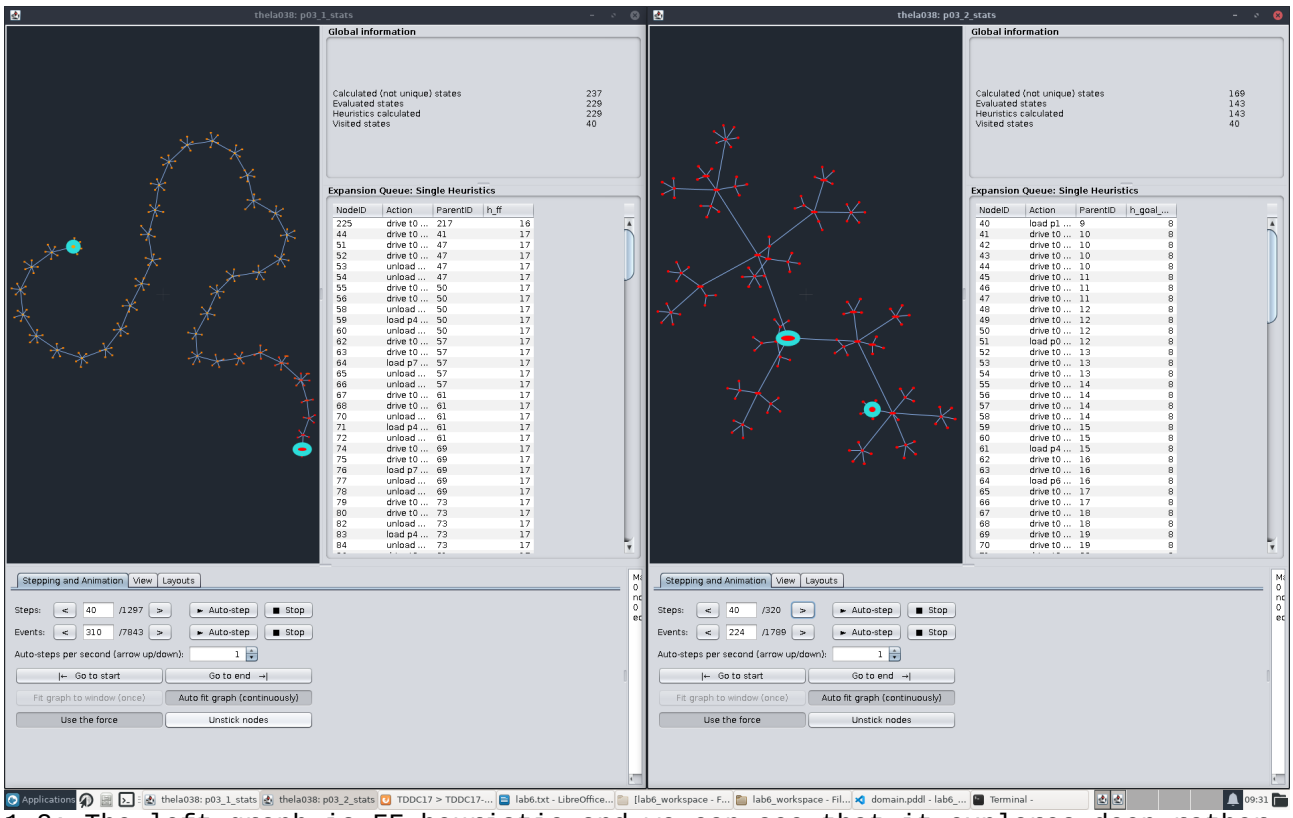
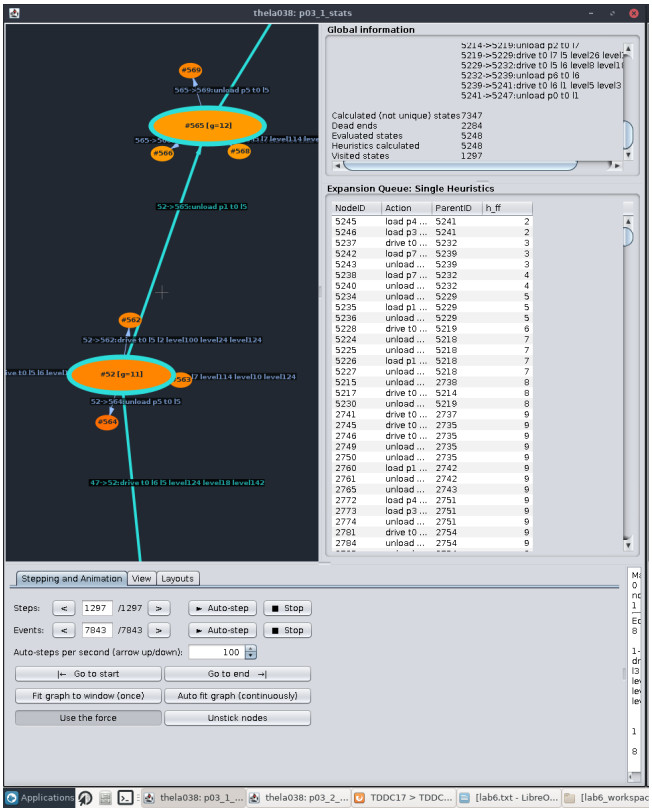


1.2: The left graph is FF heuristic and we can see that it explores deep rather than wide. The graph to the right is goal count heuristic and we see that it does the opposite and explores wide rather than deep.

1.3: They seem to be doing the same actions, but at different times.

1.4: It uses parts of the long chain, but a large amount of nodes it generated and their children remains unused in the final solution. For example, the branch in the screenshot is from the long chain and it takes a big jump in numbers.

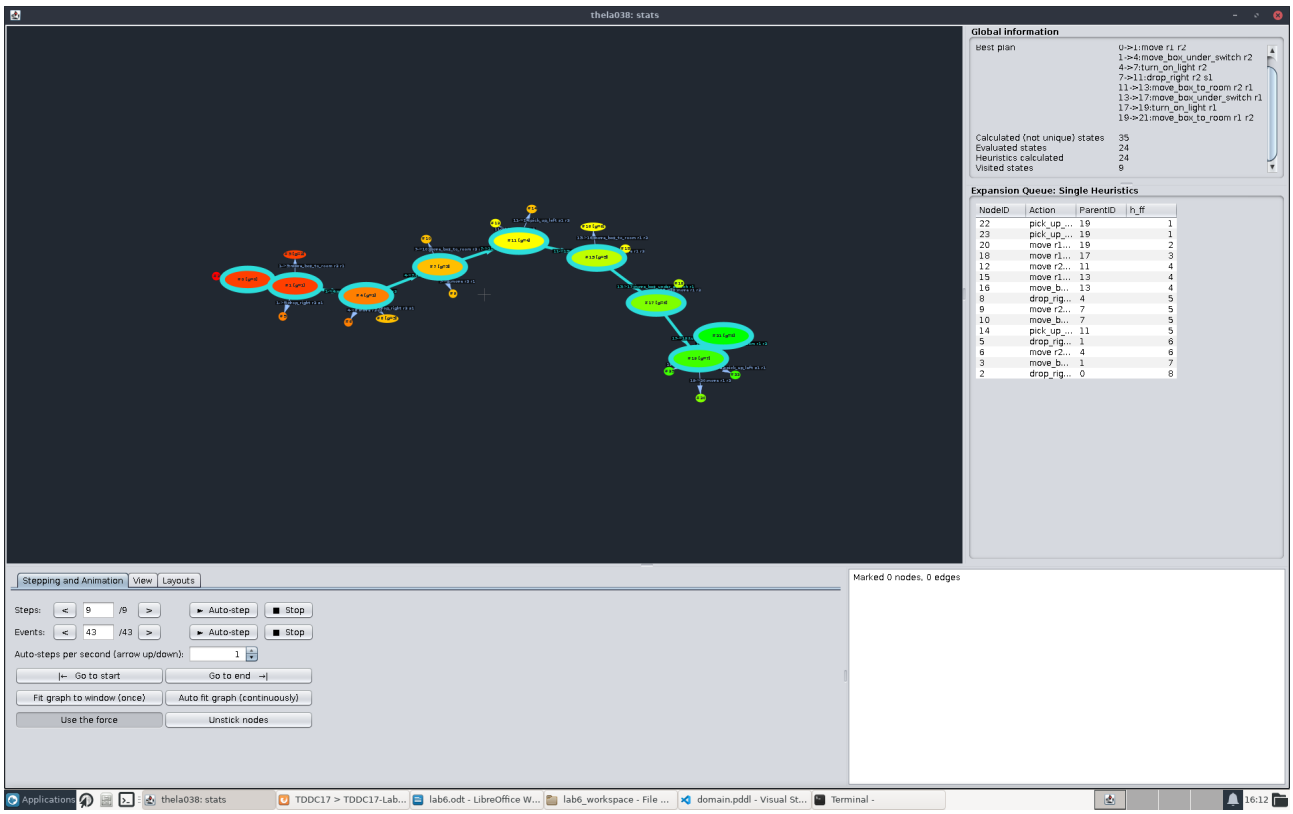


2.2: FF decreases its heuristic with the first time step. GC heuristic didn’t decrease its heuristic until step 19.

2.3: The heuristic of the furthest progressed node in step 27 has a value of 6.

2.4: Yes, it increases in value since nodes that have a low heuristic value, might not actually result in a successful path. If a path leads to a dead end, it has to back up.

3: It is not very similar. This is because the problem is very trivial in comparison.



4:
p02 FF: 27 steps
p02 GC: 12222 steps
p02 GC FF suggestions: 1575 steps

p03 FF: 1297 steps
p03 GC: 320 steps
p03 GC FF suggestions: 249 steps
The FF heuristic has the lowest mean value, but it was not best at every task.

Our findings are not applicable to all domain configurations, some heuristics will be better for some domains due to their different natures.