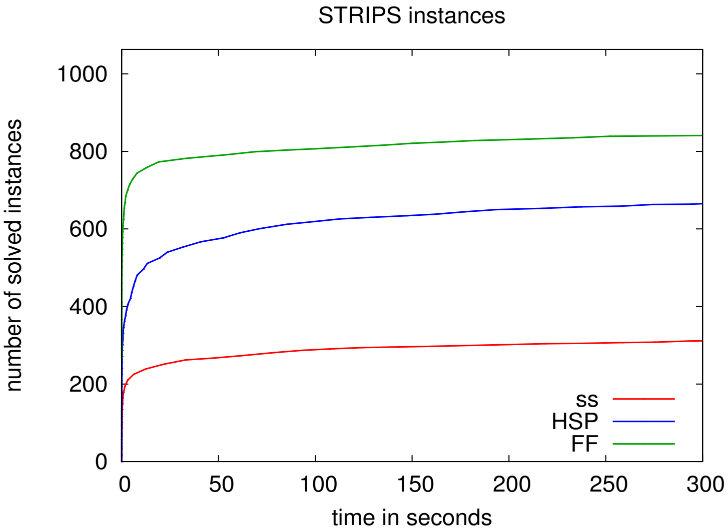
(As relevant to the planning techniques in this work)

1992-99	the approach is first developed	Kautz & Selman etc.
2004-06	practical (linear-size) encodings no more memory overflows	Rintanen et al.
2004-06	interleaved search strategies efficiency close to best planners	Rintanen et al.
2010	planning-specific heuristics for SAT efficiency $\geq$ best planners	Rintanen

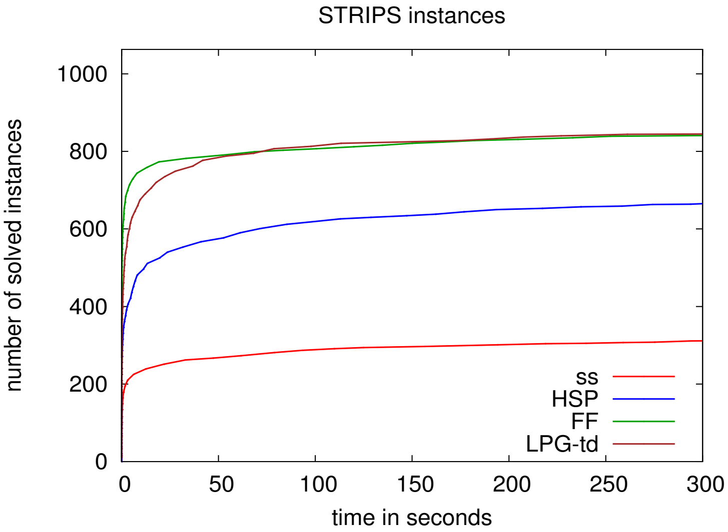


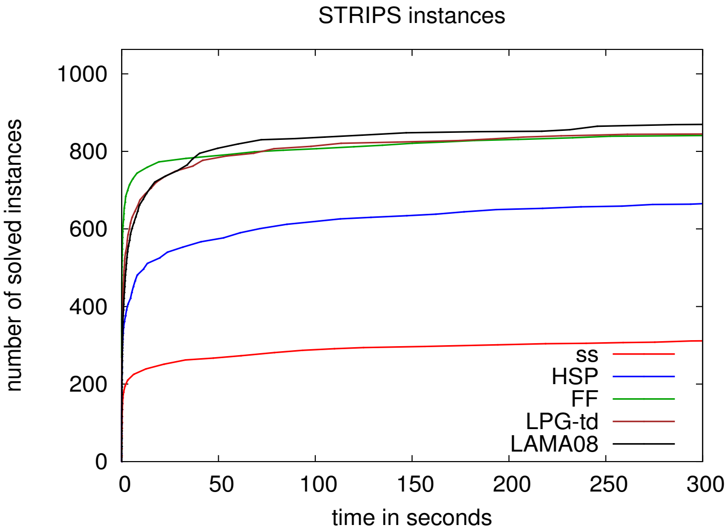
STRIPS instances
number of solved instances
time in seconds
ss
0
200
400
600
800
1000
0
50
100
150
200
250
300

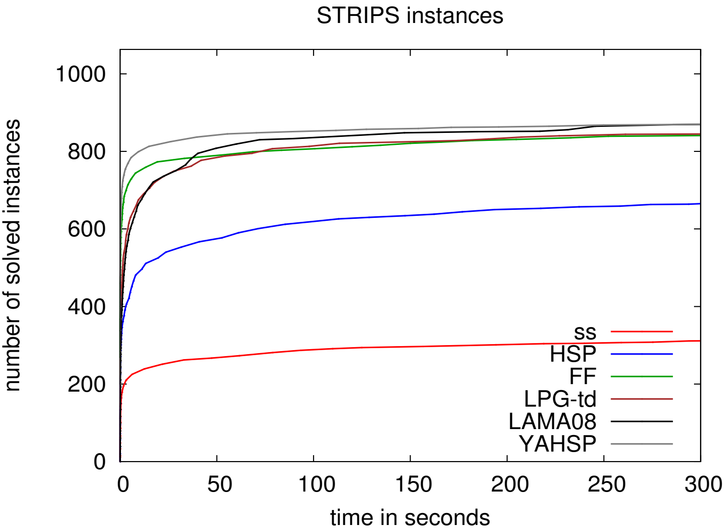

STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP

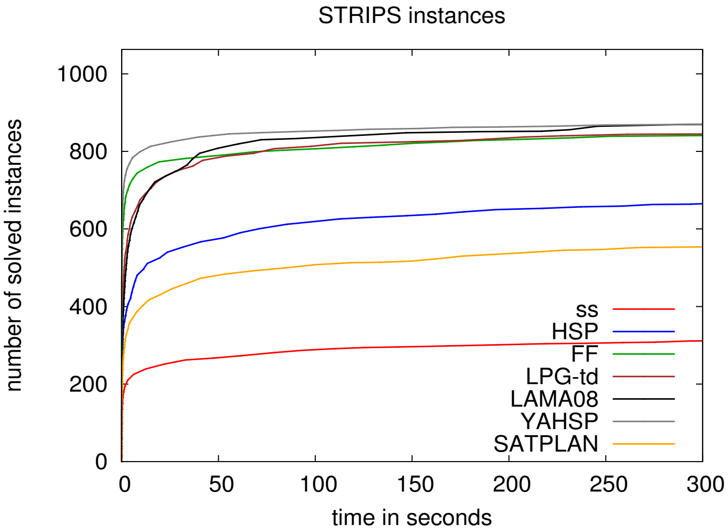


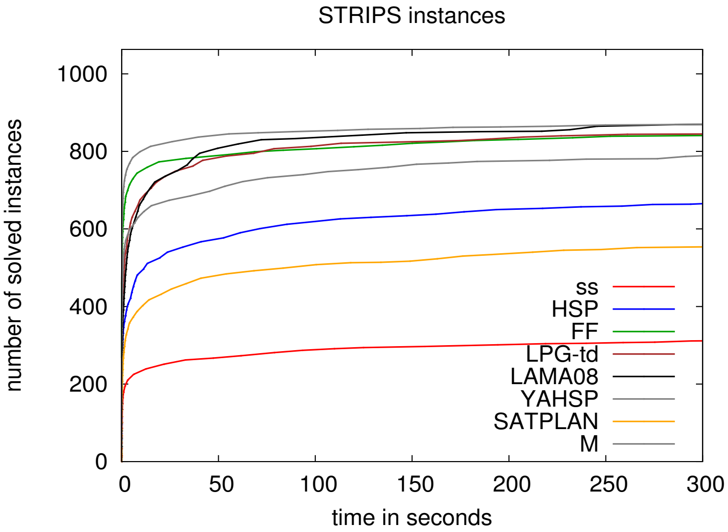
STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF


STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td

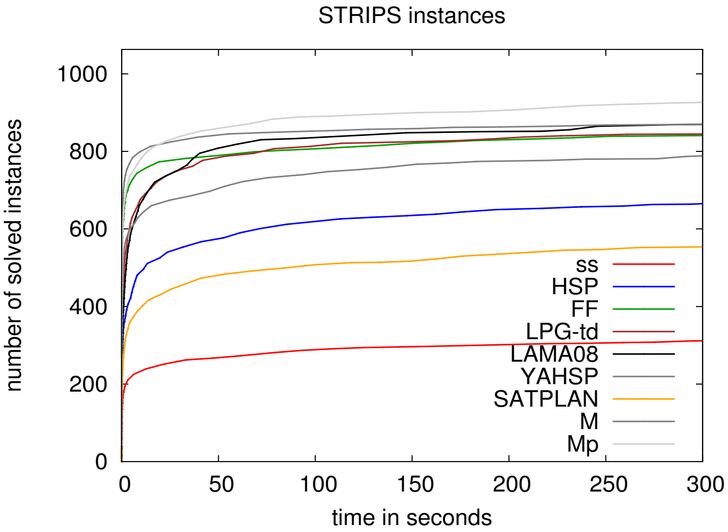

STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td
LAMA08


STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td
LAMA08
YAHSP


STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td
LAMA08
YAHSP
SATPLAN



STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td
LAMA08
YAHSP
SATPLAN
M


STRIPS instances
number of solved instances
time in seconds
0
200
400
600
800
1000
0
50
100
150
200
250
300
ss
HSP
FF
LPG-td
LAMA08
YAHSP
SATPLAN
M
Mp

The new planning heuristic for CDCL

Case 1: goal/subgoal x has no support yet

Value of a state variable x at different time points:

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0			0	0			

The new planning heuristic for CDCL

Case 1: goal/subgoal x has no support yet

Actions that make x true:

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0			0	0			

The new planning heuristic for CDCL

Case 1: goal/subgoal x has no support yet

Actions that make x true as early as possible (at $t-5$):

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0			0	0			

The new planning heuristic for CDCL

Case 1: goal/subgoal x has no support yet

Choose action 2 or 4 at $t-6$ as the next CDCL decision variable.

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0			0	0			

The new planning heuristic for CDCL

Case 2: goal/subgoal x already has support

Goal/subgoal is already made true at $t-4$ by action 4.

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0		1	0	0			

Use precondition literals of action 4 as new subgoals at $t-5$.

The new planning heuristic for CDCL

Case 2: goal/subgoal x already has support

Goal/subgoal is already made true at $t-4$ by action 4.

	$t-8$	$t-7$	$t-6$	$t-5$	$t-4$	$t-3$	$t-2$	$t-1$	t
x	0	0	0		1		1	1	**1**
action 1	0	0	0			0	0	0	
action 2	0	0		0			0		
action 3	0	0	0	0		0	0		
action 4	0	0		1	0	0			

Use precondition literals of action 4 as new subgoals at $t-5$.

Extension to Conditional Effects and Disjunction

- Conditional Effects (without disjunction)
  - Simple change of encoding scheme
- Disjunction
  - Complex subgoals reduced to a set of atomic subgoals.
  - Conceptually a bit more complicated.
  - Need to develop more heuristics for doing this right.

Encoding for Conditional Effects (No Disjunction)

- Idea: View $\langle \phi, \{a \rhd b, d \rhd e\} \rangle$ as interdependent STRIPS actions $\langle \phi \wedge a, b \rangle$ and $\langle \phi \wedge d, e \rangle$.
- New propositional variable for every conditional effect clause.
- The heuristic uses these variables exactly like action variables.
- Interdependencies of conditional effects handled automatically.

Example

Action $\langle \phi, \{\underbrace{a \rhd b}_{x_1}, \underbrace{d \rhd e}_{x_2}\} \rangle$ is translated into

$$o@t \rightarrow \phi@t$$

$$(o@t \wedge a@t) \leftrightarrow x_1@t \qquad (o@t \wedge d@t) \leftrightarrow x_2@t$$

$$x_1@t \rightarrow b@(t+1) \qquad x_2@t \rightarrow e@(t+1)$$

Encoding for Conditional Effects (No Disjunction)

- Idea: View $\langle \phi, \{a \rhd b, d \rhd e\} \rangle$ as interdependent STRIPS actions $\langle \phi \wedge a, b \rangle$ and $\langle \phi \wedge d, e \rangle$.
- New propositional variable for every conditional effect clause.
- The heuristic uses these variables exactly like action variables.
- Interdependencies of conditional effects handled automatically.

Example

Action $\langle \phi, \{\underbrace{a \rhd b}_{x_1}, \underbrace{d \rhd e}_{x_2}\} \rangle$ is translated into

$$
\begin{array}{ll}
o@t \rightarrow \phi@t & \\
(o@t \wedge a@t) \leftrightarrow x_1@t & (o@t \wedge d@t) \leftrightarrow x_2@t \\
x_1@t \rightarrow b@(t+1) & x_2@t \rightarrow e@(t+1)
\end{array}
$$

3-valued states to evaluate goals in

False: l is false (at t).

	$t-3$	$t-2$	$t-1$	t
l	$-$	$-$	$-$	0

True: l is true and there is an explanation for that.

	$t-3$	$t-2$	$t-1$	t
l	$-$	1	$-$	$-$
a	1			

Undetermined: none of the above.

	$t-3$	$t-2$	$t-1$	t
l	0	1	$-$	$-$

Complex Goal Formulas

For a goal Φ, compute atomic subgoals $\{l_1, \ldots, l_n\}$ such that

$$\{l_1, \ldots, l_n\} \models \Phi.$$

- NP-hard to do this.
- NP-hard to minimize n.
- We use an approximation.

Disjunctive Goals and Subgoals (preconditions, conditions)

(We assume NNF here)

For goal Φ and partial state s, compute $\mathsf{subg}_s(\Phi) = \{l_1, \ldots, l_n\}$ (assumption: $s \models \Phi$):

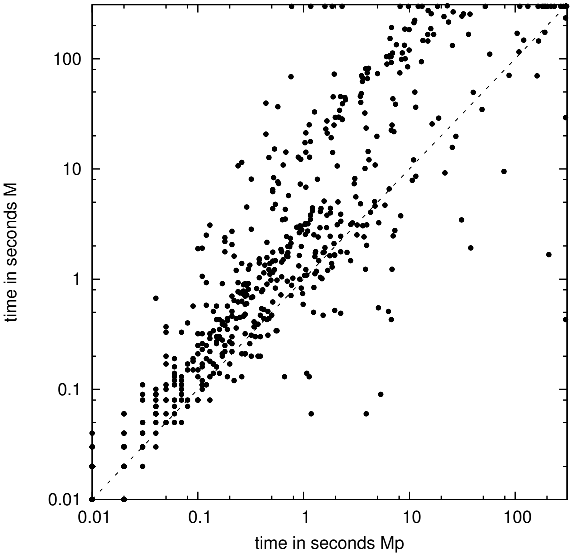
1. $\mathsf{subg}_s(\phi_1 \wedge \phi_1) = \mathsf{subg}_s(\phi_1) \cup \mathsf{subg}_s(\phi_2)$
2. $\mathsf{subg}_s(\phi_1 \vee \phi_1) =$ either $\mathsf{subg}_s(\phi_1)$ or $\mathsf{subg}_s(\phi_2)$, (one that is true in s.)
3. $\mathsf{subg}_s(l) = \{l\}$ if l is Undetermined, $\emptyset$ otherwise.

This is P-time. Heuristics can be used in case 2.

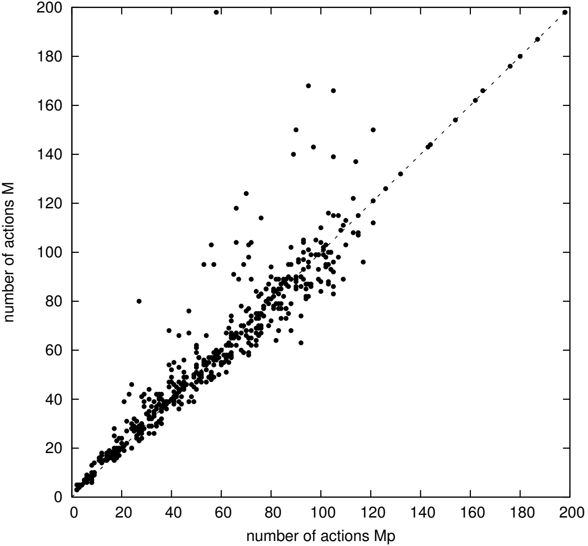
Experiments with non-STRIPS IPC Benchmarks

- Generally a runtime improvement is obtained, similarly to IPC STRIPS benchmarks.
- Both M and Mp outperform LAMA, FF. (Other well-known and efficient planners (e.g. YAHSP, LPG) only do STRIPS.)

Impact on Runtimes



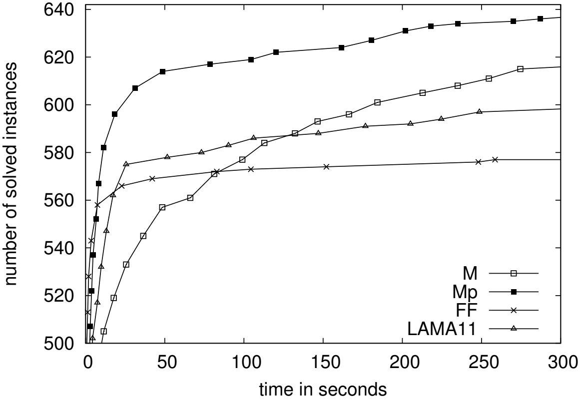
Impact on Plan Sizes



Comparison by Domain

		Mp	M	LAMA11	FF
1998-ASSEMBLY-ADL	24	23	18	24	24
2000-ELEVATOR-FULL	143	138	139	135	132
2000-ELEVATOR-SIMPLE	150	150	150	150	150
2000-SCHEDULE-ADL	150	149	144	150	150
2002-SATELLITE-ADL	20	20	20	20	20
2004-AIRPORT-ADL	50	26	21	33	21
2004-OPTICAL-TELEGRAPH-ADL	48	22	17	2	11
2004-PHILOSOPHERS-ADL	48	48	48	12	12
2006-PATHWAYS-ADL	30	30	30	29	16
2006-TRUCKS-ADL	29	15	14	14	11
2008-OPENSTACKS-ADL	30	15	14	30	30
total	722	636	615	599	577
weighted score	11	8.91	8.41	8.35	7.73

Comparison by Number of Instances Solved



Conclusions

What we have done so far:

- We presented variable selection heuristics for planning within the CDCL framework, for general PDDL actions.
- As with STRIPS, this beats other planners by a clear margin.

Future work:

- Combine this with VSIDS to do still better.
- Try with Bounded LTL Model-Checking, Discrete Event Systems diagnosis, ...