Performance Metrics

Source: (13,6) Target: (7,6)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	247	**22**	**28**	275
DFS	158	91	68	232
GBFS	**40**	35	**28**	**74**

Source: (1,20) Target: (20,1)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	270	**22**	**33**	271
DFS	212	99	46	262
GBFS	**64**	51	47	**114**

Source: (4,3) Target: (17,12)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	270	**26**	**26**	271
DFS	159	85	46	211
GBFS	**34**	38	**26**	**71**

Source: (1,1) Target: (20,1)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	185	**21**	**19**	201
DFS	127	84	67	200
GBFS	**20**	**21**	**19**	**40**

Source: (13,3) Target: (19,11)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	238	**20**	**24**	253
DFS	158	87	40	203
GBFS	**45**	48	**24**	**92**

Source: (1,14) Target: (7,20)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	68	19	**6**	80
DFS	**7**	**14**	**6**	**20**
GBFS	**7**	**14**	**6**	**20**

Source: (1,1) Target: (4,4)	Time-Efficiency (goal tests)	Memory Efficiency (max frontier size)	Solution Path Length	Vertices Visited
BFS	16	9	**3**	23
DFS	**4**	**7**	**3**	**10**
GBFS	**4**	**7**	**3**	**10**

Which Algorithm is the fastest ?
GBFS performs much better in every case.

Which is the most memory efficient ?
There is no single most efficient algorithm here which performs best in all the cases.

Which visits the fewer vertices ?
GBFS visits fewer cases in all the cases.

Which generates the shortest path length?
BFS finds the shortest path length.

Are the performance differences what you expected based on the theoretical complexity analysis ?

The results are contrasting on many parameters than expected as per theoretical complexity:

1. Time complexity: BFS was expected to perform better than DFS, but ended up performing worse.
2. Space complexity: DFS was expected to perform better than BFS, but it performed worse than BFS in most of the cases.

Does BFS always find the shortest path?

Yes, in our particular scenario it has found the shortest path in all the cases as all the edge weights are considered positive.

Does GBFS always go for "straight" to the goal or are there cases where it gets side-tracked?

There are several cases where its gets side-tracked, and its these cases where BFS ends up finding shortest path.