Building a Heuristic for Greedy Search

Christopher Wilt and Wheeler Ruml

University of New Hampshire

Department of Computer Science

Grateful thanks to NSF for support.

This Talk is About Suboptimal Search

Greedy Best-First Search (GBFS)

1. best-first on $h(n)$, no $g(n)$. maintains open list.
2. very important in applications (eg planning)
3. building block for anytime algorithms
4. heuristic is crucial

This Talk is About Suboptimal Search

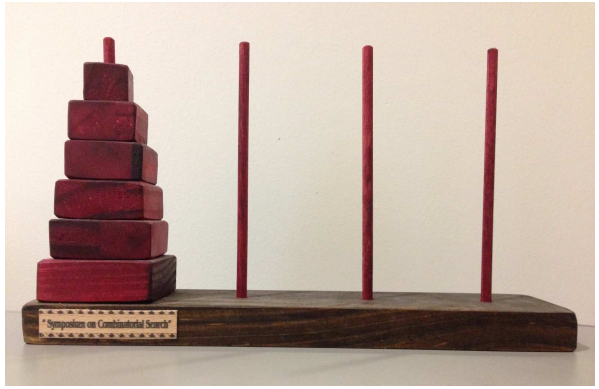
Greedy Best-First Search (GBFS)

1. best-first on $h(n)$, no $g(n)$. maintains open list.
2. very important in applications (eg planning)
3. building block for anytime algorithms
4. heuristic is crucial

Two central points:

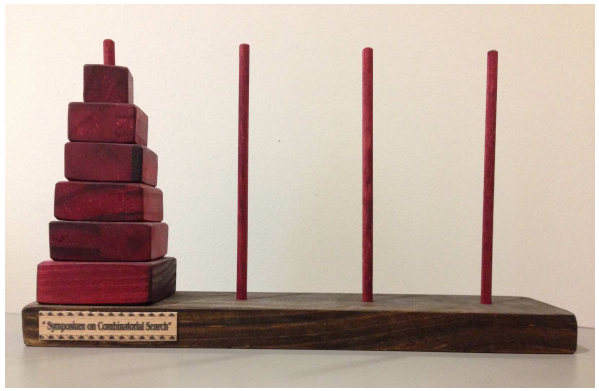
1. suboptimal is different!
   heuristics for optimal search can be inappropriate
2. not hard to do better
   Goal Distance Rank Correlation (GDRC)

Example 1/3: 12-Disk 4-Peg Towers of Hanoi



8+4 PDBs dominates 8+0, right?

Example 1/3: 12-Disk 4-Peg Towers of Hanoi



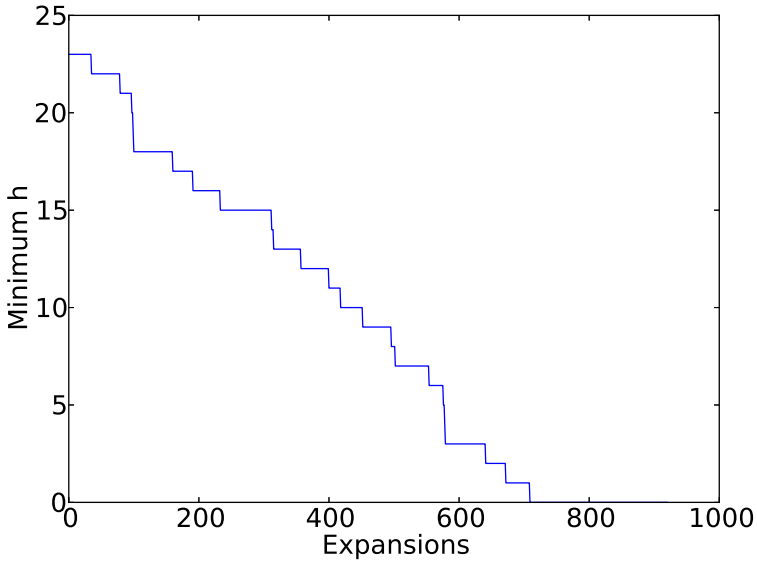
8+4 PDBs dominates 8+0, right? Yes, but...

heuristic	node expansions A*	GBFS
8+4 PDBs	2,153,558	36,023
8+0 PDB	4,618,913	771

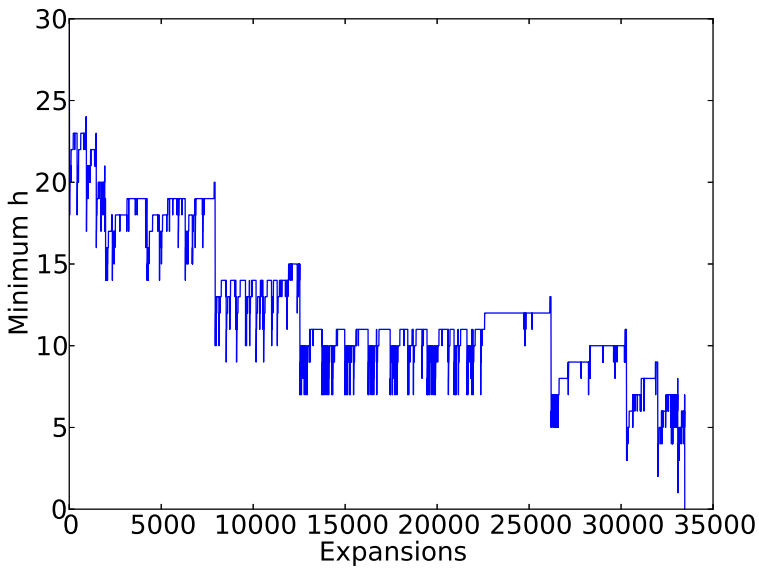
better for A* ≠ better for GBFS

GBFS Behavior on Towers of Hanoi

8 PDB



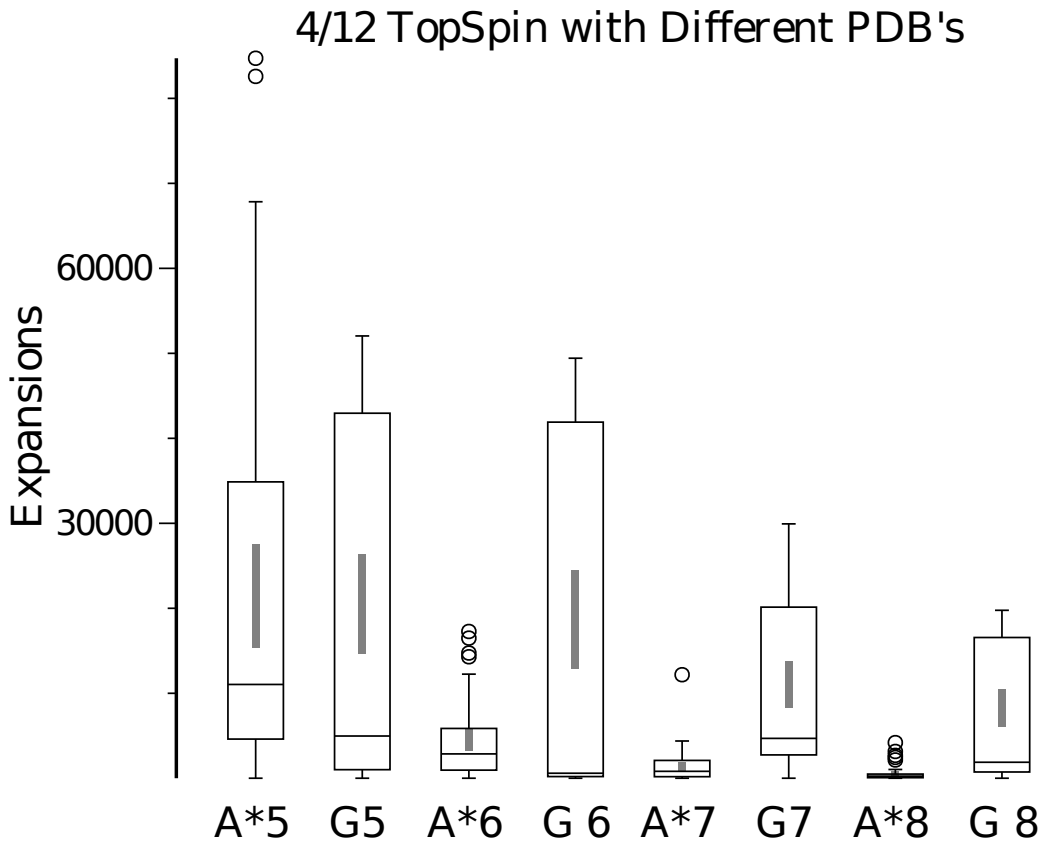
8+4 PDBs



- no local minima
- 256 states at each h value

- low h states can be far from goal
- want 'serialized h values'

Example 2/3: 12-Token 4-Turnstile TopSpin



many disconnected $h = 0$ regions

Example 3/3: 3×4 Sliding Tile Puzzle

outer L

	A	A	3
A	A	A	7
8	9	10	11

one $h = 0$ region

checkered

	1	A	3
4	A	6	A
A	9	A	11

same size

Example 3/3: 3×4 Sliding Tile Puzzle

outer L

	A	A	3
A	A	A	7
8	9	10	11

checkered

	1	A	3
4	A	6	A
A	9	A	11

one $h = 0$ region

same size

abstraction	node expansions GBFS	A*
outer L	258	1,251,260
checkered	11,583	1,423,378
average 6-tile PDB	17,641	1,596,041
worst 6-tile PDB (for GBFS)	193,849	1,911,566

GBFS is sensitive to h

GBFS wants serialized h values

Summary So Far

Two central points:

1. suboptimal is different!
   heuristics for optimal search can be inappropriate
2. not hard to do better
   Goal Distance Rank Correlation (GDRC)

Goal Distance Rank Correlation (GDRC)

goal distance: number of steps to the goal
rank correlation: how well orderings line up

$$GDRC(h) = \tau(h, d^*)$$

can use all nodes or just a sample near the goal

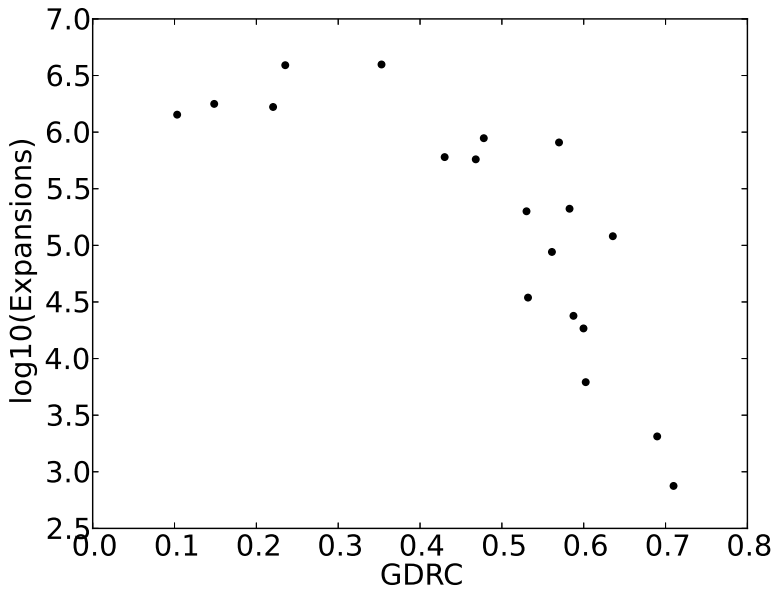
Goal Distance Rank Correlation (GDRC)

goal distance: number of steps to the goal
rank correlation: how well orderings line up

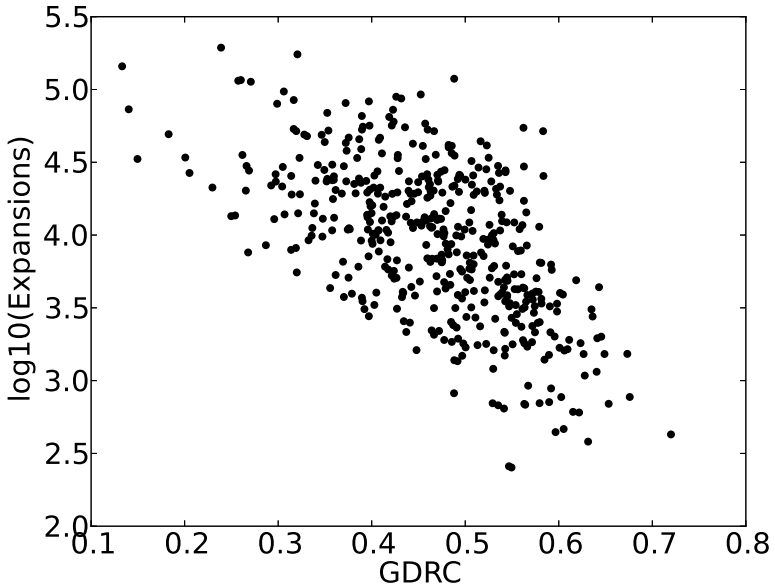
$$GDRC(h) = \tau(h, d^*)$$

can use all nodes or just a sample near the goal

Hanoi



3×4 Puzzle



Building a Heuristic for GBFS Using GDRC

example: simple hill-climbing in PDB-space

start by abstracting everything
loop until done:
  consider all refinements
  pick one with highest GDRC

Performance of GDRC-Built Heuristics

TopSpin with unit costs: good for both GBFS and A*

TopSpin with some heavy tokens:

Performance of GDRC-Built Heuristics

TopSpin with unit costs: good for both GBFS and A*

TopSpin with some heavy tokens:

	node expansions		
PDB	GBFS	A*	avg h value
GDRC-built (contiguous)	411	10,607	52
built for A* (heavy)	961	411	94
random	2,387	26,017	48

Performance of GDRC-Built Heuristics

TopSpin with unit costs: good for both GBFS and A*

TopSpin with some heavy tokens:

PDB	node expansions GBFS	A*	avg h value
GDRC-built (contiguous)	411	10,607	52
built for A* (heavy)	961	411	94
random	2,387	26,017	48

Sliding Tiles:

abstraction	GBFS	A*
outer L	258	1,251,260
checkered	11,583	1,423,378
average 6-tile PDB	17,641	1,596,041
instance-specific	8,530	480,250
GDRC-built (4th/462)	427	1,197,789

GDRC-built heuristics seem well-tuned for GBFS!

Conclusions

suboptimal is different!

- more sensitive to h
- better h for optimal can be worse for greedy
- needs its own theory and methods

not hard to do better!

Goal Distance Rank Correlation (GDRC)

- estimates seem effective
- can search space of PDBs

this area is wide open!


UNIVERSITY of NEW HAMPSHIRE
©2011 Mike Ross, UNH Photographic Services

Extra Slides

More results

foo