ANALYSIS OF SOLUTIONS TO

MAX FLOW PROBLEM

ABDULLAH JUNEJO, OWAIS WAHEED, MAHWISH AHMED
HABIB UNIVERSITY

PROBLEM INTRODUCTION

It is all about determining the maximum amount of flow that can be sent from a designated source vertex to a designated sink vertex in a flow network, assuming every vertex has capacity constraints on the corresponding edges. Despite its abstractness, it has numerous applications in network flow optimization, transportation planning, and resource allocation.

OBJECTIVE

To empirically analyze and compare the performance of three maximum flow algorithms - FordFulkerson, Edmonds-Karp, and Dinic's algorithm - under varying graph sizes with the hope to find optimal solution

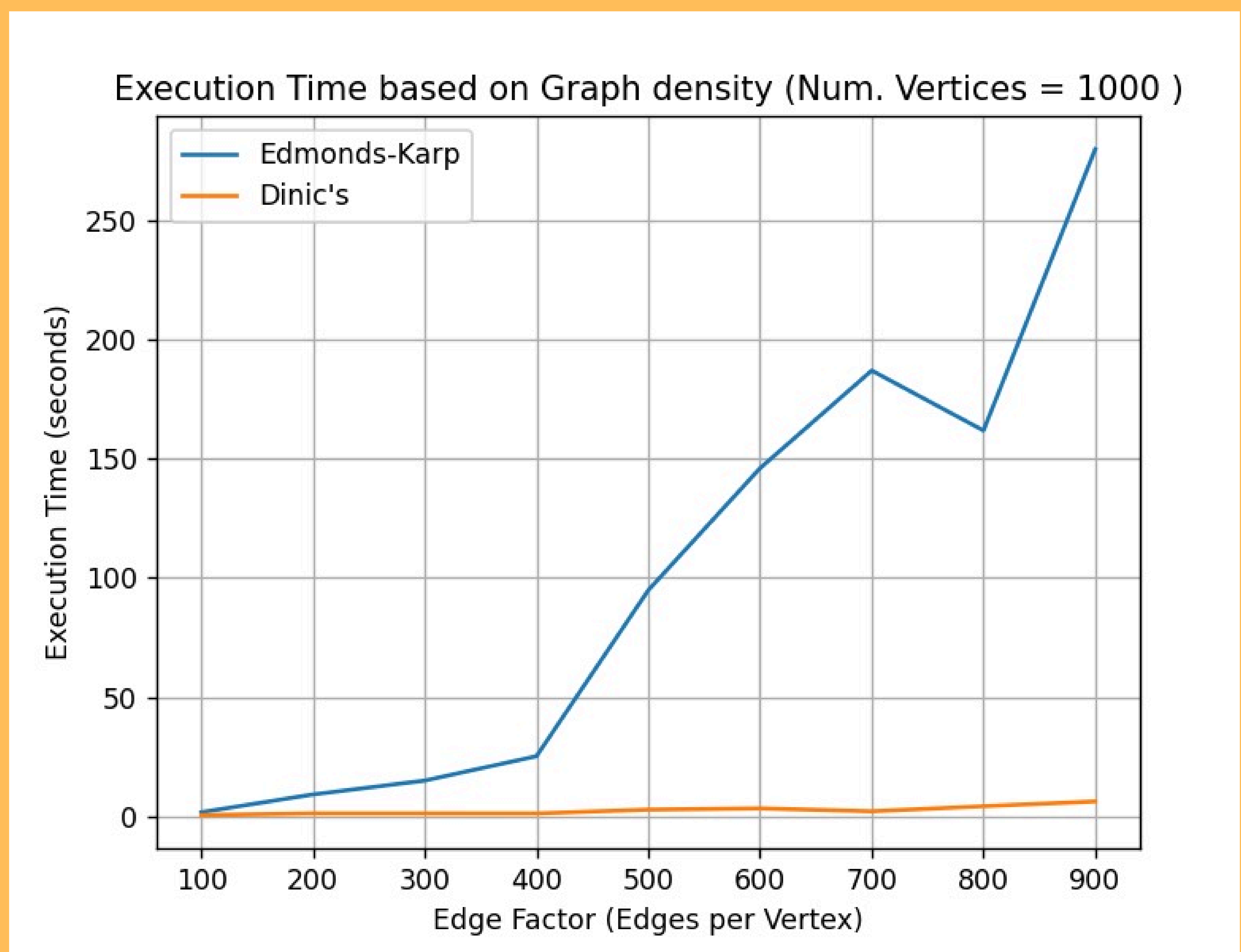
METHODOLOGY

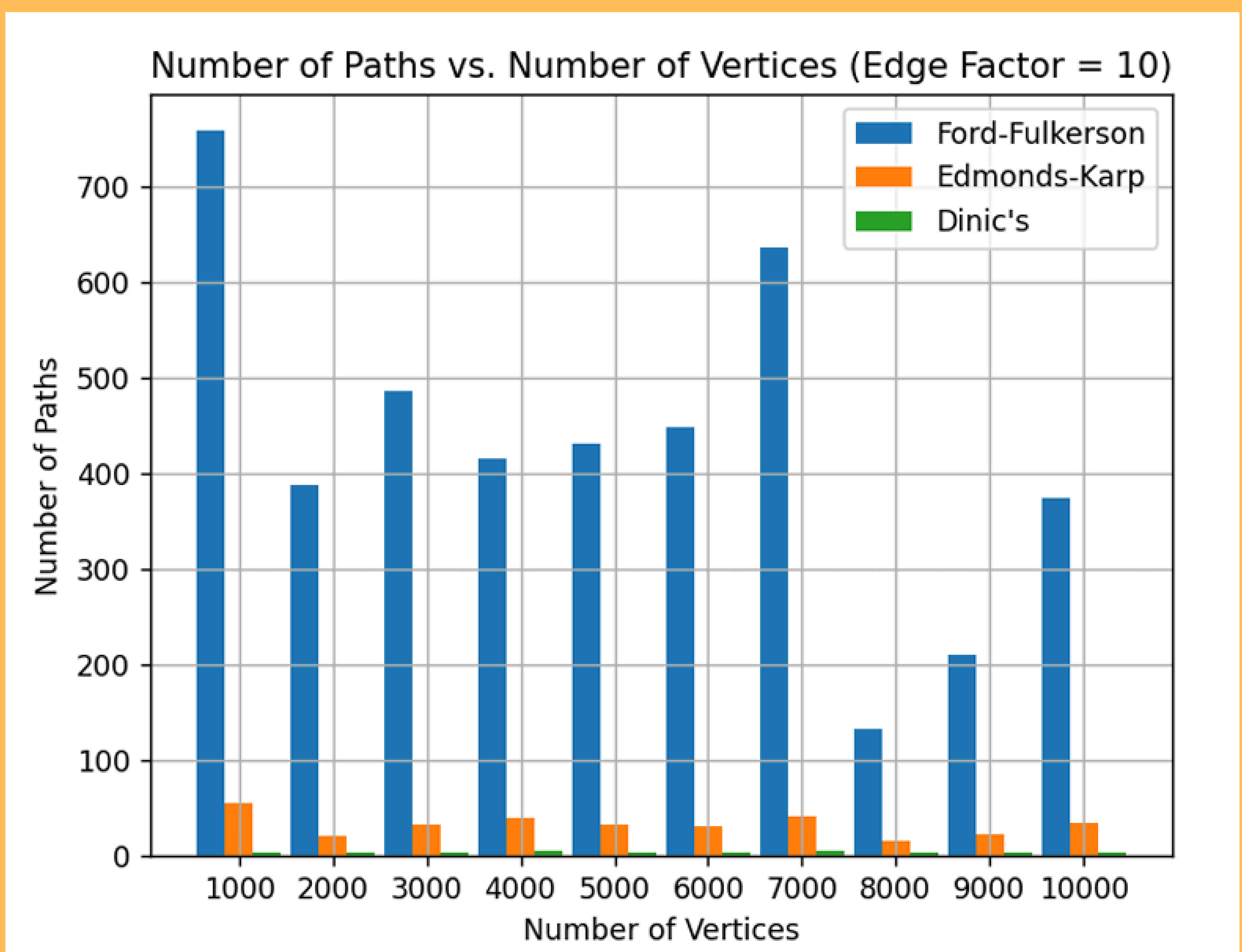
The key focus was on measuring and comparing their execution times and associated computational complexities using randomized graphs with controlled parameters i.e. No. of vertices/Edge Factor. Due to lack of resources the maximum number of edges used was restricted to 100,000. A final comparison was of No. of Residual Graphs created.

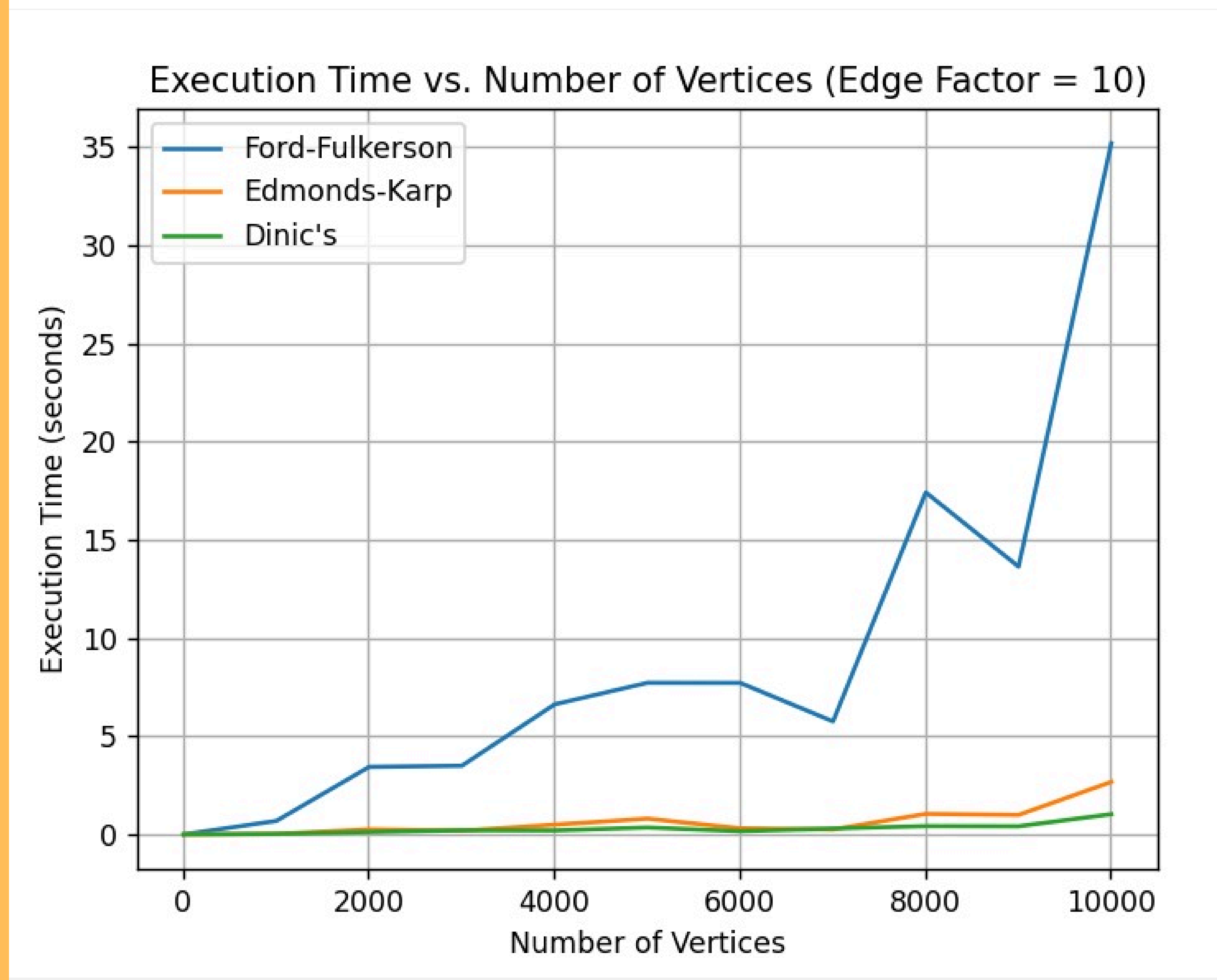
ANALYSIS

The execution time analysis compares the performance of each algorithm in terms of time complexity, measured in seconds against two different variables. i.e. increasing no. of vertices, and increasing no. of edges when vertices are set constant at 1000.

- Ford Fulkerson $O(E|f|)$ is unable to calculate on Dense Graphs.
- Edmond Karp despite the efficient performance, the algorithm exhibited increased execution times with highly dense graphs due to the $O(VE2)$ time complexity.
- Dinics $O(EV2)$ outperformed Edmond Karp by huge margins in dense graphs.
- Compared to the Ford-Fulkerson, both Dinic and Edmond-Karp algorithm uses significantly fewer paths as they use BFS to update the residual graph.







RESULTS

While Ford-Fulkerson exhibits increasing execution times with larger and denser graphs, especially struggling with highly dense networks, Our Empirical Analysis demonstrated Edmonds-Karp's improved efficiency over Ford-Fulkerson in less dense graphs but is limited in denser scenarios. Dinic's algorithm outperformed both by remaining relatively stable, utilizing fewer residual graph constructions.

GITHUB REPOSITORY

Comparitive-Analysis-Of-MaxFlow-Algorithms. https: //github.com/Owais-Waheed/Comparitive-Analysis-Of-MaxFlow-Algorithms.

REFERENCES

[1] Ravindra K. Ahuja, Murali Kodialam, Ajay K. Mishra, and James B. Orlin. 1997. Computational investigations of maximum flow algorithms. European Journal of Operational Research 97, 3 (1997), 509–542. https://doi.org/10.1016/S0377- 2217(96)00269-X

[2] L. R. Ford and D. R. Fulkerson. 1956. Maximal Flow Through a Network. Canadian Journal of Mathematics 8 (1956), 399–404. https://doi.org/10.4153/CJM-1956-045-5

[3] Antonio Sedeño-Noda, M. González-Sierra, and C. González-Martín. 2000. An algorithmic study of the Maximum Flow problem: A comparative statistical analysis. TOP: An Official Journal of the Spanish Society of Statistics and Operations Research 8 (02 2000), 135–162. https://doi.org/10.1007/BF02564832

[4] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. 2022. Introduction to Algorithms (4th ed.). MIT Press.