University of Basel

Combining Novelty-Guided and Bounded Suboptimal Search

Gian-Andrea Wetten <gian-andrea.wetten@stud.unibas.ch>

Department of Mathematics and Computer Science, University of Basel

September 11th 2017

Classical Planning

- Find a series of actions which lead from the initial state to a goal state
- *Examples:*
  - Tidybot
  - Blocks world

Best-First Search

- Forward state space search
- Expanding the most promising node first
- *Examples:*
  - GBFS: $f(n) = h(n)$
  - A^*: $f(n) = g(n) + h(n)$

Novelty

Definition

The **novelty** $w(s)$ of a state s is i iff there is a tuple t of i atoms such that s is the first state in the search that makes all the atoms in t true, and no tuple of smaller size has this property.

Examples: IW, IW+, SIW

Novelty

- List of subsets
- List of offsets
- Lookup table for encountered partial states

Novelty

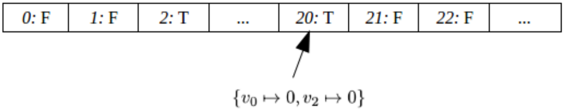
Is the novelty of the state $s = \{v_0 \mapsto 2, v_1 \mapsto 4, v_2 \mapsto 1, v_3 \mapsto 0\}$ within our novelty bound of 2?

- Variables: $V = \{v_0, v_1, v_2, v_3\}$
- Variable domain sizes: $|D_{v_0}| = 4, |D_{v_1}| = 5, |D_{v_2}| = 3, |D_{v_3}| = 1$

Novelty

Is the novelty of the state $s = \{v_0 \mapsto 2, v_1 \mapsto 4, v_2 \mapsto 1, v_3 \mapsto 0\}$ within our novelty bound of 2?

- Variables: $V = \{v_0, v_1, v_2, v_3\}$
- Variable domain sizes: $|D_{v_0}| = 4, |D_{v_1}| = 5, |D_{v_2}| = 3, |D_{v_3}| = 1$
- Subsets: *subs* $= \langle\{v_0, v_1\}, \{v_0, v_2\}...\{v_2, v_3\}\rangle$
- Offsets: *off* $= \{0, 20, 32, 36, 51, 56\}$



Novelty

Is the novelty of the state $s = \{v_0 \mapsto 2, v_1 \mapsto 4, v_2 \mapsto 1, v_3 \mapsto 0\}$ within our novelty bound of 2?

Example

- Subset: $s_u = \{v_1, v_2\}$ produces $s^+ = \{v_1 \mapsto 4, v_2 \mapsto 1\}$
- *off*$(s_u) = 36$
- *off*$^*(s^+) = 36 + s[v_1] \cdot |D_{v_2}| + s[v_2] = 49$

Yes if cell 49 in *encountered_states* holds *false*

Novelty

Is the novelty of the state $s = \{v_0 \mapsto 2, v_1 \mapsto 4, v_2 \mapsto 1, v_3 \mapsto 0\}$ within our novelty bound of 2?

Example

- Subset: $s_u = \{v_1, v_2\}$ produces $s^+ = \{v_1 \mapsto 4, v_2 \mapsto 1\}$
- *off*$(s_u) = 36$
- *off*$^*(s^+) = 36 + s[v_1] \cdot |D_{v_2}| + s[v_2] = 49$

Yes if cell 49 in *encountered_states* holds *false*

Continue iterating through all remaining subsets

Bounded Suboptimal Search

Motivation

- Reduce memory requirements
- Reduce search time
- Bind solution cost

Bounded Suboptimal Search

Earlier approaches

- Weighted A^*: $f'(n) = g(n) + w \cdot h(n)$
- Optimistic Search
- A^*_ϵ
- Explicit Estimation Search

Bounded Suboptimal Search

A^*_ϵ:

- *open* list: sorted on $f(n)$
- *focal* list: nodes with $f(n) \leq w \cdot f(\textit{best}_f)$ sorted on distance-to-go estimator $\hat{d}(n)$
- Thrashing problem leads to poor performance

Bounded Suboptimal Search

Explicit Estimation Search (EES):

- *open* list: sorted on $\hat{f}(n)$
- *cleanup* list: sorted on $f(n)$
- *focal* list: nodes with $\hat{f}(n) \leq w \cdot \hat{f}(best_{\hat{f}})$ sorted on distance-to-go estimator $\hat{d}(n)$

Bounded Suboptimal Search

Explicit Estimation Search (EES):

selectNode

1. **if** $\hat{f}(\textit{best}_{\hat{d}}) \leq w \cdot f(\textit{best}_f)$ **then** $\textit{best}_{\hat{d}}$
2. **else if** $\hat{f}(\textit{best}_{\hat{f}}) \leq w \cdot f(\textit{best}_f)$ **then** $\textit{best}_{\hat{f}}$
3. **else** $\textit{best}_f$

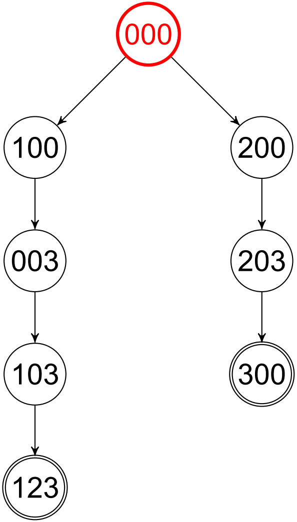
$\hat{h}(n) \geq h(n)$

Combination

Combining Novelty-Guided and Bounded Suboptimal Search:

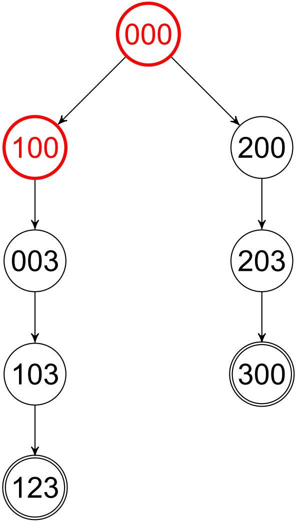
- $w_h(s)$: one lookup table per unique heuristic value h(s)
- EES with *focal* sorted on w_h instead of $\hat{d}$
- Additional inclusion restriction for *focal*: $w_h(s) \leq$ *nov*$_b$

Example



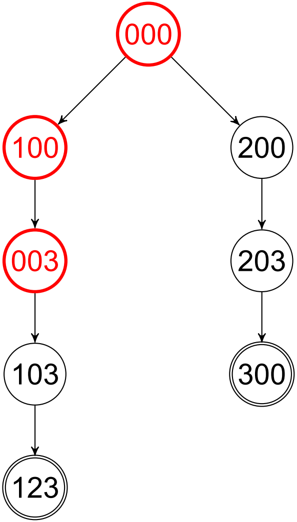
s	w	$\hat{f}$	f
100	1	4	4
200	1	5	3

Example



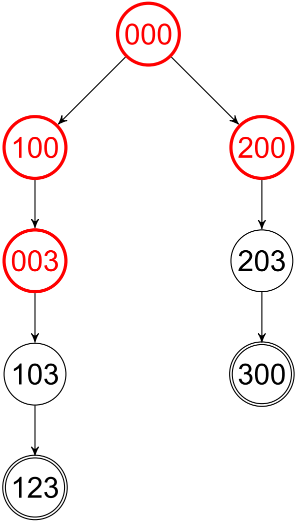
s	w	$\hat{f}$	f
100	1	4	4
200	1	5	3
003	1	4	4

Example



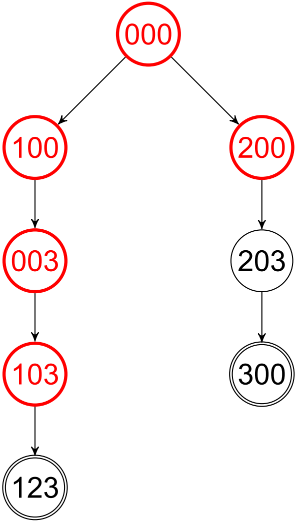
s	w	$\hat{f}$	f
100	1	4	4
200	1	5	3
003	1	4	4
103	2	4	4

Example



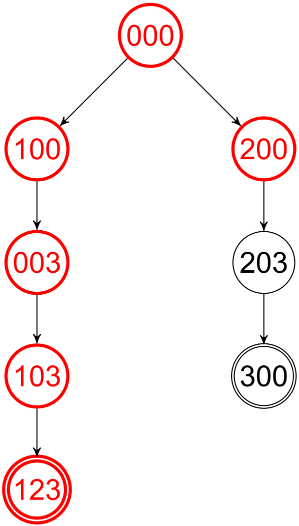
s	w	$\hat{f}$	f
100	1	4	4
200	1	5	3
003	1	4	4
103	2	4	4
203	2	5	3

Example



s	w	$\hat{f}$	f
100	1	4	4
200	1	5	3
003	1	4	4
103	2	4	4
203	2	5	3
123	2	4	4

Example



Evaluation

- Implemented in the Fast Downward planning system (C++)
- 2GB memory limit
- Three minute time limit

Evaluation

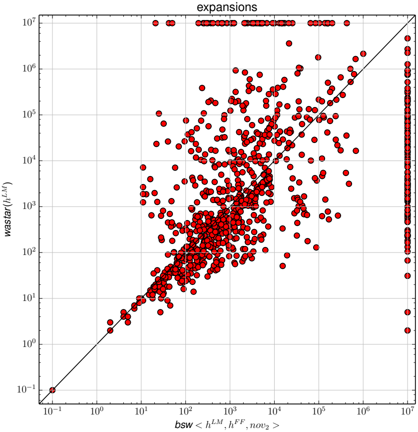
- Picking a novelty bound
- Effect of the weighting parameter
- Heuristics used:
  - $h = h^{LM}$
  - $\hat{h} = h^{FF}$
  - $w_{h^{FF}}(s)$

Evaluation

Summary of the search attributes of the runs using *bsw* with different novelty values

Summary	bsw($n_b = 1$)	bsw($n_b = 2$)	bsw($n_b = 3$)	wA^*
Plan length - Sum	26962	**26936**	26945	27348
Memory - Sum	**8337552**	8491108	20315652	10079064
Generated - Geometric mean	9383.61	9379.55	**9375.64**	12664.49
Expansions - Geometric mean	**1098.68**	1100.06	1099.75	1493.06
Coverage - Sum	947	947	882	**1027**
Search time - Geometric mean	0.80	0.85	1.44	**0.25**

Evaluation



Comparison of expansions between bsw($n_n = 2$) and weighted A^* with $w = 2$

Evaluation

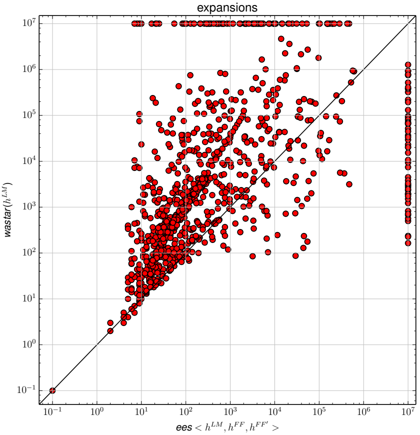
	$w = 1.5$	$w = 2.0$		$w = 3.0$	
Coverage	bsw($n_b = 2$)	bsw($n_b = 2$)	weighted A^*	bsw($n_b = 2$)	weighted A^*
Nr. of problems 1667	869	947	1027	1008	**1071**

Evaluation

Summary of the search attributes for the runs using the implementations of the search engine components separately

Summary	wA^*	tie$<$w$A^*, nov_2>$	ees
Plan length - Sum	**35999**	37096	36504
Memory - Sum	14876532	15126440	**9523920**
Generated - Geometric mean	20926.90	19148.35	**2576.71**
Expansions - Geometric mean	2360.67	2164.33	**298.77**
Coverage - Sum	1027	1029	**1070**
Search time - Geometric mean	0.46	**0.37**	0.47

Evaluation



Comparison of expansions between ees and weighted A^* with $w = 2$

Conclusion

- Tendency of a lower number of expanded and generated nodes
- Higher mean search time
- Lower coverage
- Good results with pure EES

Questions?