On Variable Dependencies and Compressed Pattern Databases

Malte Helmert[1] Nathan Sturtevant[2] Ariel Felner[3]

[1]University of Basel, Switzerland
[2]University of Denver, USA
[3]Ben Gurion University, Israel

SoCS 2017

Introduction

Quotation

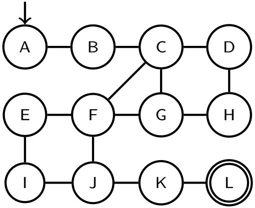
previous work on compressed pattern databases:

Sturtevant, Felner and Helmert (SoCS 2014)

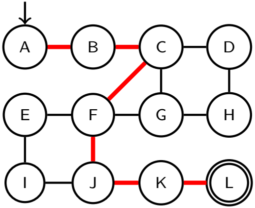
"This approach worked very well for the 4-peg Towers of Hanoi, for instance, but its success for the sliding tile puzzles was limited and no significant advantage was reported for the Top-Spin domain (Felner et al., 2007)."

this paper: try to understand why

Compressed PDBs

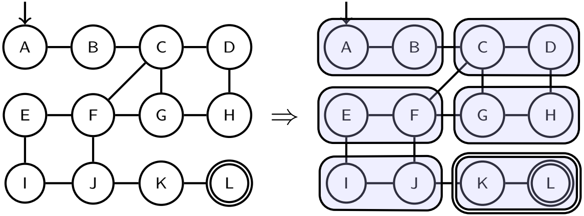


Compressed PDBs



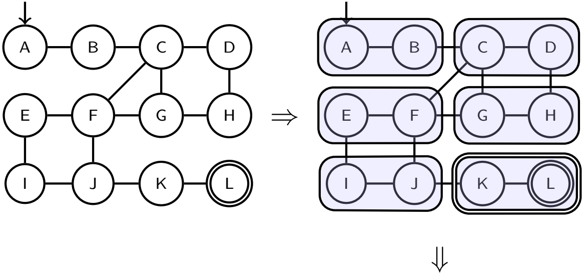
$$h^*(\mathsf{A}) = 6$$

Compressed PDBs



$$h^*(A) = 6$$

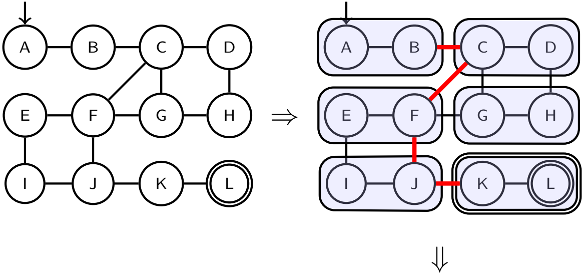
Compressed PDBs



$h^*(A) = 6$

AB	4
CD	3
EF	2
GH	3
IJ	1
KL	0

Compressed PDBs

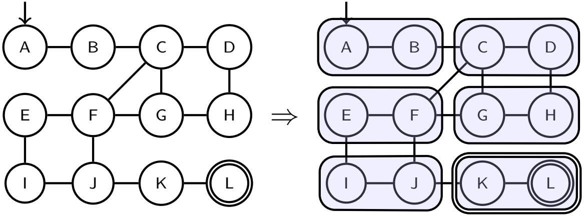


$$h^*(A) = 6$$

$$h_{\mathrm{PDB}}(A) = 4$$

AB	4
CD	3
EF	2
GH	3
IJ	1
KL	0

Compressed PDBs



⇓

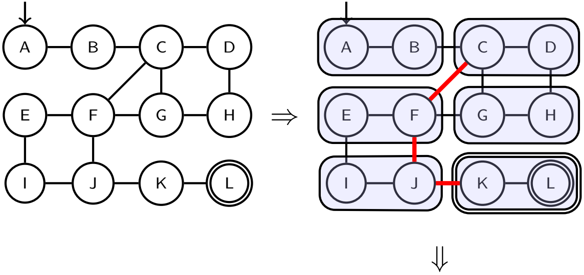
$$h^*(\mathrm{A}) = 6$$

$$h_{\mathrm{PDB}}(\mathrm{A}) = 4$$

$$h^{comp}_{\mathrm{PDB}}(\mathrm{A}) = 3$$

AB	4	3
CD	3	
EF	2	2
GH	3	
IJ	1	0
KL	0	

Compressed PDBs



$$h^*(\mathsf{A}) = 6$$

$$h_{\mathsf{PDB}}(\mathsf{A}) = 4$$

$$h^{comp}_{\mathsf{PDB}}(\mathsf{A}) = 3$$

AB	4	3
CD	3	
EF	2	2
GH	3	
IJ	1	0
KL	0	

Comparing PDBs to Compressed PDBs

Assume we have N units of memory.

Consider three heuristics:

- h_F: fine-grained PDB ($M \gg N$ entries)
- h_F^{comp}: compressed fine-grained PDB (N entries)
- h_C: coarse-grained PDB (N entries)

Which one should we use, h_F^{comp} or h_C?

Experimental Results

State Space	M/N	h_F	h_F^{comp} MOD	h_F^{comp} DIV	h_F^{comp} random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Experimental Results

State Space	M/N	h_F	h_F^{comp} MOD	h_F^{comp} DIV	h_F^{comp} random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

h_F^{comp} better than h_C on average

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Experimental Results

State Space	M/N	h_F	h_F^{comp} MOD	h_F^{comp} DIV	h_F^{comp} random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

h_F^{comp} worse than h_C on average

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Experimental Results

State Space	M/N	h_F	h_F^{comp} MOD	h_F^{comp} DIV	h_F^{comp} random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

h_F^{comp} equal to h_C on average

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Good News

Dominance of Compressed PDBs

Theorem (dominance of compressed PDBs)

Let h_F and h_C be heuristics such that h_F is a refinement of h_C.

Consider compressed heuristics with a compression regime that is compatible with h_F and h_C.

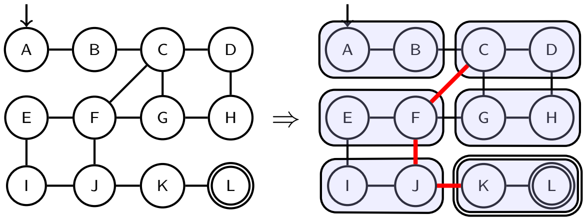
Then

$$h_F^{comp}(s) \geq h_C(s)$$

for all states s.

informally: compression step applies further abstraction on top of the abstraction h_F

Dominance of Compressed PDBs: Proof Idea

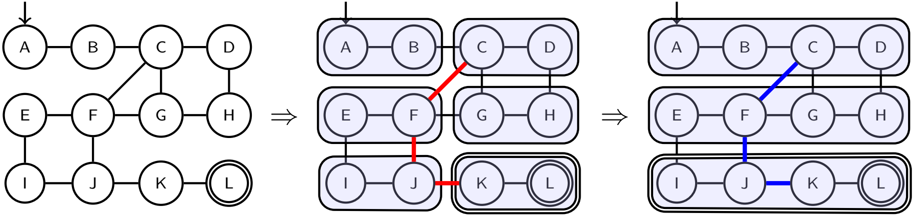


⇓

$$h^*(\mathrm{A}) = 6$$
$$h_F(\mathrm{A}) = 4$$
$$h_F^{comp}(\mathrm{A}) = 3$$

AB	4	3
CD	3	
EF	2	2
GH	3	
IJ	1	0
KL	0	

Dominance of Compressed PDBs: Proof Idea



$$h^*(\text{A}) = 6$$
$$h_F(\text{A}) = 4$$
$$h_F^{comp}(\text{A}) = 3$$

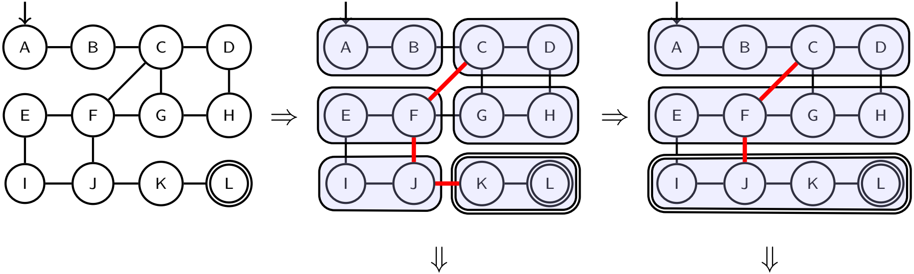
⇓

AB	4	3
CD	3	
EF	2	2
GH	3	
IJ	1	0
KL	0	

⇓

AB CD	2
EF GH	1
IJ KL	0

Dominance of Compressed PDBs: Proof Idea



$$h^*(A) = 6$$
$$h_F(A) = 4$$
$$h_F^{comp}(A) = 3$$
$$h_C(A) = 2$$

AB	4	3
CD	3	
EF	2	2
GH	3	
IJ	1	0
KL	0	

AB CD	2
EF GH	1
IJ KL	0

Dominance of Compressed PDBs: Experimental Results

			h_F^{comp}			
State Space	M/N	h_F	MOD	DIV	random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Dominance of Compressed PDBs: Experimental Results

State Space	M/N	h_F	h_F^{comp} MOD	h_F^{comp} DIV	h_F^{comp} random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

$h_F^{comp}(s) \geq h_C(s)$ for all states according to the theorem

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Bad News

State Variables

States are described in terms of state variables.

Examples:

- Towers of Hanoi: position of one disk
- sliding tiles: position of a tile (or blank)
- TopSpin: position of a token

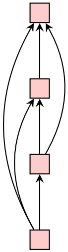
PDBs project to a subset of variables (the "pattern").

Variable Dependencies

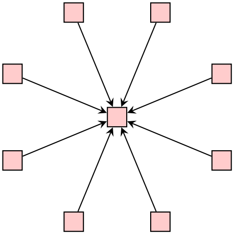
Variable u depends on variable v if changing u is conditioned in any way on v.

Variable Dependencies

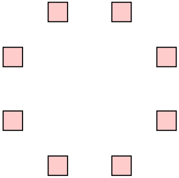
Variable u depends on variable v if changing u is conditioned in any way on v.



Towers of Hanoi



sliding tiles



TopSpin

Improvements vs. Dependencies

Theorem (no improvements without dependencies)

Consider the patterns $F \supseteq C$ in an undirected state space.

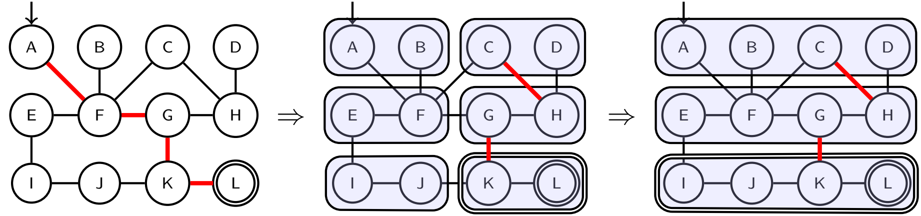
Let h_F^{comp} be a compressed PDB heuristic with a compression regime compatible with the refinement relation between F and C.

If no variable in C depends on any variable in $F \setminus C$, then

$$h_F^{comp}(s) = h_C(s)$$

for all states s.

Improvements vs. Dependencies: Proof Idea



$$h^*(\mathsf{A}) = 4$$
$$h_F(\mathsf{A}) = 3$$
$$h_F^{comp}(\mathsf{A}) = 2$$
$$h_C(\mathsf{A}) = 2$$

AB	3	2
CD	2	
EF	2	1
GH	1	
IJ	1	0
KL	0	

AB CD	2
EF GH	1
IJ KL	0

Improvements vs. Dependencies: Experimental Results

			h_F^{comp}			
State Space	M/N	h_F	MOD	DIV	random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Improvements vs. Dependencies: Experimental Results

			h_F^{comp}			
State Space	M/N	h_F	MOD	DIV	random	h_C
Hanoi	4	104.32	87.04	103.76	90.08	87.04
Sliding Tiles A	10	34.99	29.89	32.08	26.38	32.08
Sliding Tiles B	10	34.99	30.50	32.84	26.38	15.29
TopSpin	12	10.78	9.29	9.59	8.73	9.59

$h_F^{comp}(s) = h_C(s)$ for all states according to the theorem

- Hanoi: 4 pegs and 16 disks; pattern with 15 disks
- Sliding Tiles A: 4×4 puzzle; pattern $\langle \text{blank}, 1, 2, 3, 4, 5, 6 \rangle$
- Sliding Tiles B: 4×4 puzzle; pattern $\langle 6, 5, 4, 3, 2, 1, \text{blank} \rangle$
- TopSpin: 18 tokens and turnstile size 4; pattern with 7 tokens

all use lexicographic ranking

Related Work in Classical Planning

our result:

- $h_F^{comp} = h_C$
- for undirected state spaces
- under certain dependency conditions

Related Work in Classical Planning

our result:

- $h_F^{comp} = h_C$
- for undirected state spaces
- under certain dependency conditions

literature (Haslum et al. 2007; Pommerening et al. 2013):

- $h_F = h_C$
- for arbitrary state spaces
- under certain (different) dependency conditions

neither result entails the other

$\rightsquigarrow$ many more details in paper

Conclusion

Conclusion

When is entry compression a good idea?

- never bad when compatible with refinement
- never good when refinement does not capture a dependency

What does this mean for the benchmarks?

- Towers of Hanoi: must compress smaller disks away
- sliding tile: compressing blank the only useful refinement
- TopSpin: no dependencies, hence no gain
  (ditto: Pancakes, Rubik's Cube)

Thank You

Thank you for your attention!