CPSC 322, Practice Exercise
Solutions to Uninformed Search

1 Directed Questions

- What is meant by search algorithm *completeness*? **Answer:** If an algorithm is complete, it means that if at least one solution exists then the algorithm is guaranteed find a solution in a finite amount of time.
- What is meant by search algorithm *optimality*? **Answer:** If a search algorithm is optimal, then when it finds a solution it finds the *best* solution.
- What are the advantages of breadth-first search (BFS) over depth-first search (DFS)? **Answer:** BFS is complete and optimal, while DFS is not guaranteed to halt when there are loops.
- What is the advantage of DFS over BFS? **Answer:** If m is the maximum path length and b is the branching factor, the space complexity for DFS is mb while for BFS it is b^m.

2 Uninformed Search

Consider the search problem represented in Figure 1, where a is the start node and f is the goal node. Would you prefer DFS or BFS for this problem? Why?

Answer: If we were just running vanilla DFS (no pruning or loop checking) then we would prefer BFS, because DFS could get stuck in an infinite loop. Note that DFS is sensitive to the ordering of the nodes. If it explores to the left first it will get stuck in the loop, whereas if it explores to the right first it will find the goal very quickly.

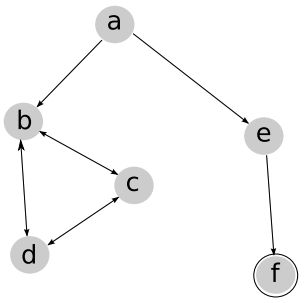

Figure 1: Comparing BFS and DFS

Which sequences of paths are explored by BFS and DFS in this problem? **Answer:** DFS explores $a->b->d->b->d$ and keeps oscillating between the two nodes b and d. BFS first adds $a->b$ and $a->e$ to the frontier. It expands ab and adds abd and abc to the frontier. Path ae is then expanded, adding aef to the frontier. Path abd is selected and removed from the frontier, and expanded so that $abdb$ and $abdc$ are added to the frontier. Path

abc is selected and expanded, adding *abcb* and *abcd* to the frontier. Finally, aef is selected and the goal node is reached.

3 Learning Goals

You can:

- Apply basic properties of search algorithms: completeness, optimality, time and space complexity of search algorithms.
- Select the most appropriate search algorithms for specific problems.