An Analysis of Breadth-First Search Performance In Adjacency List and Adjacency Matrix Representations of Simple Graphs

Sheikh Azizul Hakim
S201705002, CSE, BUET

A simple graph can be represented in both the adjacency list and adjacency matrix representations. Here, we have applied breadth-first search in both representations of the same "randomly created" graph and tried to analyze the effect of the underlying data structure on runtime.

Adjacency List:

Different |E| for same |V|:

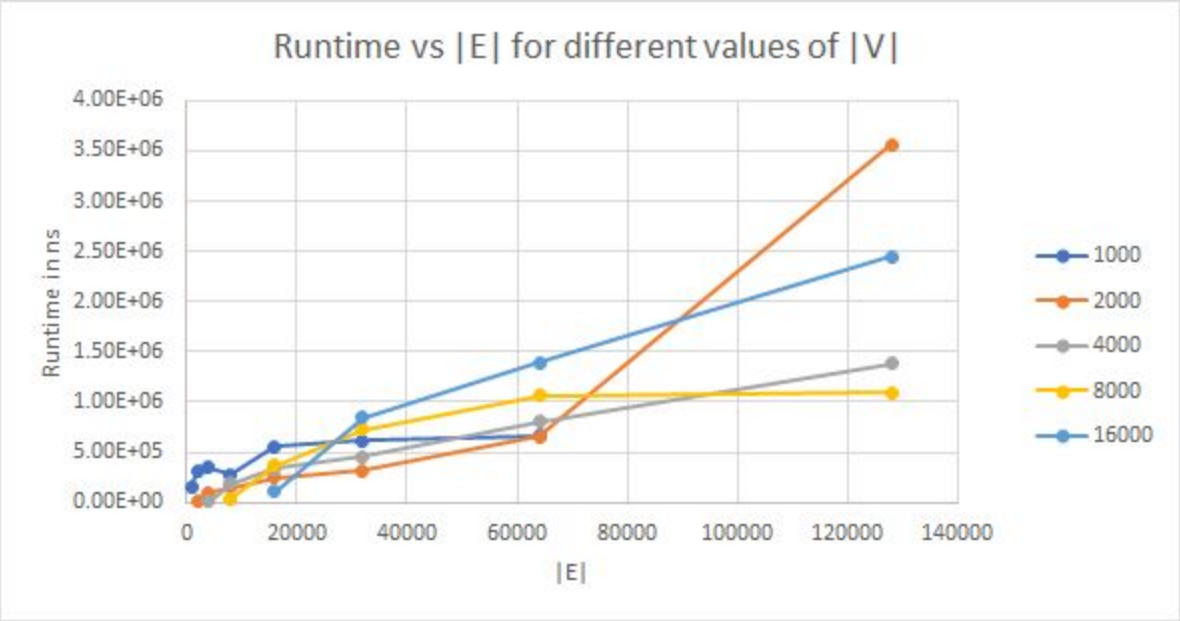


Fig: Analysis for Adjacency Lists

The trends in the attached graph clearly indicate that there is a linear relationship between runtime and |E| for a fixed |V|. The relationship is more apparent in larger values of |V|, where runtime gets proportional to |E|.

The rocketing up of |V|=2000 line is unexpected, and randomness can be a crucial factor behind those results.

Different |V| for same |E|:

The above chart can also be used to infer that ignoring random chaos, for fixed |E|, we have greater runtime value for greater |V|.

Adjacency Matrix:

Different |E| for same |V|:

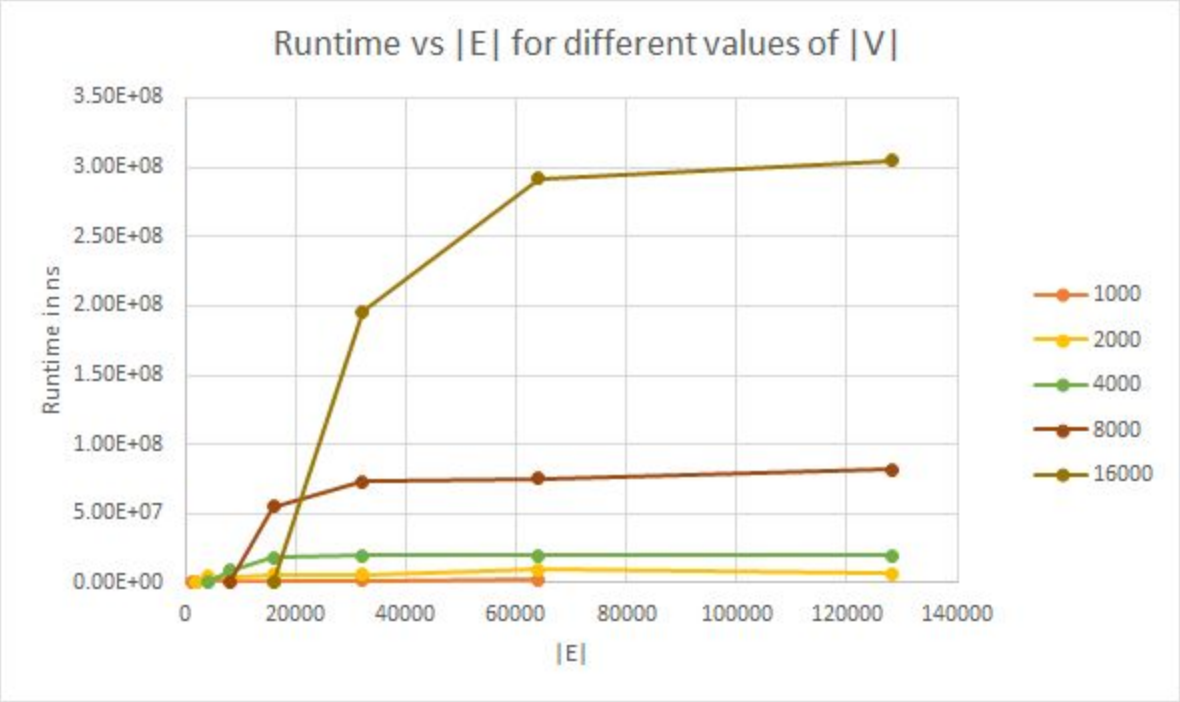


Fig: Analysis for Adjacency Matrices

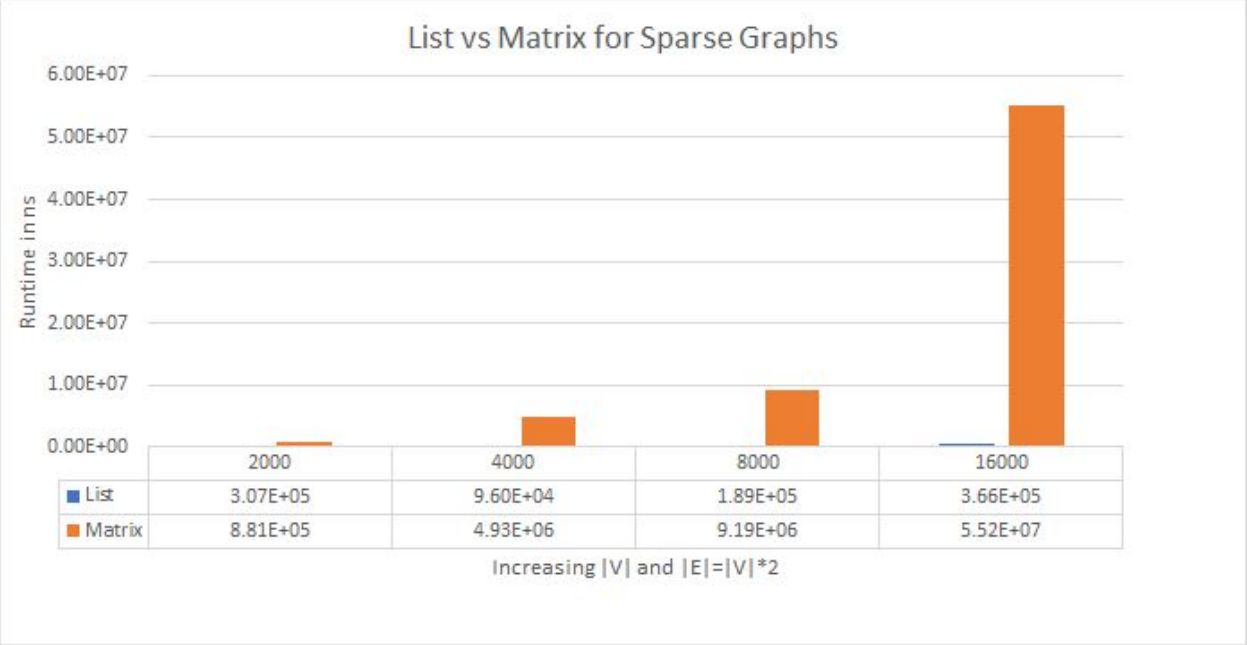
As a general trend, with sufficient |E|, the runtime almost gets constant for a particular value of |V|. However, for small values of |E|, the graph is extremely sparse, and it is very likely, the randomly selected source vertex could not find explore many vertices in the graph and thereby exited early.

Different |V| for same |E|:

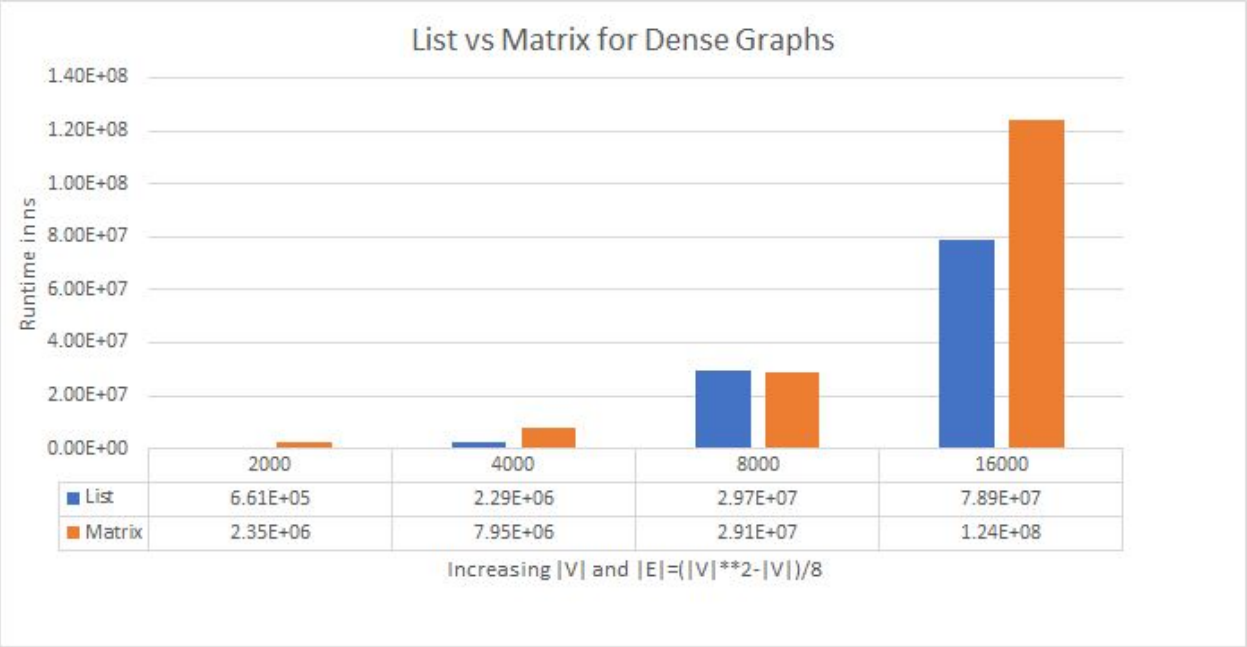
The relationship of runtime over |V| is much more apparent in the matrix case than in the list case. Using the values, it follows that runtime increases proportionally with the square of |V|.

Comparison:

For sparse graphs, where |E| is almost linear with |V|, the adjacency list well outperforms the adjacency matrix representation.



However, for dense graphs, the relationship changes. There, the runtime stays almost the same for both representations.



Summary:

A breadth-first search analysis certainly prefers adjacency list representations for sparse graphs and performs almost the same for sparse and dense graphs in adjacency matrix representations.