Report

Part a

	1 edge	5 edges	10 edges	15 edges	20 edges
mvc_1	1.00	1.02	1.00	1.00	1.00
mvc_2	2.58	1.68	1.33	1.19	1.11
mvc_3	2.00	1.79	1.55	1.39	1.35

Table 1. Rounded to 3 significant digits, sum of the lengths of MVCs generated by the approximation algorithms of 100 graphs of 1, 5, 10, 15, and 20 edges, divided by the sum of the lengths of actual MVCs for the same 100 graphs to obtain how many times longer than minimum the approximations were.

The information in Table 1 is also displayed in Graph 2 in the Graphs section. On average, mvc_1 is 1.01 times the minimum, mvc_2 is 1.58 times, and mvc_3 is 1.62 times longer than the minimum. The averages are also displayed in Graph 3.

Part b

As apparent in Graph 2, mvc_1's expected performance shows no correlation to the number of edges in a graph, however, mvc_2 and mvc_3 both get better as the number of edges increase and the graph gets fuller.

Part c

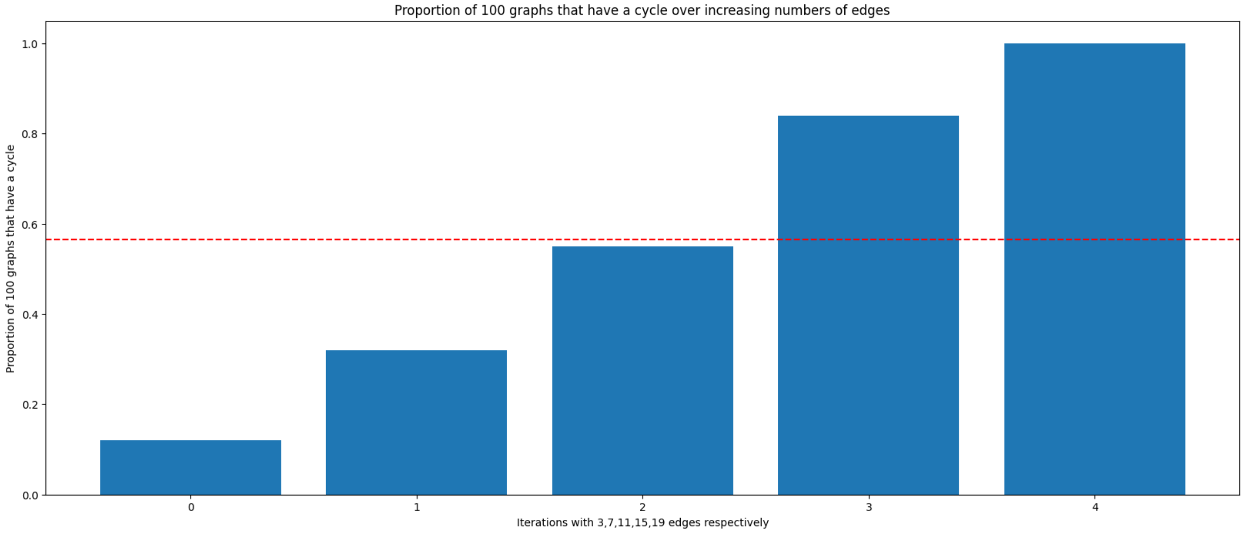
As apparent in Graph 5, mvc_1's expected performance shows no correlation to the number of nodes in a graph, however, mvc_2 and mvc_3 both get better as the number of edges decrease and the graph gets fuller (as the number of edges stay the same).

Part d

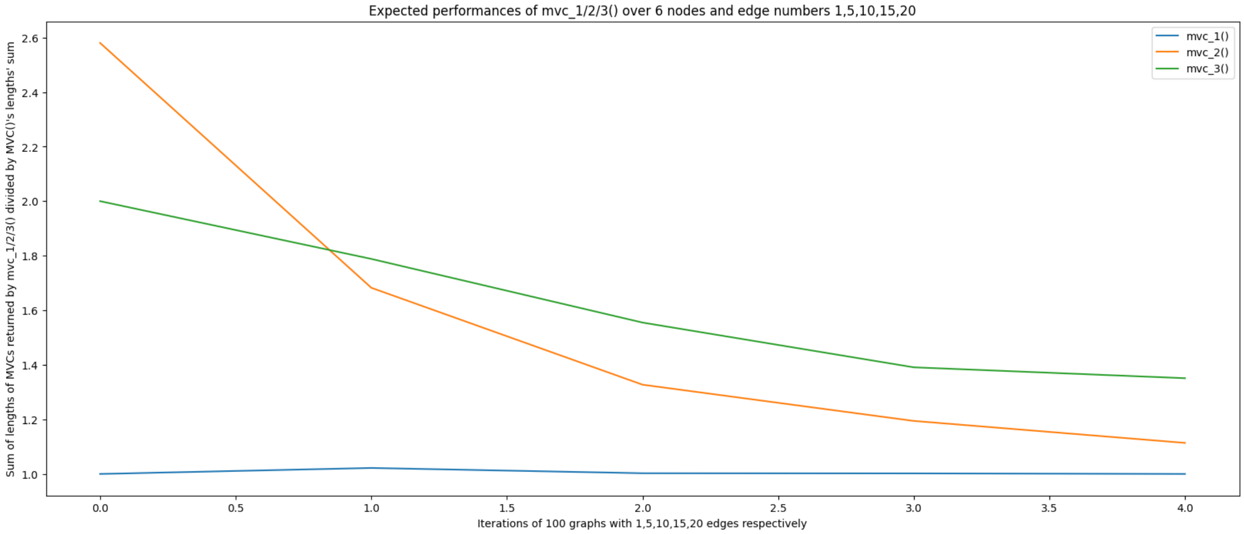
I used the suggested experiment design in the assignment sheet and ran the 100 trials for 5 iterations with 3,7,11,15, and 19 edges and 20 nodes. Respectively, 12%, 32%, 55%, 84%, and

100% of graphs had cycles. This information is also displayed in Graph 1, which displays the overall average as well.

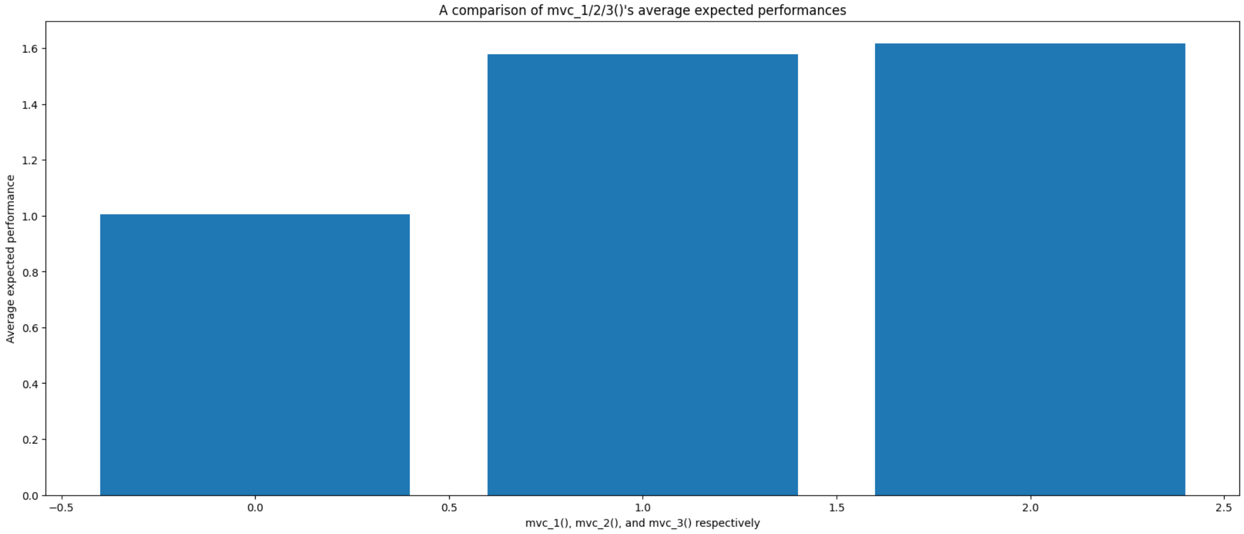
Graphs



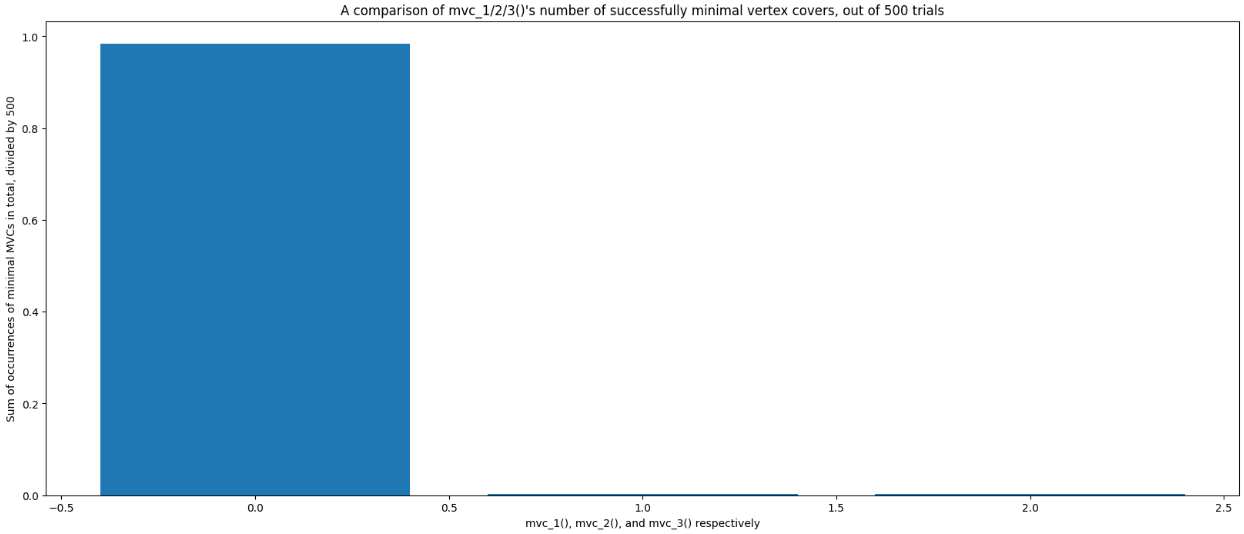
Graph 1. Generated from experiment_1



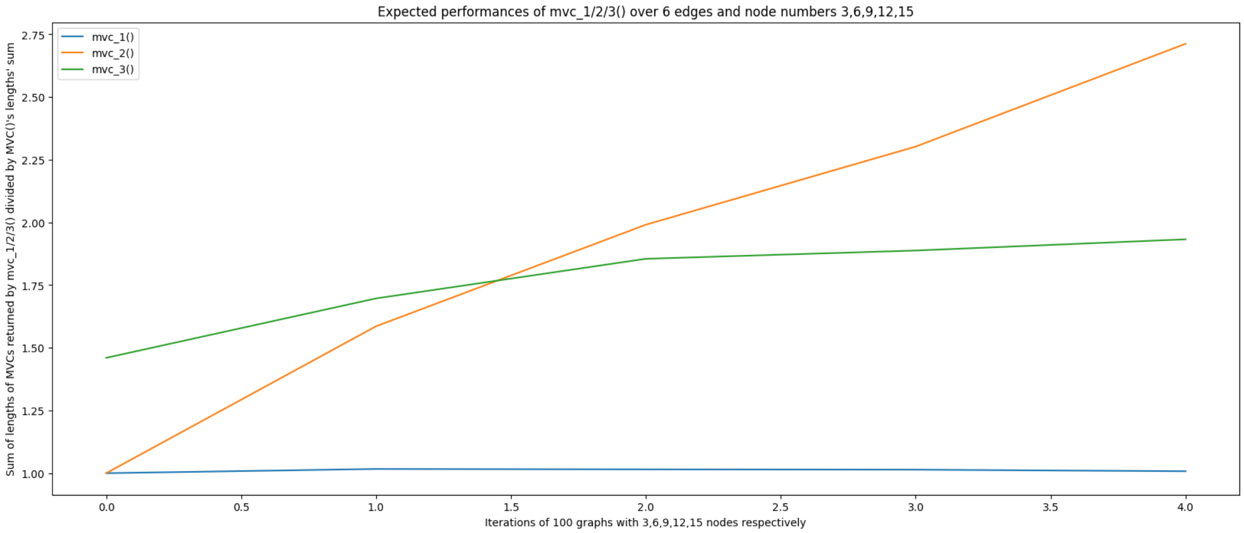
Graph 2. Generated from experiment_2



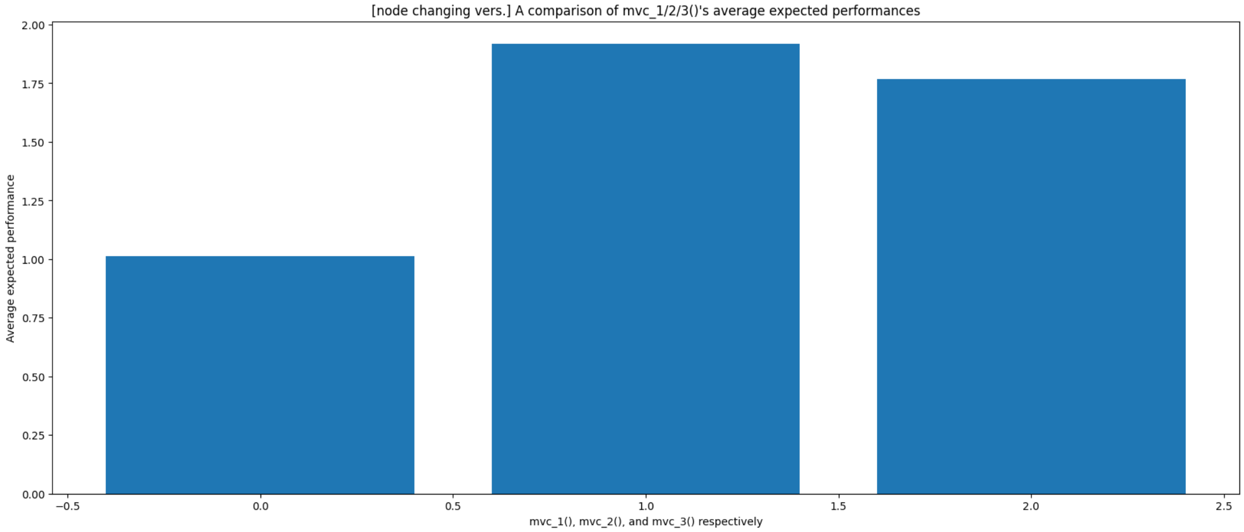
Graph 3. Generated from experiment_2



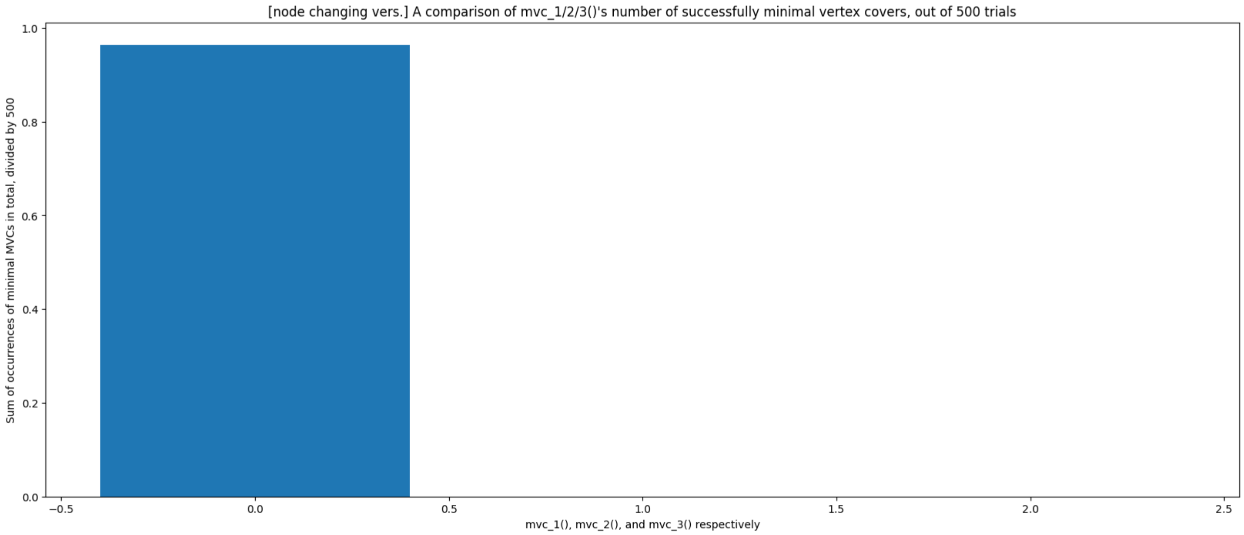
Graph 4. Generated from experiment_2



Graph 5. Generated from experiment_3



Graph 6. Generated from experiment_3



Graph 7. Generated from experiment_3