

1. Success/Completeness & Node Expansions:

The search limit of 4 was incomplete, failing to find the goal because the depth limit was smaller than the actual distance to the goal (6 steps). This led to fewer nodes to be expanded which would have saved time if the goal were to be closer to the start. On the other hand, the search limit of 8 was successful, because the limit was greater than the goal depth which also lead to more nodes to be expanded and thus consuming more resources as compared to the limit of 4 but it got to the goal. Also, DLS returns the first path it finds within the limit, not necessarily the shortest. So if memory is not a constraint BFS would be a better approach.

2. Contrast DLS with BFS/DFS:

DFS has a major risk, if the graph has a cycle or is infinite, DFS will go down a forever and possibly crash (infinite loop). DLS fixes this by limiting how deep it can go before backtracking. Making it safer but requires you to guess the correct or approximate depth.

As for BFS, while it is memory consuming, it will definitely find the shortest/optimal goal first. On the other hand, DLS is more memory efficient because it only remembers the current path similar to DFS, but it will not get you the shortest path if the limit is set incorrectly.

Success/completeness and node expansions:

The search limit of 4 was incomplete because the target was not found, as the depth limit was smaller than the actual distance to the target (6 steps). This resulted in fewer nodes being expanded, which would have saved time if the target had been closer to the start. On the other hand, the search limit of 8 was successful because the limit was greater than the target depth, which also resulted in more nodes being expanded and thus more resources being consumed than with the limit of 4, but the target was reached. In addition, DLS returns the first path it finds within the limit, not necessarily the shortest one. So if memory space is not a constraint, BFS would be the better approach.

Comparison between DLS and BFS/DFS:

DFS carries a big risk: if the graph has a cycle or is infinite, DFS will run endlessly and possibly crash (infinite loop). DLS solves this problem by limiting the depth to which it can go before backtracking. This makes it safer, but requires you to guess the correct or approximate depth.

BFS consumes a lot of memory, but it will definitely find the shortest/optimal path first. DLS, on the other hand, is more memory-efficient because, similar to DFS, it only stores the current path, but it will not find the shortest path if the limit is set incorrectly.