Pattern Database Heuristics for Greedy Search

Carmen St. Jean

Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Algorithm: Greedy Search

Assume b is branching factor and m is maximum depth.

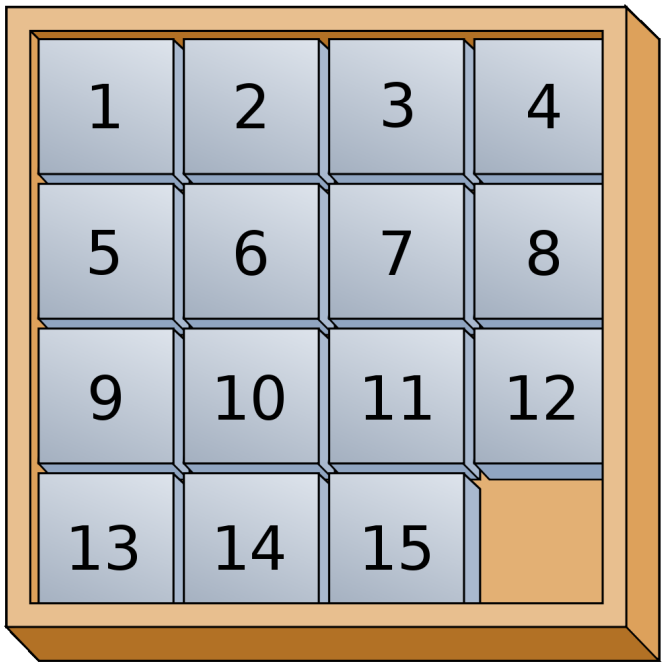
- Best-first search with queue ordered by heuristic value
- Complete in finite spaces
- Inadmissible
- b^m time
- b^m space

Algorithm: Greedy Search

Assume b is branching factor and m is maximum depth.

- Best-first search with queue ordered by heuristic value
- Complete in finite spaces
- Inadmissible
- b^m time
- b^m space
- **Tends to yield suboptimal solutions in a reasonable time**

Domain: N-Puzzle



- N numbered square tiles, one blank tile
- Objective: rearrange tiles by sliding the blank space to reach goal configuration
- Commonly solved with A* using Manhattan Distance heuristic

Pattern Databases

- A *pattern* is a partial specification of a state

Pattern Databases

- A *pattern* is a partial specification of a state
  - Some elements of the state are abstracted

Pattern Databases

- A *pattern* is a partial specification of a state
  - Some elements of the state are abstracted
- A *target pattern* is a partial specification of the goal state

Pattern Databases

- A *pattern* is a partial specification of a state
  - Some elements of the state are abstracted
- A *target pattern* is a partial specification of the goal state
- A *pattern database* is set of all patterns obtained by permuting the target pattern

Pattern Databases

- A *pattern* is a partial specification of a state
  - Some elements of the state are abstracted
- A *target pattern* is a partial specification of the goal state
- A *pattern database* is set of all patterns obtained by permuting the target pattern
- Every pattern knows its exact solution cost for the target pattern

Pattern Databases

- A *pattern* is a partial specification of a state
  - Some elements of the state are abstracted
- A *target pattern* is a partial specification of the goal state
- A *pattern database* is set of all patterns obtained by permuting the target pattern
- Every pattern knows its exact solution cost for the target pattern
  - Admissible heuristic

N-Puzzle PDBs

- Abstract away some tiles

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance
- Can use multiple disjoint pattern databases at once

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance
- Can use multiple disjoint pattern databases at once
- Lots of different abstractions possible

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance
- Can use multiple disjoint pattern databases at once
- Lots of different abstractions possible
  - Fringe abstraction (outer edge)
  - Special abstraction (keep tiles furthest from goal position)

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance
- Can use multiple disjoint pattern databases at once
- Lots of different abstractions possible
  - Fringe abstraction (outer edge)
  - Special abstraction (keep tiles furthest from goal position)
- Less tiles abstracted, more powerful pattern database

N-Puzzle PDBs

- Abstract away some tiles
- Can be more accurate than Manhattan distance
- Can use multiple disjoint pattern databases at once
- Lots of different abstractions possible
  - Fringe abstraction (outer edge)
  - Special abstraction (keep tiles furthest from goal position)
- Less tiles abstracted, more powerful pattern database
- More timely to calculate and larger space required when fewer tiles are abstracted

Fringe Abstraction

Independent of start configuration.

Fringe Abstraction

Independent of start configuration.

	1	2	3
4	5	6	7
8	9	10	11

Fringe Abstraction

Independent of start configuration.

	1	2	3
4	5	6	7
8	9	10	11

	A	2	3
A	A	6	7
A	A	A	A

	A	A	A
A	A	A	A
8	9	10	11

Fringe Abstraction

Independent of start configuration.

	1	2	3
4	5	6	7
8	9	10	11

	A	2	3
A	A	6	7
A	A	A	A

	A	A	A
A	A	A	A
8	9	10	11

Example:

2	6	3	4
10	9	5	
8	7	1	11

Fringe Abstraction

Independent of start configuration.

	1	2	3
4	5	6	7
8	9	10	11

	A	2	3
A	A	6	7
A	A	A	A

	A	A	A
A	A	A	A
8	9	10	11

Example:

2	6	3	4
10	9	5	
8	7	1	11

2	6	3	A
A	A	A	
A	7	A	A

A	A	A	A
10	9	A	
8	A	A	11

Fringe Abstraction

Independent of start configuration.

	1	2	3
4	5	6	7
8	9	10	11

	A	2	3
A	A	6	7
A	A	A	A

	A	A	A
A	A	A	A
8	9	10	11

Example:

2	6	3	4
10	9	5	
8	7	1	11

2	6	3	A
A	A	A	
A	7	A	A

A	A	A	A
10	9	A	
8	A	A	11

Heuristic value of state:

$h = cost + cost$

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

2	6	3	4
10	9	5	
8	7	1	11

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

2	6	3	4
10	9	5	
8	7	1	11

2_2	6_2	3_1	4_4
10_3	9_1	5_1	
8_0	7_3	1_3	11_0

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

2	6	3	4
10	9	5	
8	7	1	11

2_2	6_2	3_1	4_4
10_3	9_1	5_1	
8_0	7_3	1_3	11_0

A	A	A	4
10	A	A	
A	7	1	A

2	6	3	A
A	A	5	
A	A	A	A

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

2	6	3	4
10	9	5	
8	7	1	11

2_2	6_2	3_1	4_4
10_3	9_1	5_1	
8_0	7_3	1_3	11_0

A	A	A	4
10	A	A	
A	7	1	A

2	6	3	A
A	A	5	
A	A	A	A

	1	2	3
4	5	6	7
8	9	10	11

Special Abstraction

Created for a specific start configuration, using the eight tiles furthest from their goal positions.

2	6	3	4
10	9	5	
8	7	1	11

2_2	6_2	3_1	4_4
10_3	9_1	5_1	
8_0	7_3	1_3	11_0

A	A	A	4
10	A	A	
A	7	1	A

2	6	3	A
A	A	5	
A	A	A	A

	1	2	3
4	5	6	7
8	9	10	11

	1	A	A
4	A	A	7
A	A	10	A

	A	2	3
A	5	6	A
A	A	A	A

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe
  - Solving remaining tiles will not disturb solved tiles

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe
  - Solving remaining tiles will not disturb solved tiles
  - $h = 0$ means you're actually close to the goal

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe
  - Solving remaining tiles will not disturb solved tiles
  - $h = 0$ means you're actually close to the goal
- Specialized

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe
  - Solving remaining tiles will not disturb solved tiles
  - $h = 0$ means you're actually close to the goal
- Specialized
  - Solving remaining tiles might disturb solved tiles

Insight into Proposal

Greedy search will solve the sliding tile puzzle better with the fringe pattern database than a more specialized pattern database as a heuristic.

Why?

- Fringe
  - Solving remaining tiles will not disturb solved tiles
  - $h = 0$ means you're actually close to the goal
- Specialized
  - Solving remaining tiles might disturb solved tiles
  - $h = 0$ does not guarantee you're close to the goal

Results

	Greedy	A*
Fringe	44,171	218,816
Specialized	196,073	333,928

Table 1: Number Nodes Expanded

Results

	Greedy	A*
Fringe	44,171	218,816
Specialized	196,073	333,928

Table 1: Number Nodes Expanded

This shows:

- Fringe is better than specialized for both greedy and A* on puzzles of this size

Results

	Greedy	A*
Fringe	44,171	218,816
Specialized	196,073	333,928

Table 1: Number Nodes Expanded

This shows:

- Fringe is better than specialized for both greedy and A* on puzzles of this size
- Previously, it was shown specialized works better on big puzzles with big pattern databases, this does not generalize to smaller puzzles with disjoint pattern databases

Results

	Greedy	A*
Fringe	44,171	218,816
Specialized	196,073	333,928

Table 1: Number Nodes Expanded

This shows:

- Fringe is better than specialized for both greedy and A* on puzzles of this size
- Previously, it was shown specialized works better on big puzzles with big pattern databases, this does not generalize to smaller puzzles with disjoint pattern databases (Which one?)