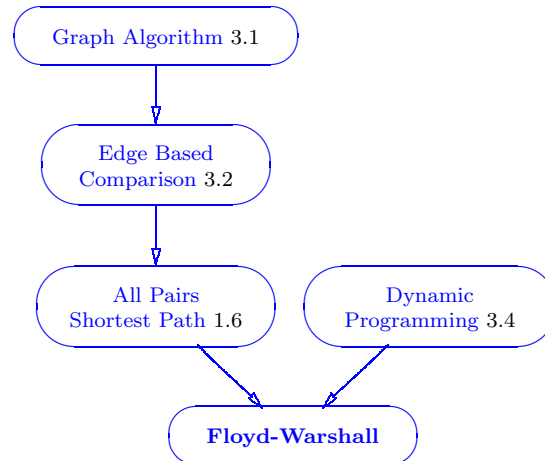


3.10.1 Floyd-Warshall

Section authors: Anurat Chapanond, Lauren Foutz, Scott Hill, Joseph Urban.



Prototype: `bool Floyd_Warshall_all_pairs_shortest_paths(
const std::vector<std::vector<int> > & graph,
std::vector<std::vector<double> > & distance_matrix,
const std::vector<std::vector<double> >
& weight_map)`

Refinement of: Graph Algorithm (§3.1),
therefore of Edge Based Comparison (§3.2)
and of All Pairs Shortest Path (§3.4).
Also a refinement of the Dynamic Programming (§1.6) design strategy.

Effects: Standard effects of an All Pairs Shortest Path (§3.4) algorithm. In brief:
The algorithm detects the presence of a negative weight cycle in a directed

or undirected graph. If such a cycle is found, the algorithm terminates returning false immediately. If no such cycle is found, the algorithm returns true and the shortest path between each pair of vertices, (u, v) , is stored in $distance_matrix[u][v]$.

Asymptotic complexity: For a graph $G = (V, E)$.

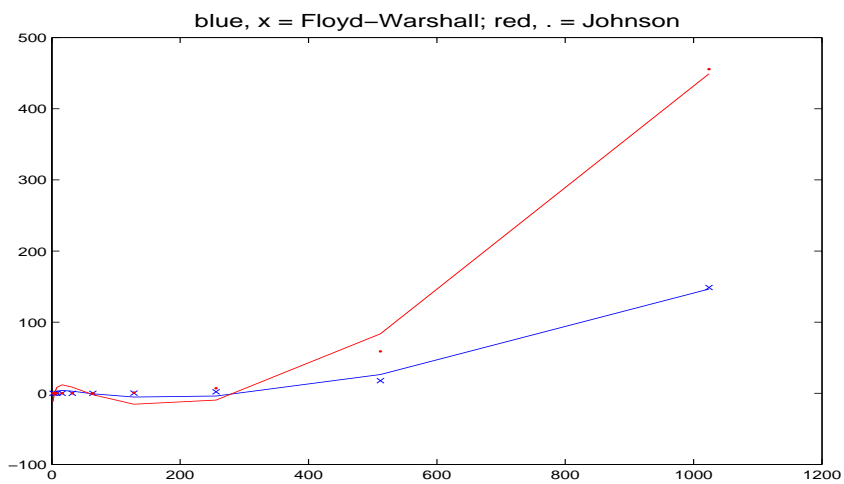
- Average case: $\Theta(V^3)$
- Worst case (fully connected graph): $\Theta(V^3)$

3.10.2 Floyd-Warshall Fully Connected Graph Timings

Comparison of timings on a fully connected graph against those for Johnson's Algorithm (§??).

Number Vertices	Floyd Results	Johnson Results
2	0.00	0.00
4	0.00	0.00
8	0.00	0.00
16	0.00	0.00
32	0.01	0.02
64	0.04	0.14
128	0.28	1.02
256	2.20	7.18
512	17.79	58.97
1024	148.74	455.57

3.10.3 Least Squares Fitting of Floyd-Warshall Versus Johnson Fully Connected Graph Timings

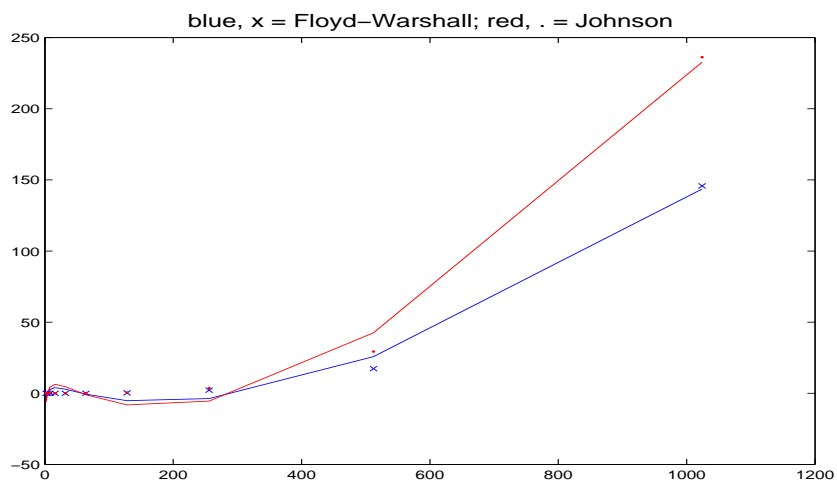


3.10.4 Floyd-Warshall Dense Graph Timings

Comparison of timings on a dense, but not fully connected graph against those for Johnson's Algorithm (§??).

Number Vertices	Floyd Results	Johnson Results
2	0.00	0.00
4	0.00	0.00
8	0.00	0.00
16	0.00	0.00
32	0.01	0.01
64	0.04	0.08
128	0.28	0.53
256	2.18	3.63
512	17.34	29.30
1024	145.76	236.21

3.10.5 Least Squares Fitting of Floyd-Warshall Versus Johnson Dense Graph Timings

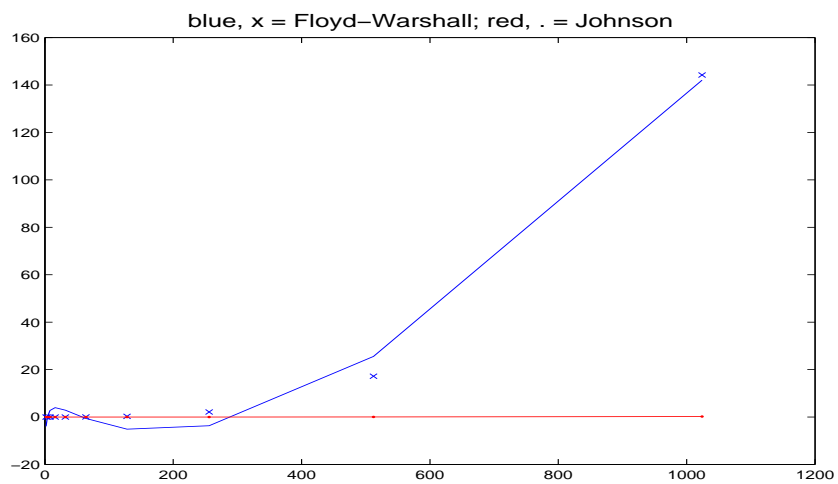


3.10.6 Floyd-Warshall Sparse Graph Timings

Comparison of timings on a sparse graph against those for Johnson's Algorithm (§??).

Number Vertices	Floyd Results	Johnson Results
2	0.00	0.00
4	0.00	0.00
8	0.00	0.00
16	0.00	0.00
32	0.00	0.00
64	0.03	0.00
128	0.30	0.00
256	2.13	0.01
512	17.20	0.05
1024	144.23	0.19

3.10.7 Least Squares Fitting of Floyd-Warshall Versus Johnson Sparse Graph Timings



3.10.8 Notes

The following conventions are used in the above analyses:

- A dense graph contains $(V^2)/2$ edges.
- A sparse graph contains $(V/2)$ edges.
- All Graphs are represented by adjacency matrices.