Analysis of different algorithms for Maze solving

This document captures an analysis of different aspects of using different algorithms to solve the "maze problem" defined as, "Given a maze in the format of an image, find the shortest path from the source to the destination and highlight it."

Algorithms tested

- Breadth-first search (BFS)
- A-star with Manhattan distance heuristic

Maze image information

- Image format : BMP (.bmp)
- Bytes per pixel : 3

Image	Resolution (width*height)	Clear pixel count	% of clear pixels
1	10*10	43	43.00
2	15*15	100	44.44
3	41*41	801	47.65
4	201*201	20692	51.22
5	401*401	82724	51.45
6	1940*2001	2006882	51.70
7	2001*2001	2000001	49.95
8	4001*4001	8000001	49.98

Running time

- Unit: second(s)

- Typical environment:

  - *OS*: Ubuntu 16.04
  - *Processor*: Intel i5 (7th gen) @ 3GHz
  - *Physical RAM*: 4GiB
  - *Swap space* : 8GB

Image	Breadth-first search (BFS)	A-star (Manhattan distance)	BFS is faster by X%	A-star (Euclidean distance)	BFS is faster by X%
1	0.003	0.003	0.00	0.001	-200.00

Image	Breadth-first search (BFS)	A-star (Manhattan distance)	BFS is faster by X%	A-star (Euclidean distance)	BFS is faster by X%
2	0.005	0.002	-150.00	0.001	-400.00
3	0.002	0.003	33.00	0.002	0.00
4	0.010	0.014	28.50	0.012	16.67
5	0.031	0.035	11.42	0.033	6.06
6	0.625	0.746	16.22	0.822	23.97
7	0.537	0.649	17.25	0.667	19.49
8	2.376	2.976	20.16	3.103	23.43

Node expansion

Note: Values in parantheses represent the % of nodes expanded

Image	Breadth-first search	A-star (Manhattan distance)	X% fewer nodes than BFS	A-star (Euclidean distance)	X% fewer nodes than BFS
1	36 (83.72)	27 (62.79)	25.00	28 (65.12)	22.22
2	88 (88.00)	72 (72.00)	18.18	74 (74.00)	15.91
3	780 (97.38)	770 (96.13)	1.28	771 (96.25)	1.15
4	13866 (67.01)	10104 (48.83)	27.13	10559 (51.03)	23.85
5	70167 (84.82)	39943 (48.28)	43.07	47091 (56.93)	32.89
6	1949485 (97.14)	1248266 (62.20)	35.97	1568188 (78.14)	19.56
7	918257 (45.19)	837378 (41.86)	8.80	857383 (42.87)	6.63
8	6201755 (77.52)	6023496 (75.29)	2.87	6079769 (76.00)	1.97