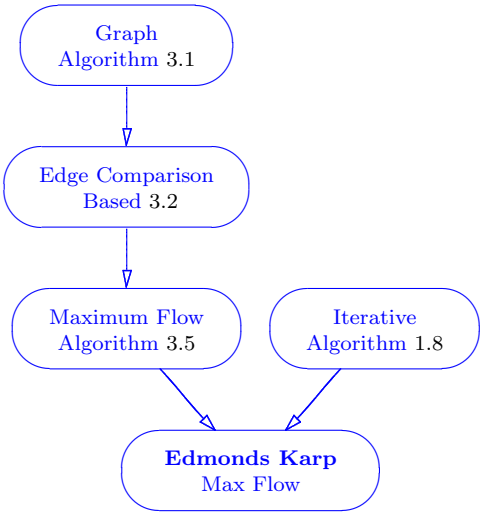
### 3.5.1 Edmonds Karp MaxFlow

Section authors: Shyam Sunder, Kartik Babu, Viswanath Ramakkagari and Ryan Lewis




**Refinement of:** Iterative Algorithm (§1.8) and of Maximum Flow Algorithm (§3.5), therefore of Edge Comparison Based Graph Algorithm (§3.2), and Graph Algorithm (§3.1).

**Prototype:**

```
template <class Graph,
          class CapacityEdgeMap,
          class ResidualCapacityEdgeMap,
          class ReverseEdgeMap,
          class ColorMap, class PredEdgeMap>
        edmonds_karp_max_flow
          (graph, src, sink,
           capacityEdgeMap,
           residualCapacityEdgeMap,
           reverseEdgeMap,
           colorMap, predEdgeMap)
```

**Input:** Map with edge capacities, map with residual edge capacities, a source vertex and a sink vertex.

**Output:** The edges along with their flow when there is maximum flow across the network.

**Effects:** No effects on the edge capacities map. The reverseEdgemap is modified every time there is a flow augmentation.

**Asymptotic complexity:** Let $V$ = Number of Vertices, $E$ = Number of Edges

- Average case (random data): $O(VE^2)$
- Number of Flow Augmentations: $O(VE)$
- If Edges are integers and bounded by a constant U: $O(VEU)$

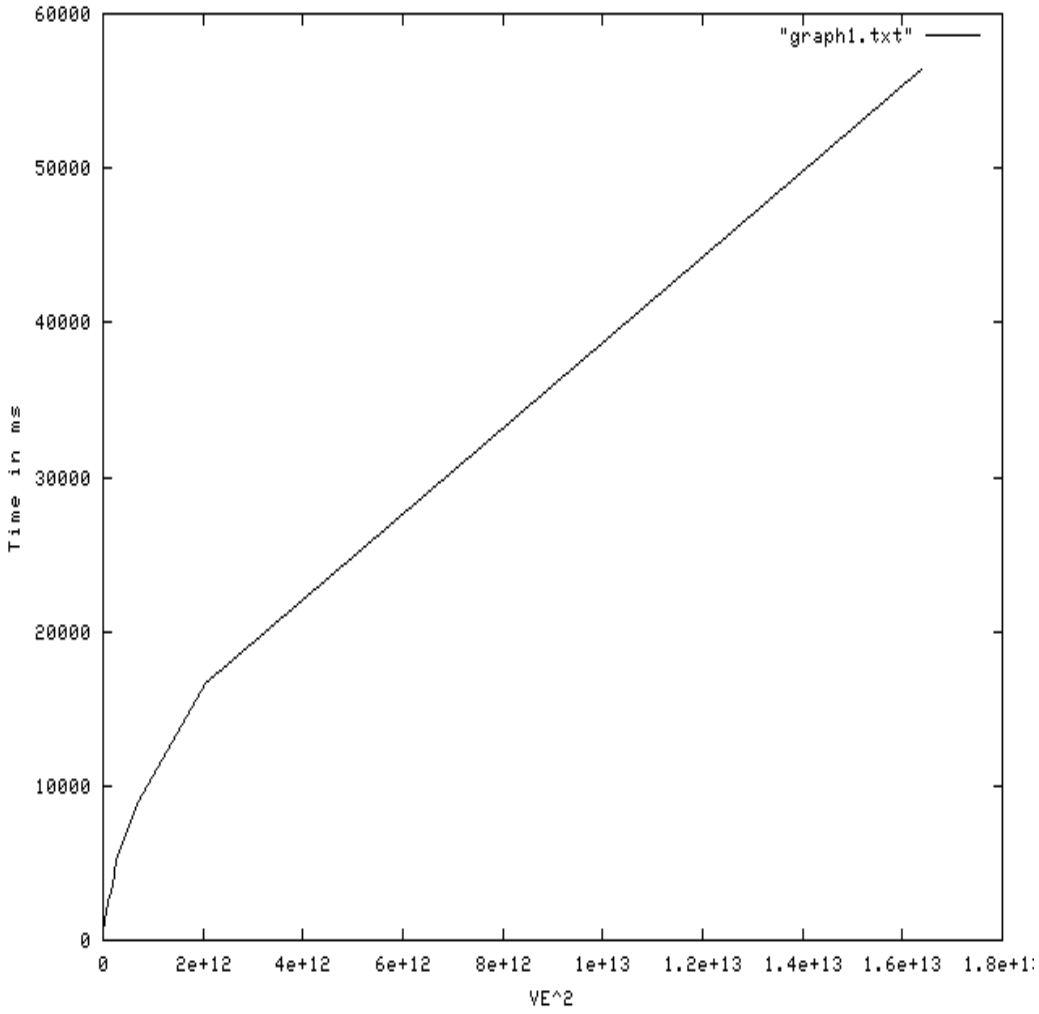
**Observations:** Rate of increase in computation time per unit increase in $VE^2$ (on a Sun Solaris machine): 2.68294e-09 Milliseconds

| $VE^2$ | **Time-Milliseconds** |
|---|---|
| 1.94872E+11 | 8686203 |
| 4.33565E+11 | 14405318 |
| 6.6248E+11 | 16263615 |
| 9.03579E+11 | 26395942 |
| 1.59031E+12 | 33465490 |
| 1.78149E+12 | 41952388 |
| 3.34884E+12 | 65954277 |
| 3.53187E+12 | 57264910 |
| 5.35824E+12 | 68732205 |
| 7.34961E+12 | 108421827 |
| 1.28487E+13 | 121616605 |
| 1.4472E+13 | 150823403 |
| 2.71748E+13 | 250152619 |
| 2.85101E+13 | 198503309 |
| 4.31005E+13 | 197291663 |
| 5.92838E+13 | 312122709 |
| 1.03297E+14 | 334086115 |
| 2.18943E+14 | 898246640 |
| 2.29105E+14 | 543809613 |
| 4.76224E+14 | 884482571 |
| 9.36844E+14 | 1239366802 |
| 9.36844E+14 | 1576837805 |
| 1.75774E+15 | 1845466334 |
| 1.75774E+15 | 2277356017 |

| **Vertices** | **Edges** | **Time-Milliseconds** |
|---|---|---|
| 1024 | 2046 | 2599 |
| 1024 | 9216 | 3127003 |
| 1024 | 6136 | 244050 |
| 1024 | 7936 | 8030 |
| 1024 | 8160 | 1457482 |
| 1024 | 9088 | 4206140 |

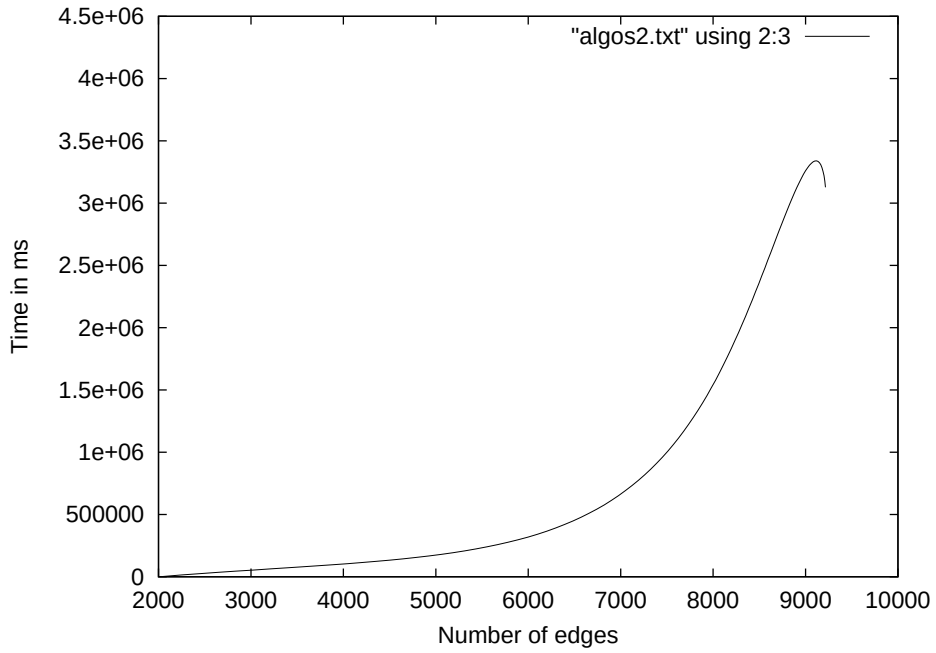
| Vertices | Edges | Time-Milliseconds |
|---|---|---|
| 4096 | 8190 | 10070 |
| 4096 | 24568 | 419455 |
| 4096 | 32256 | 35071 |
| 4096 | 32736 | 6312287 |
| 4096 | 36736 | 25277827 |
| 4096 | 37888 | 64404029 |
| 4096 | 38400 | 52177025 |

### 3.5.2 Plot of time Vs $VE^2$



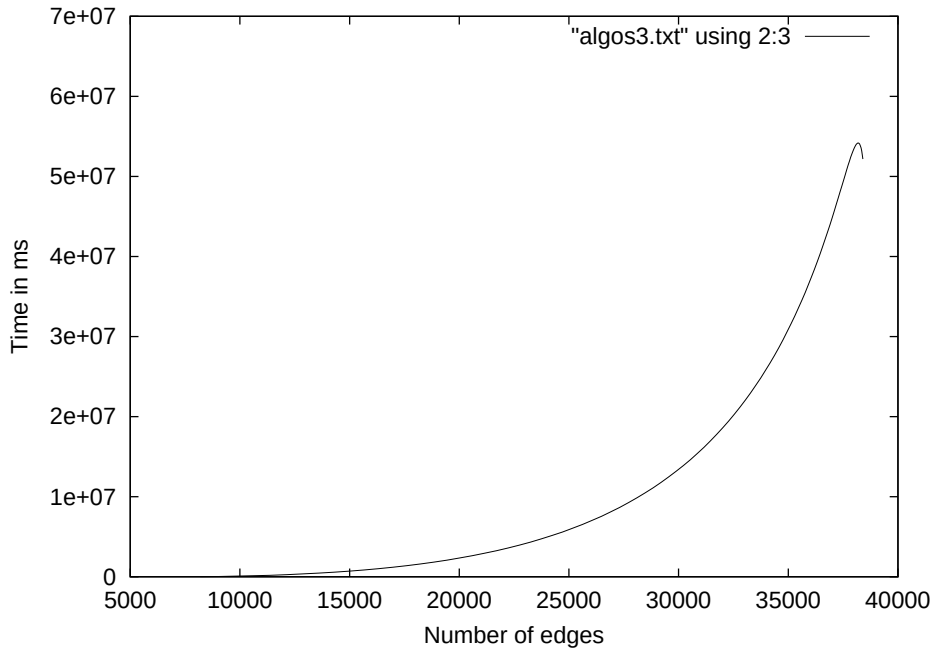

Plot: X - Axis ($VE^2$) Y - Axis (time)

### 3.5.3 Plot of time Vs $E$




Plot: X - Axis ($E$) Y - Axis (time-ms) V = 1024

### 3.5.4 Plot of time Vs $E$




Plot: X - Axis ($E$) Y - Axis (time-ms) V = 4096

The Edmonds Karp Algorithm was used for computing the maximum flow for network graphs with number of vertices ranging from 8 to tens of thousands. The network graphs were generated using *genrmf*.

The Edmonds Karp Algorithm is from the Boost library version 1.30. *genrmf* is the max-flow generator in DIMACS format. The generator produces the RMF-GEN networks developed by Goldfarb and Grigoriadis.