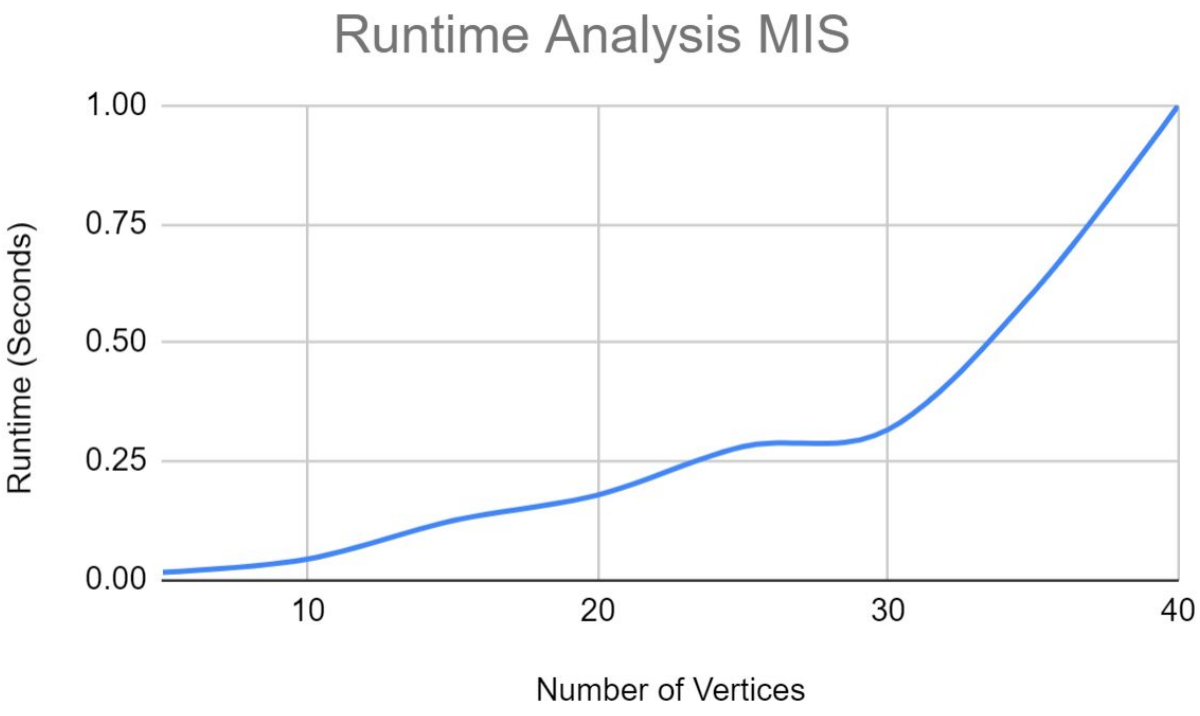
1. The structure of a graph, which is input to a state of the art MIS or maximum clique algorithm, can have an effect even more profound than that of its size. For example, using any of the algorithms described in [1] on a large (10,000+ vertices) graph with no edges would finish very quickly since the maximum independent set is all of the nodes. On the contrary a small graph (100 vertices), which is complete [2], would have an enormous search space.
2. The AFIT graph program is used to compute the maximum independent for a desired input graph. It is also capable of performing other common graph search algorithms but here we are focusing on the MIS/clique problems. To test the quality of the software I created 8 graphs of varying size from 5-40 vertices and ran the MIS algorithm on each one, recording the execution time as well as some other related data (presented below). The software performs as expected for graphs of increasing size. As a critique I would have implemented an option to pass in the desired graph text file and algorithm option as command line parameters. This way a batch or bash file could be written to test many different graphs and compile statistics on them quickly and efficiently. The way the user interface works currently this is not possible.
3. The partial search graph for problem two is contained in partialSearchGraph.pdf in the same directory as this document.



Number of Vertices	MIS Execution Time Seconds	Independence Number	Clique Cardinality
5	0.016	2	3
10	0.044	4	4
15	0.125	5	4
20	0.179	6	6
25	0.281	6	5
30	0.317	6	6
35	0.606	6	7
40	1	6	7

Reference

[1]https://en.wikipedia.org/wiki/Maximal_independent_set#Finding_a_single_maximal_independent_set

[2]https://en.wikipedia.org/wiki/Complete_graph