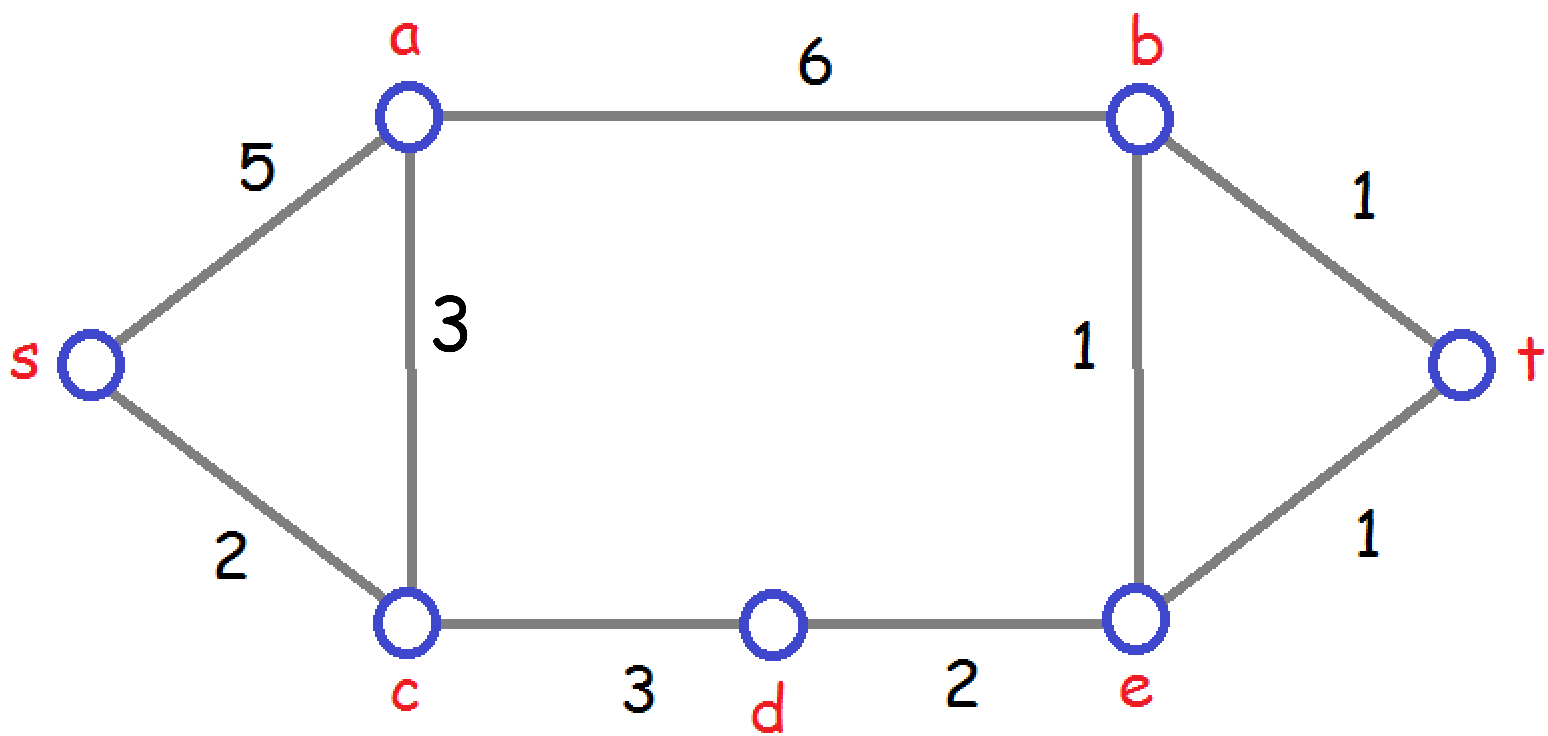
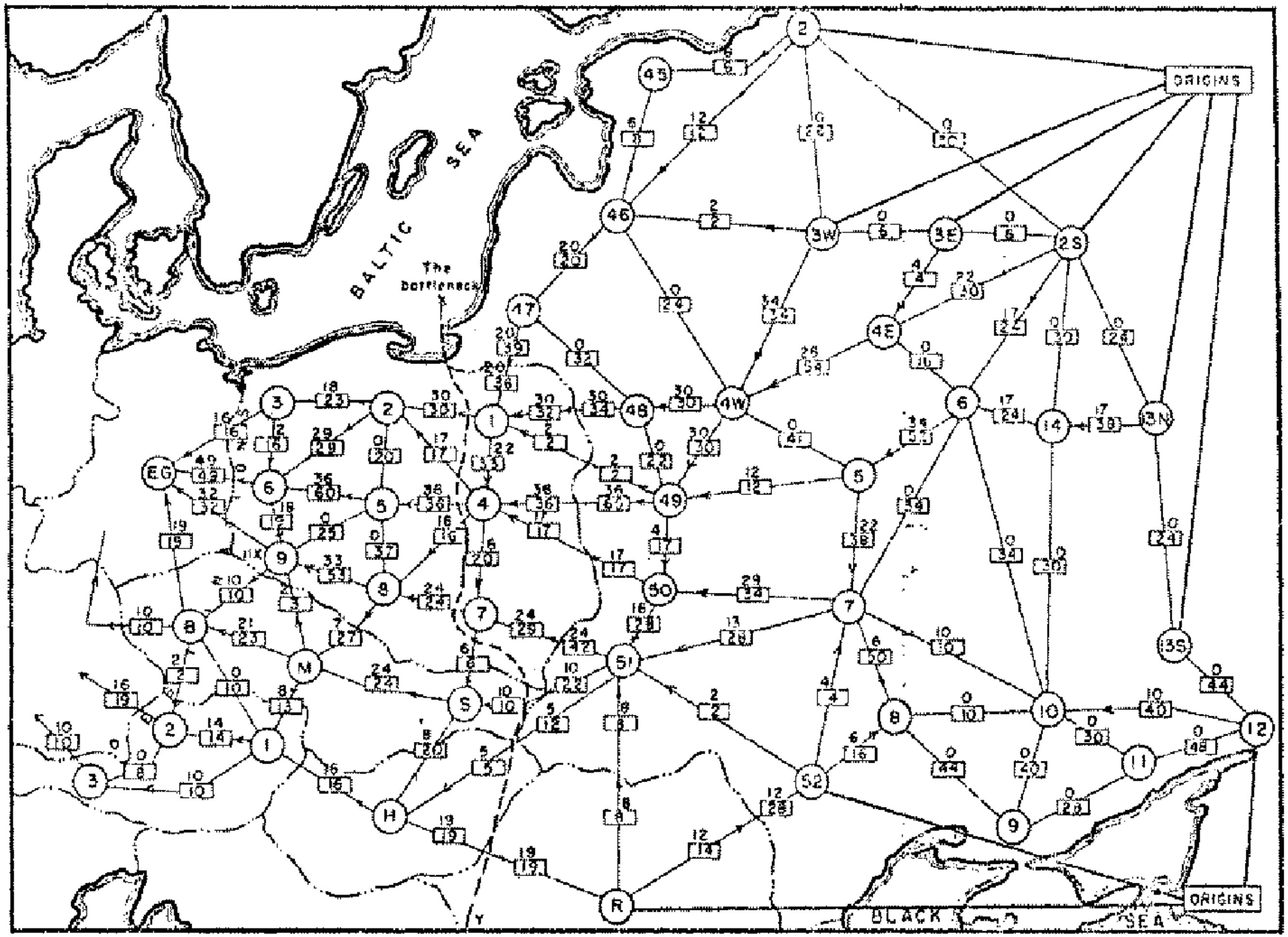
Use the Edmonds-Karp algorithm to find a maximum flow then indicate the resulting minimum cut.



Soviet Rail Network, 1955



Reference: *On the history of the transportation and maximum flow problems.*
Alexander Schrijver in Math Programming, 91: 3, 2002.

Taken from Kevin Wayne slides

Maximum Flow and Minimum Cut

Max flow and min cut.

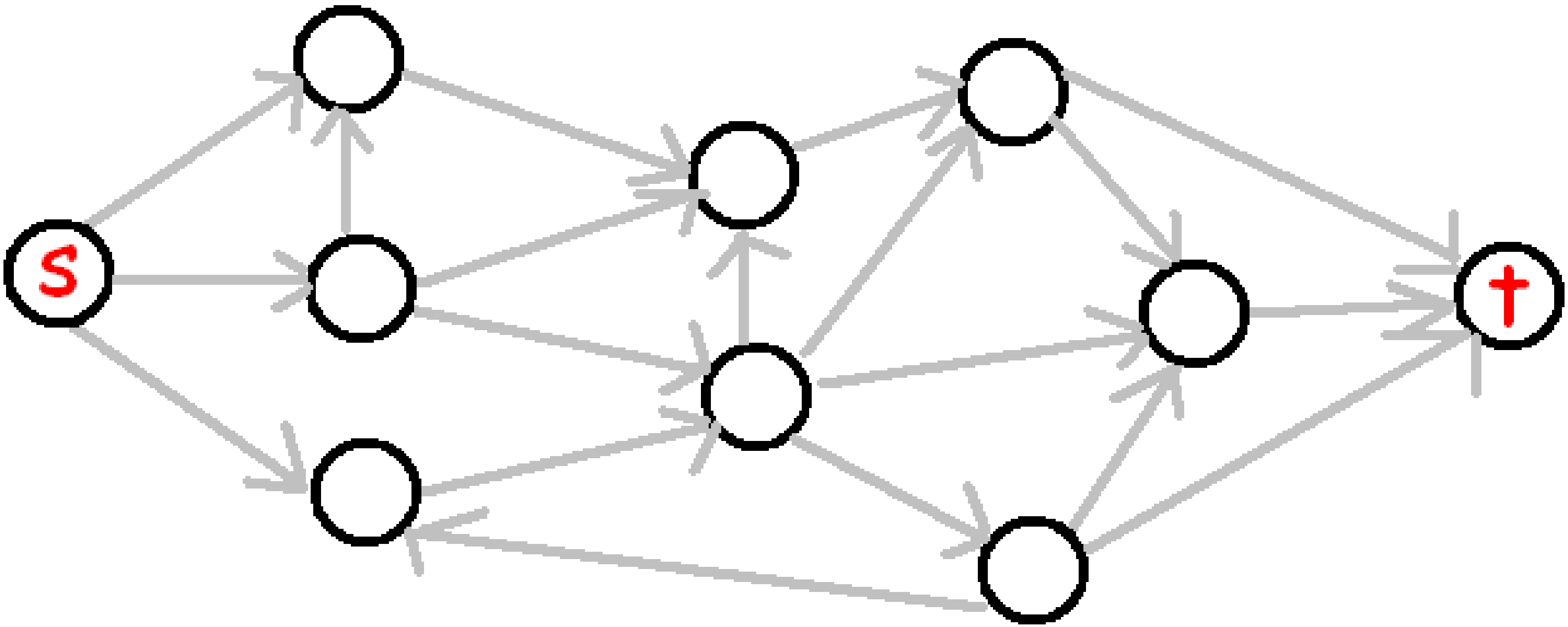
- Two very rich algorithmic problems.
- Cornerstone problems in combinatorial optimization.
- Beautiful mathematical duality.

Nontrivial applications / reductions.

- Data mining.
- Open-pit mining.
- Project selection.
- Airline scheduling.
- Bipartite matching.
- Baseball elimination.
- Image segmentation.
- Network connectivity.
- Network reliability.
- Distributed computing.
- Egalitarian stable matching.
- Security of statistical data.
- Network intrusion detection.
- Multi-camera scene reconstruction.
- Many many more …

Taken from Kevin Wayne slides

An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

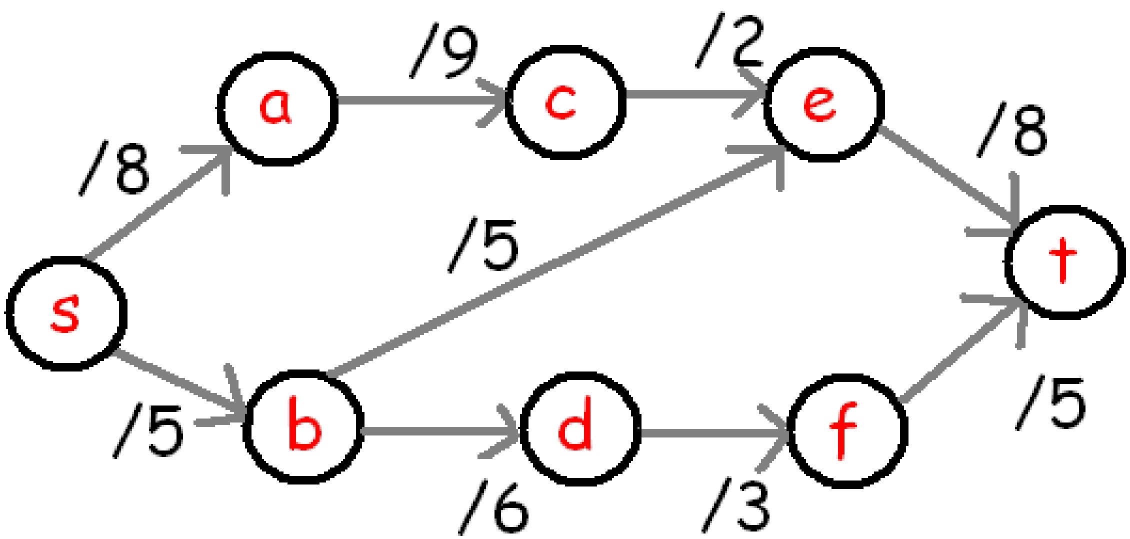


Some applications of the minimum-cut maximum-flow algorithm:

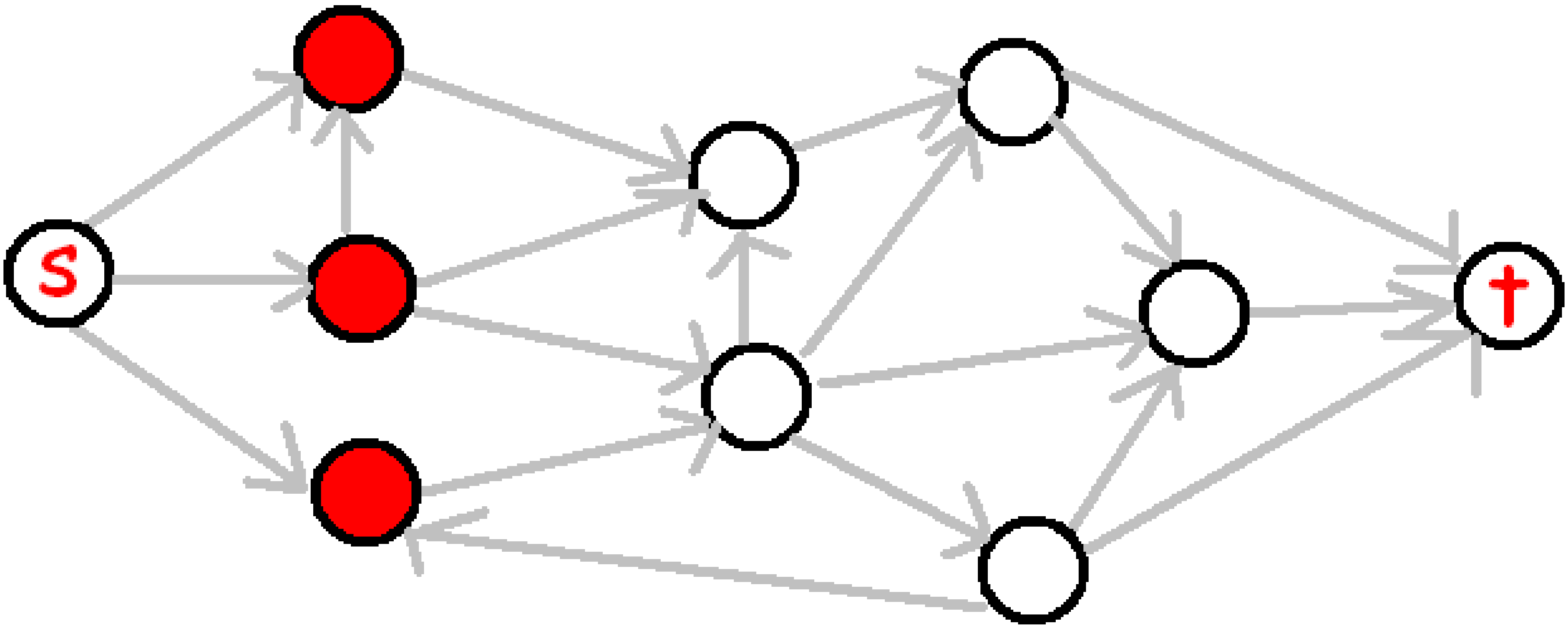
Vertex connectivity

Maximum matching in bipartite graphs

Minimum cut between each pair of nodes in an undirected graph

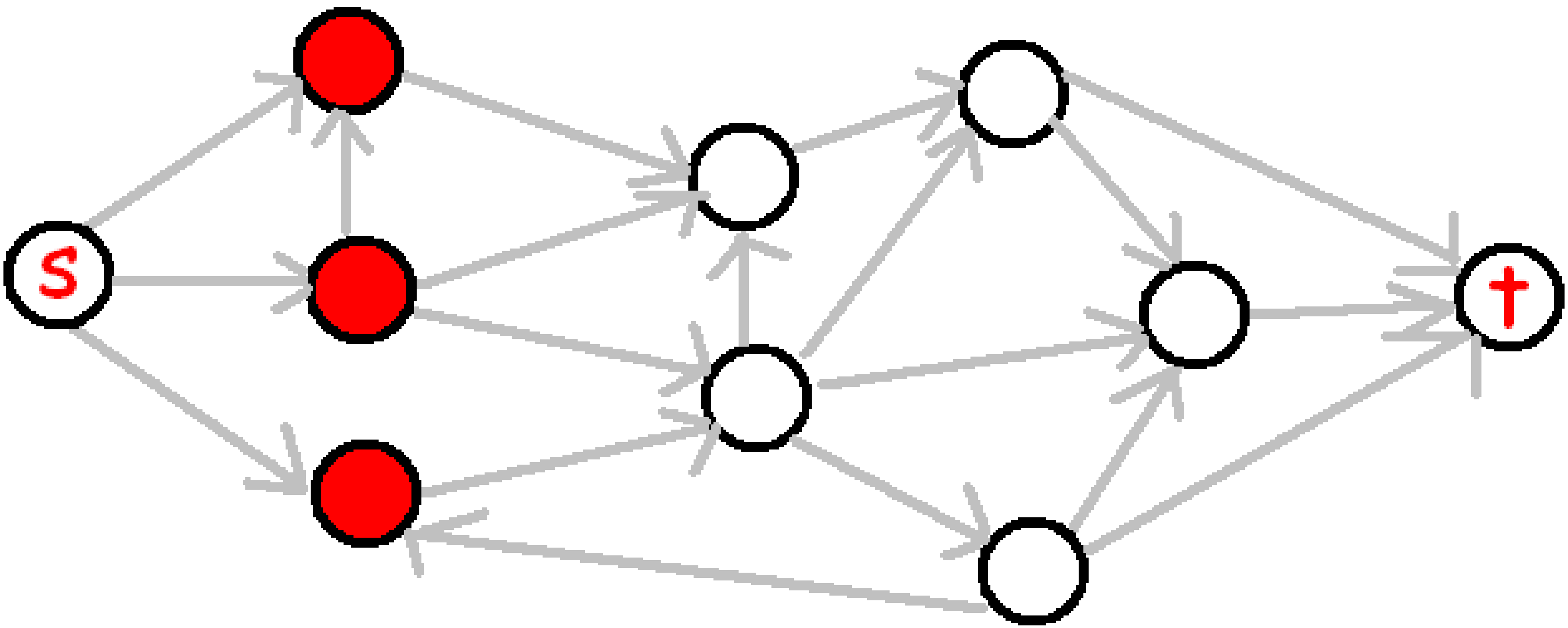


An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.



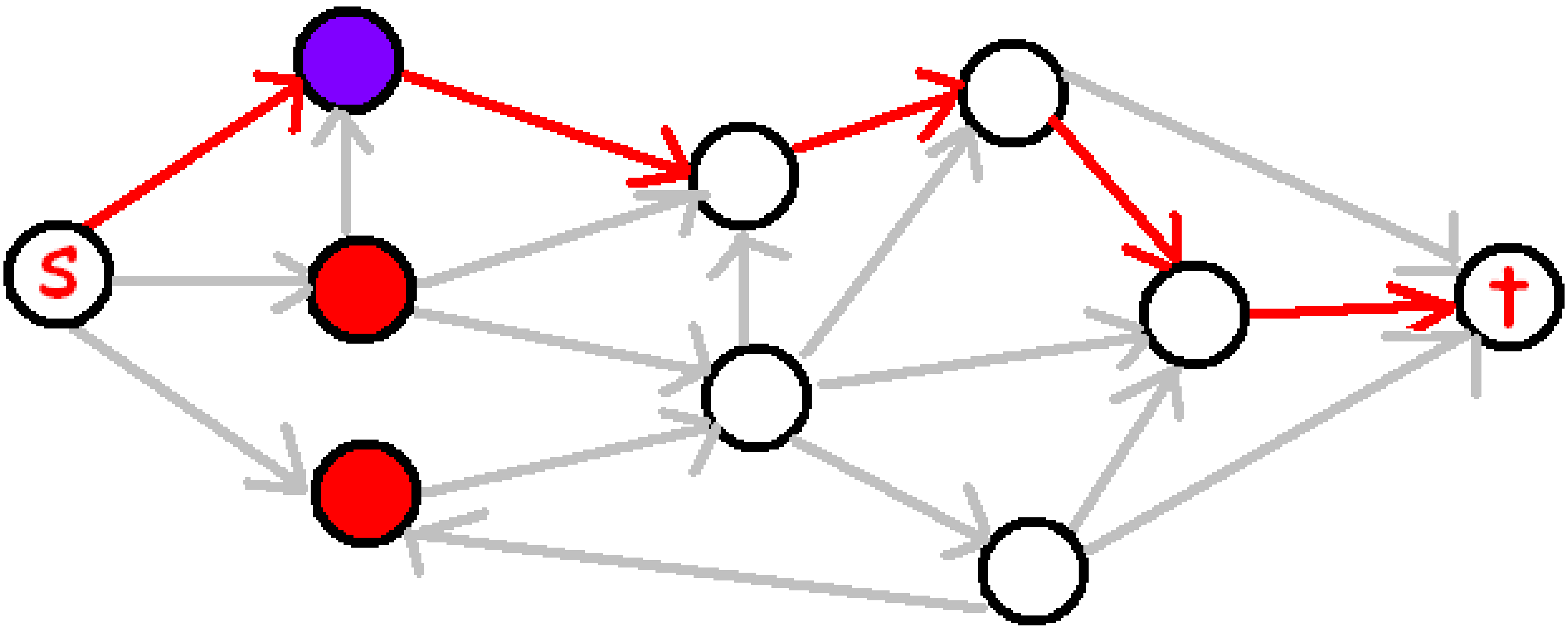
An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

A minimal vertex cut:



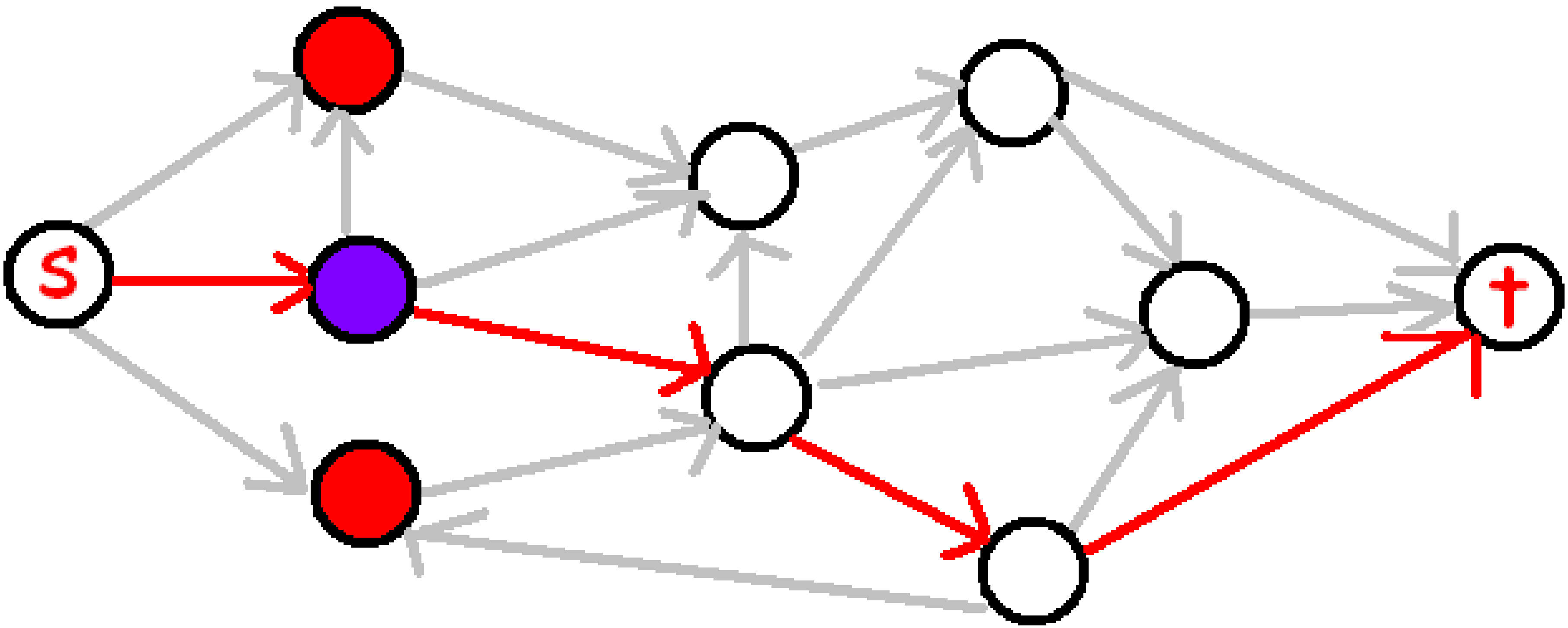
An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

A minimal vertex cut:



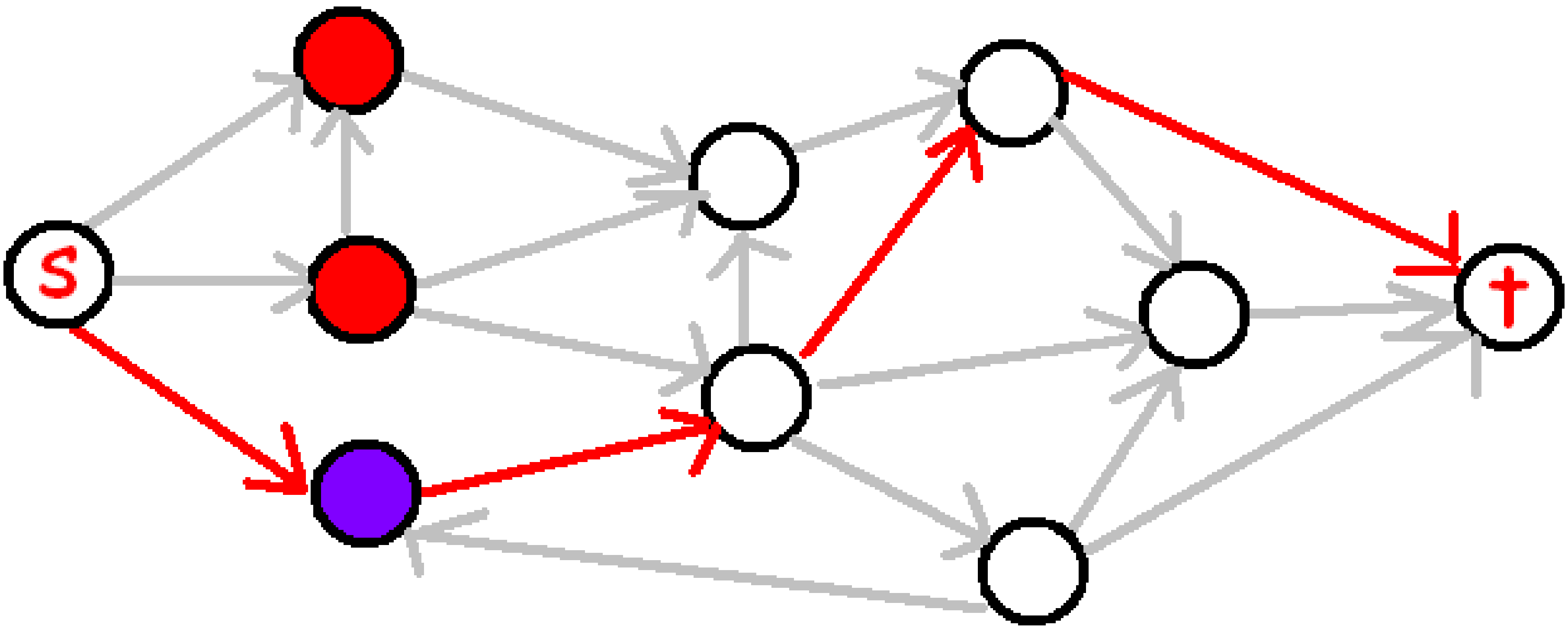
An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

A minimal vertex cut:



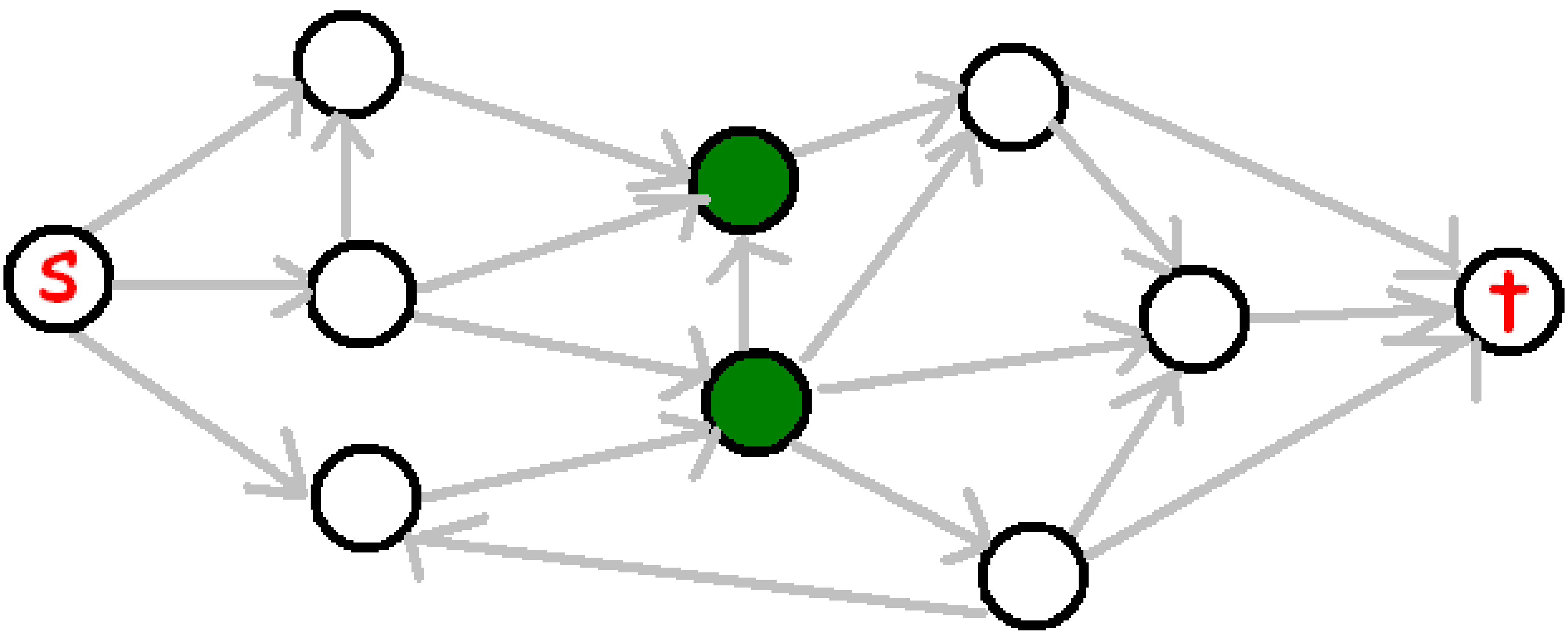
An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

A minimal vertex cut:



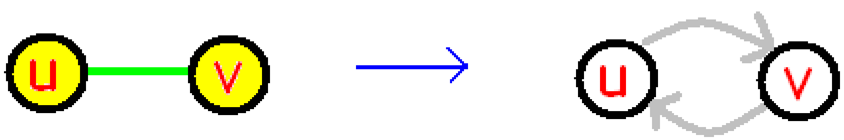
An s,t-cut: subset S of the vertices not including s or t so that G-S has no directed paths from s to t.

A MINIMUM vertex cut:



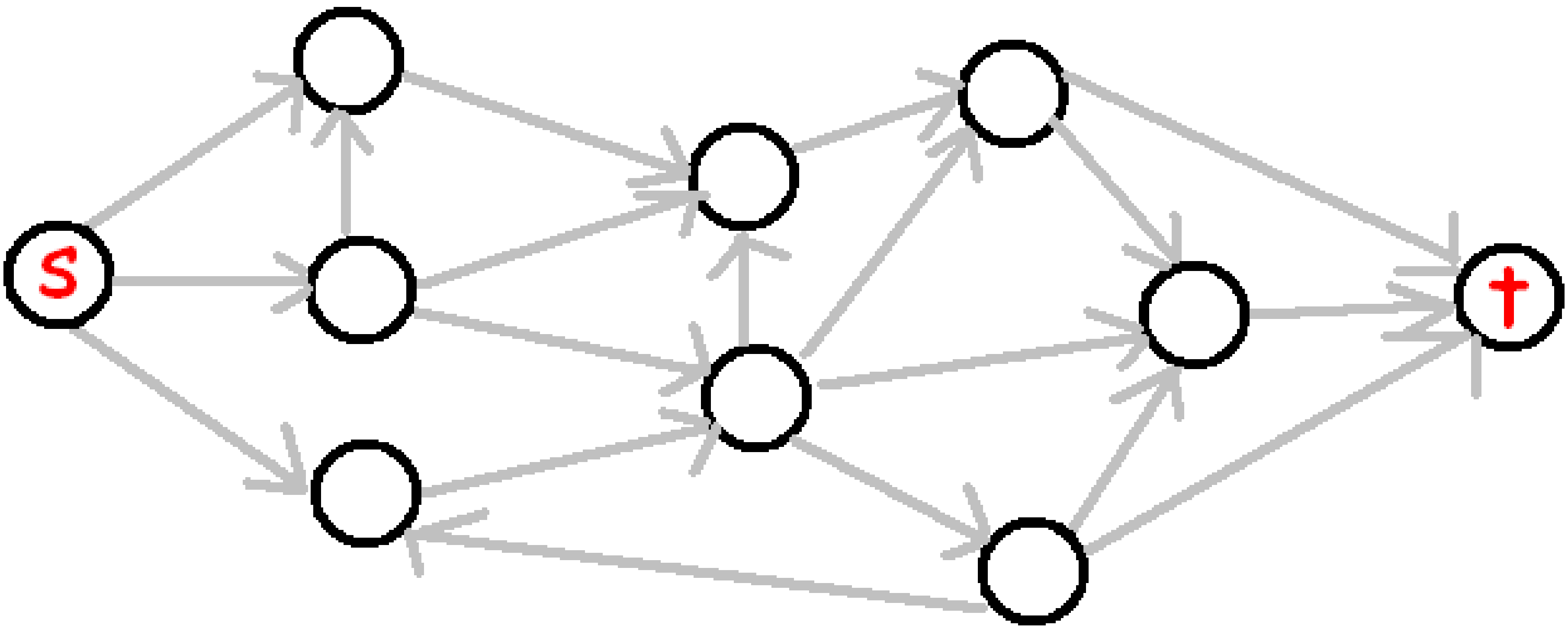
Given a directed graph G, source s, and sink t, find a minimum vertex cut between s and t.

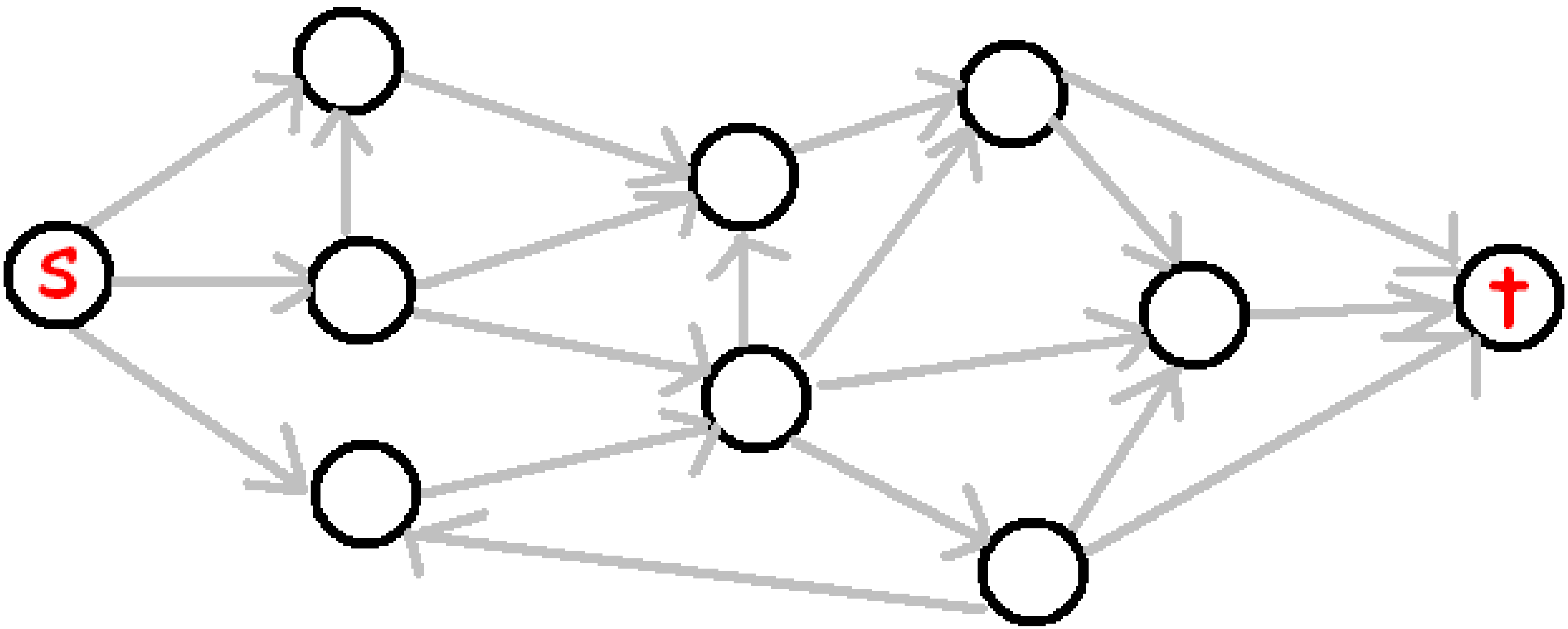
Undirected graph:

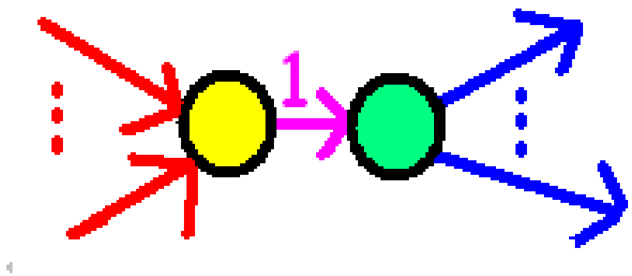


Vertex-connectivity= Minimum over all pairs s and t of the s,t-vertex connectivity in the corresponding directed graph.

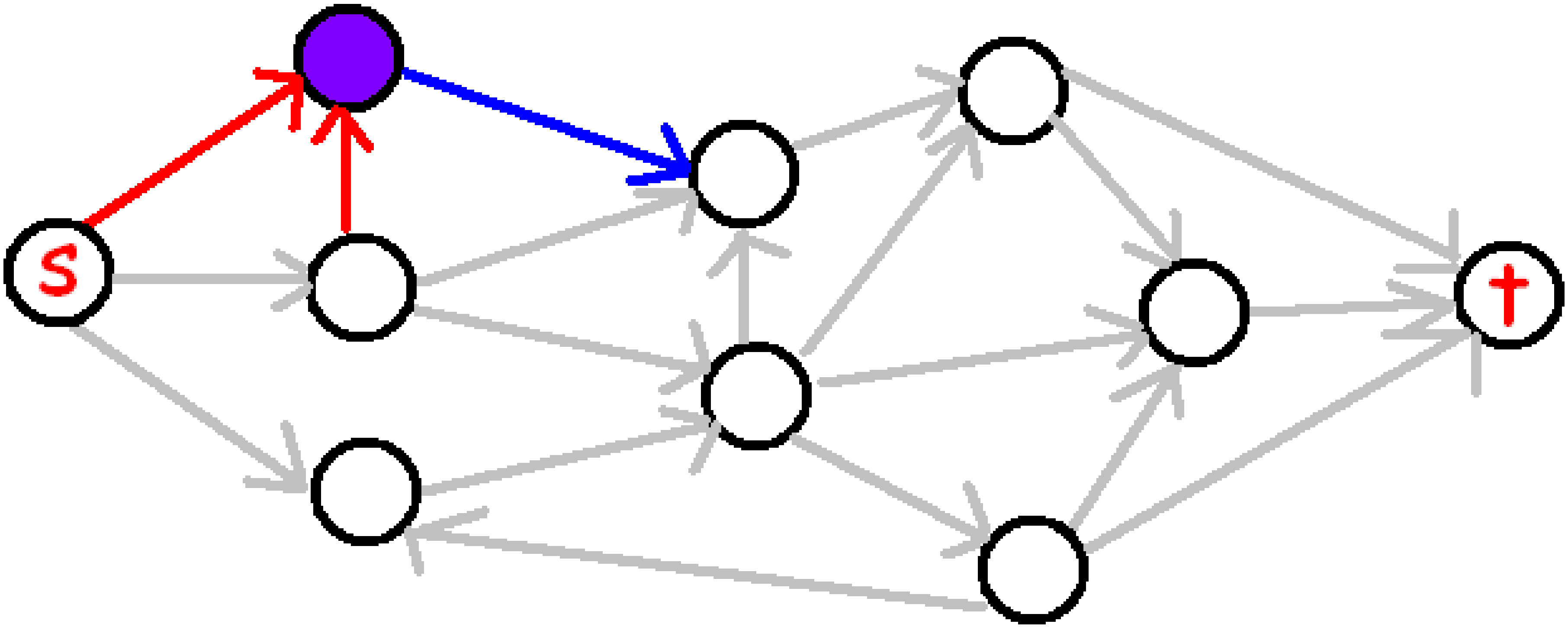
We can use min-cut max-flow to find a minumum vertex cut by first changing the network.

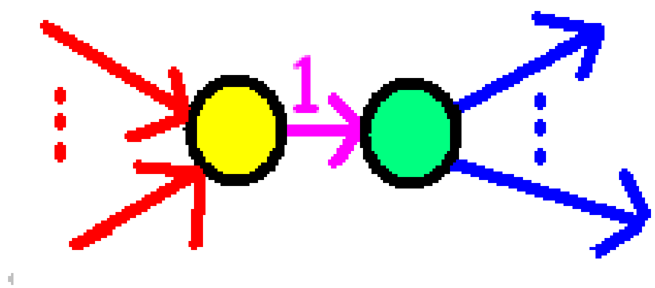


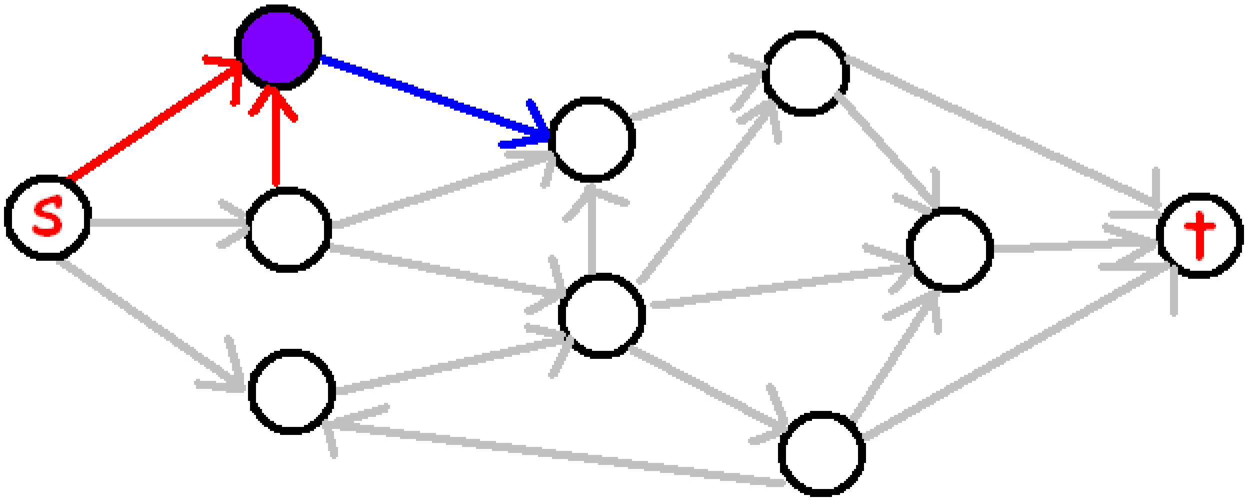


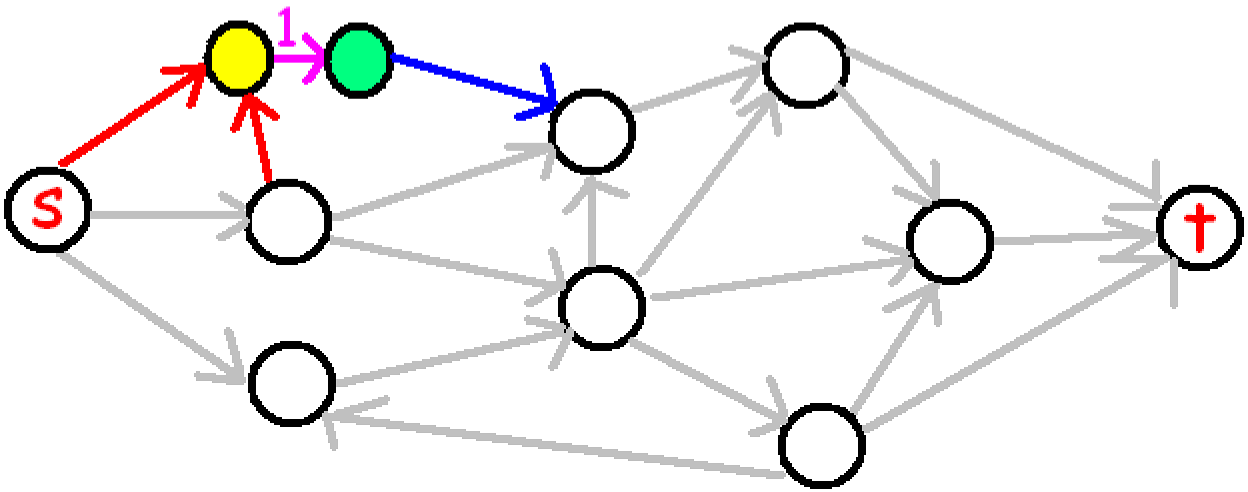


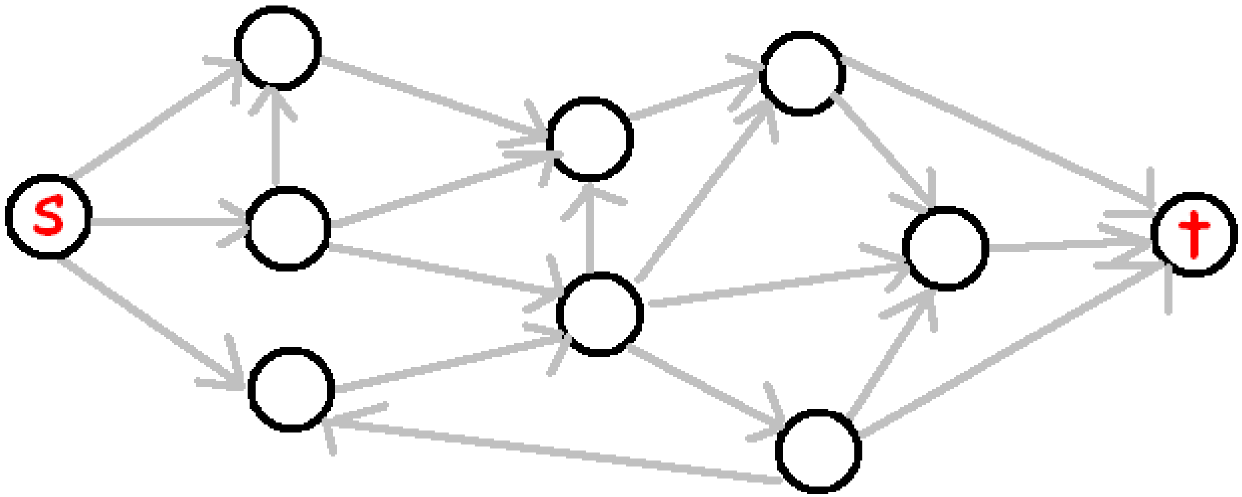
Gadget to replace vertices (but not s or t)

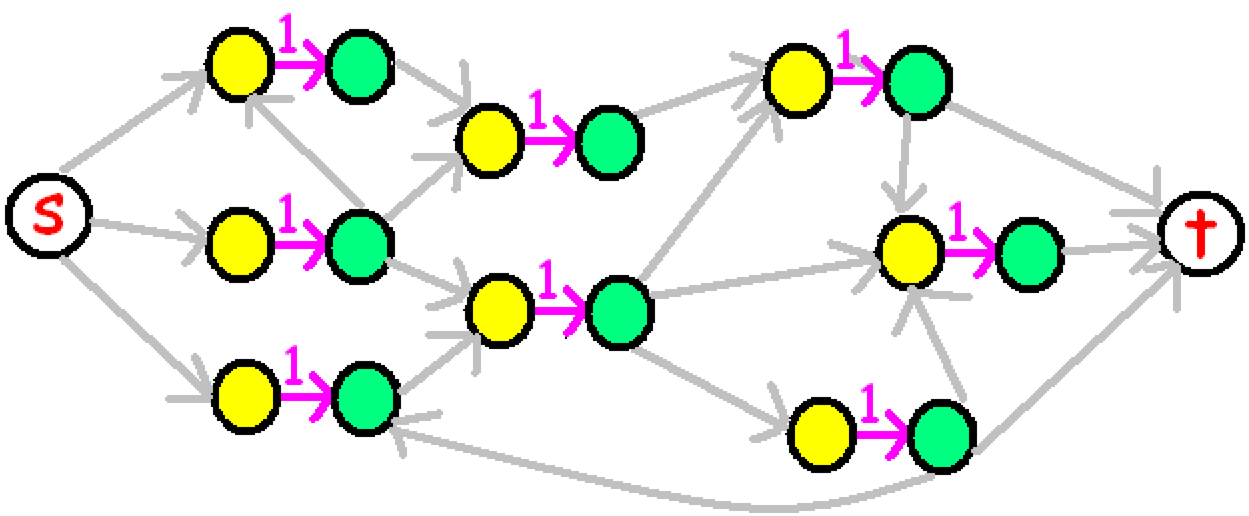

s
t

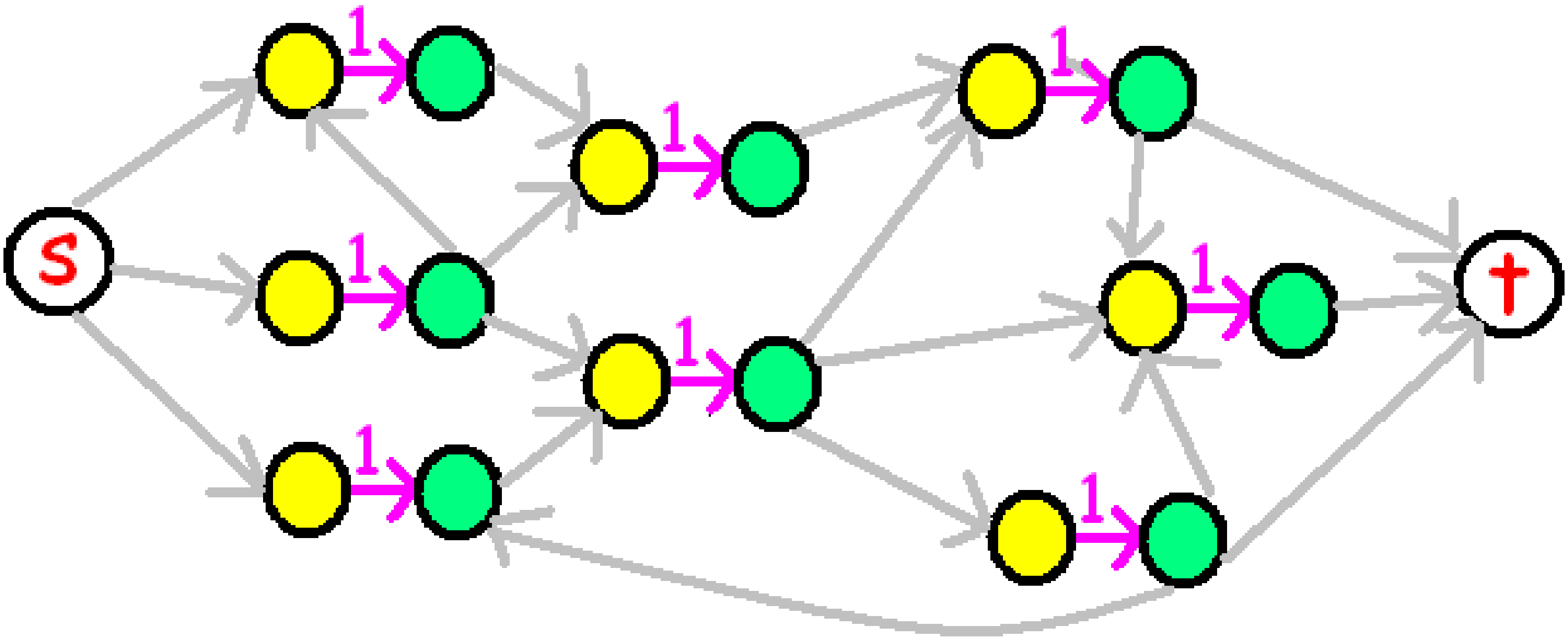

1


s
t


1
s
t

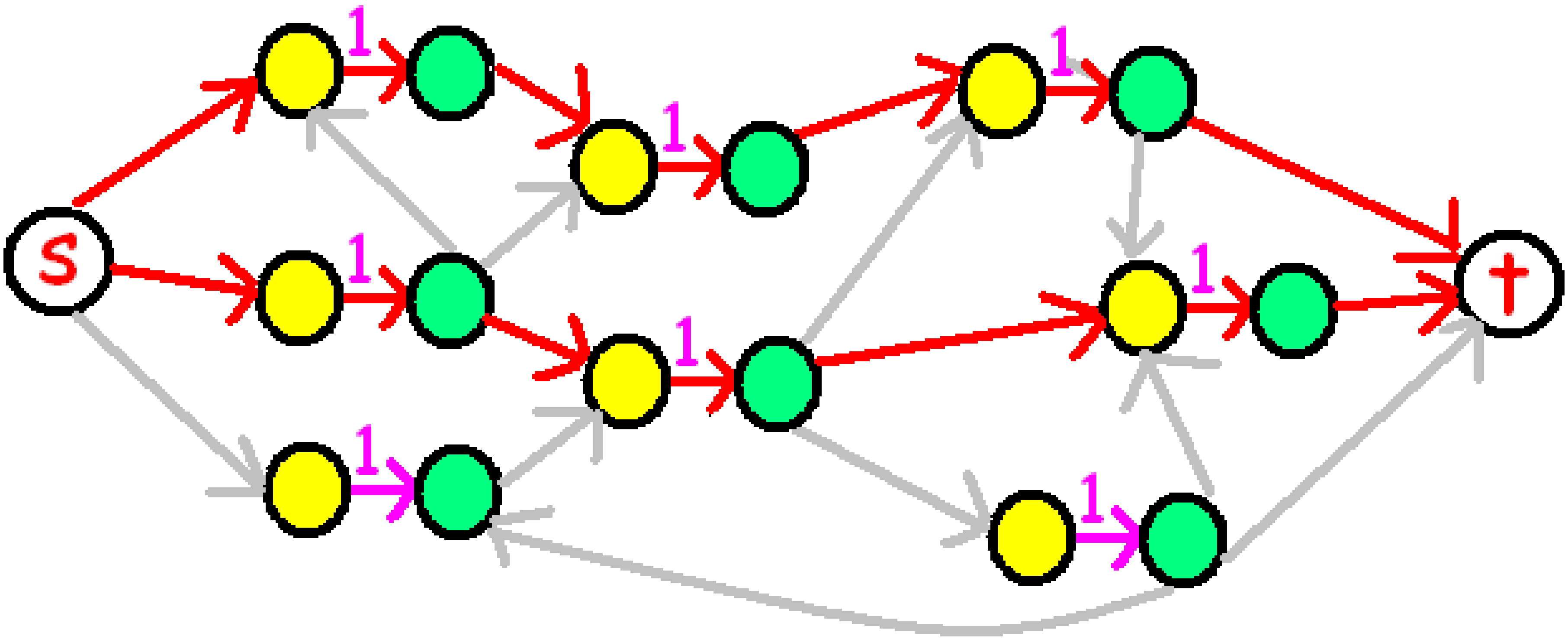

s
t


s
1
1
1
1
1
1
1
1
t

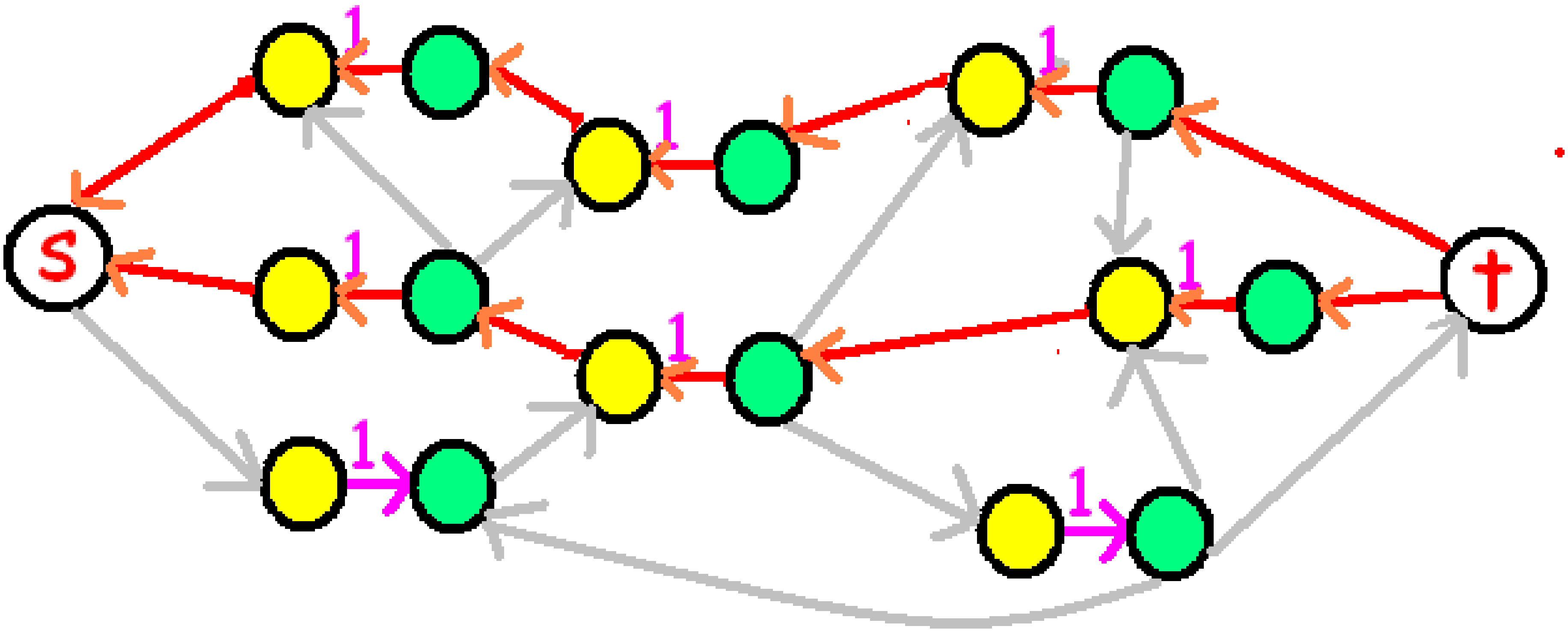


Directed Graph for maximum flow.

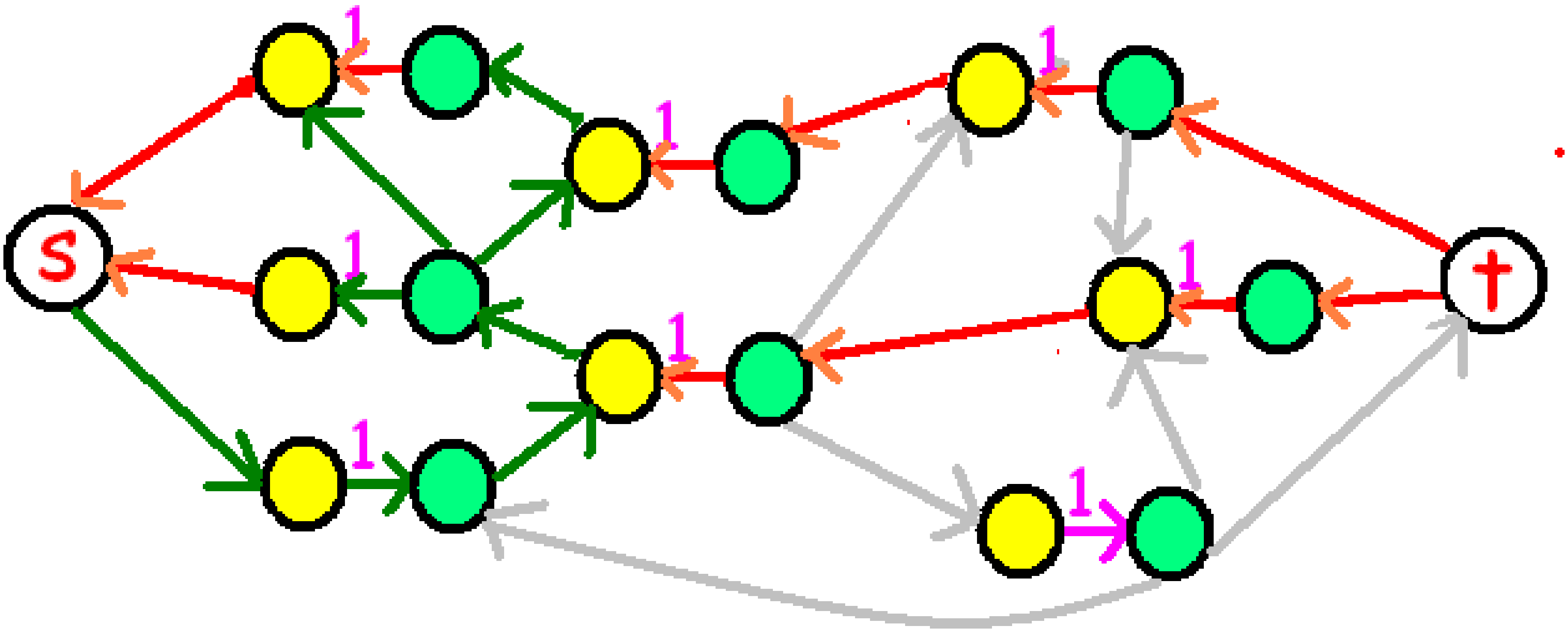
All arcs have capacity 1.



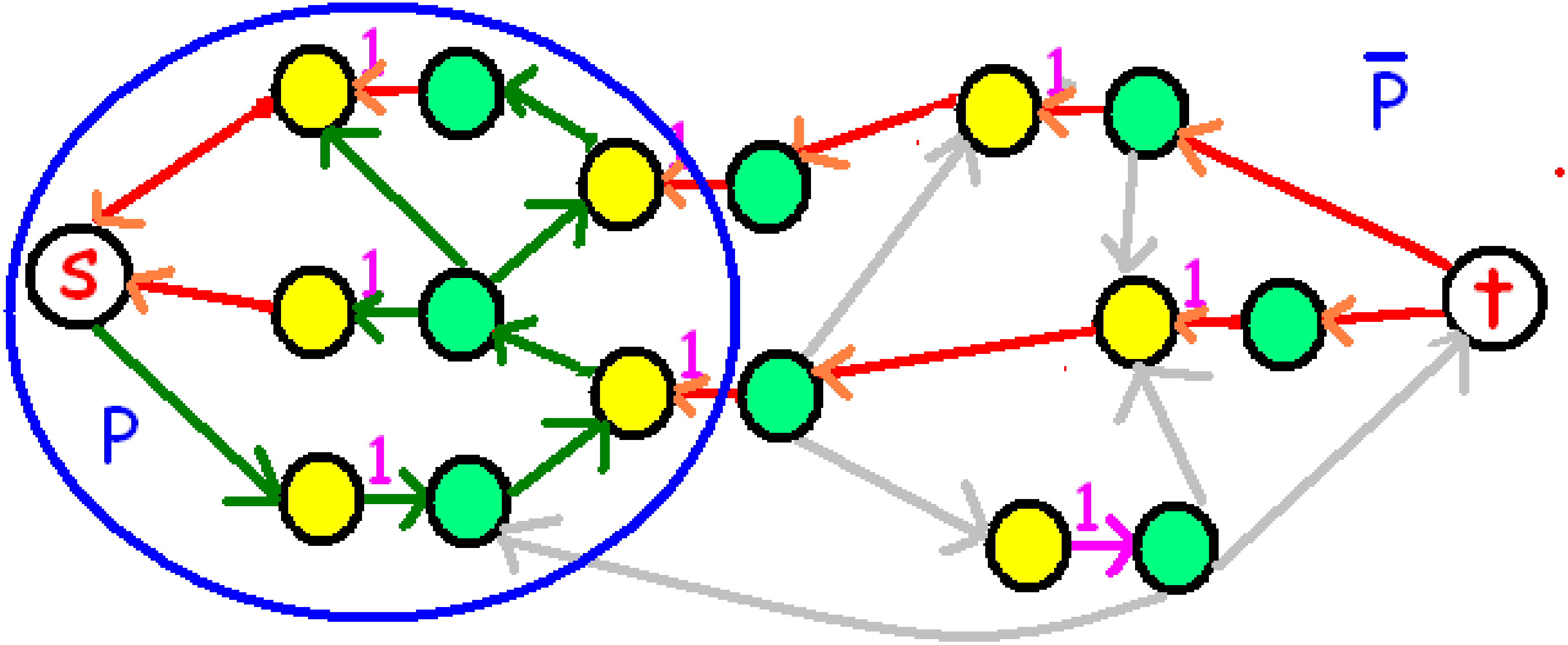
Maximum flow



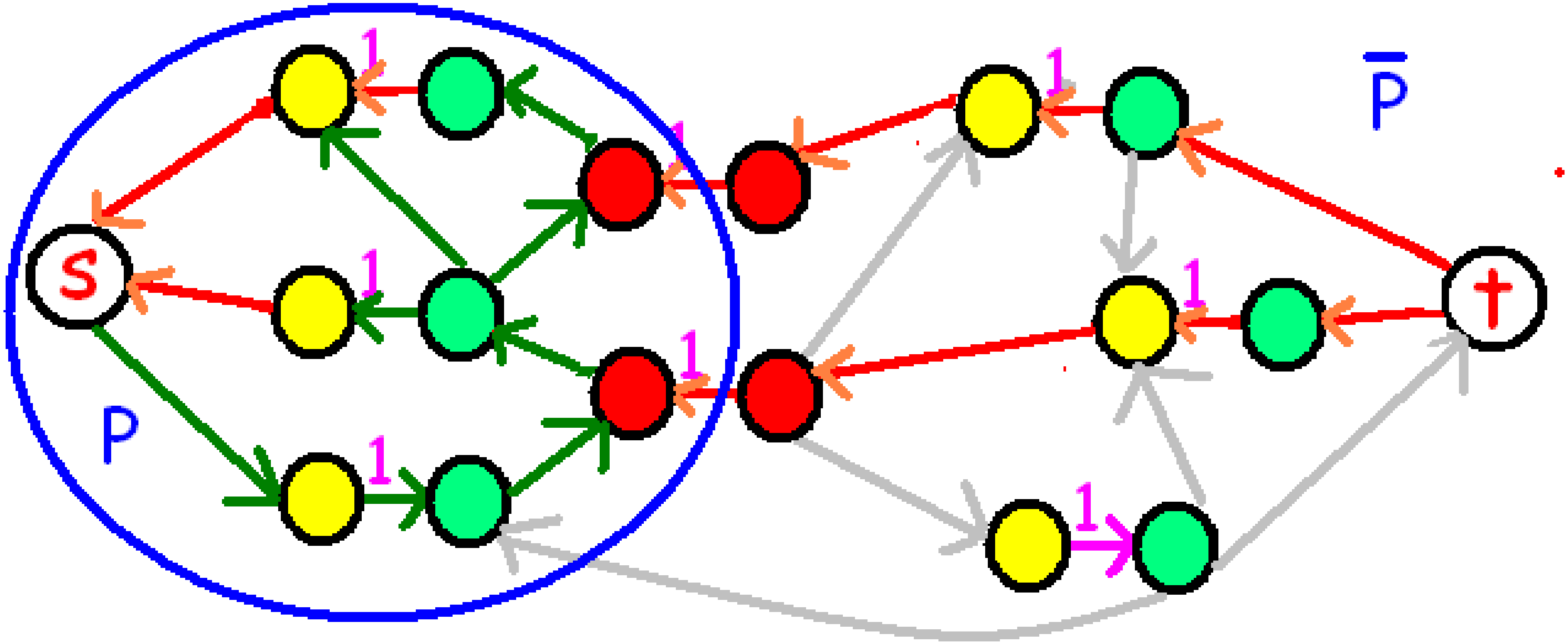
Auxillary graph.



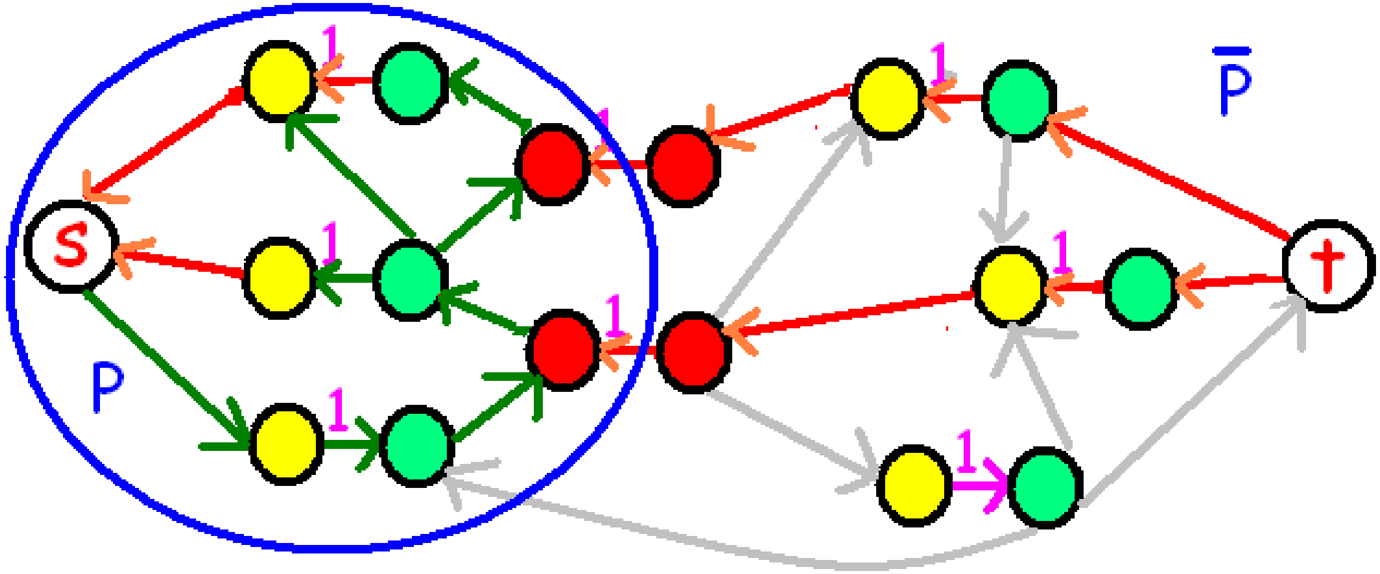
Green arcs are on BFS tree rooted at s.

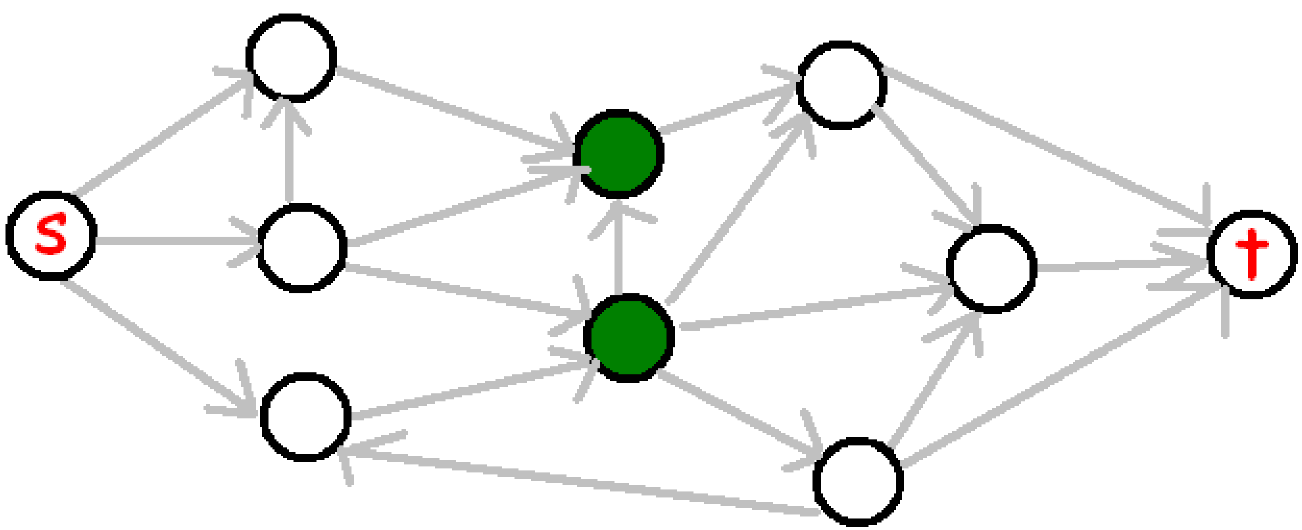


The cut in this network.



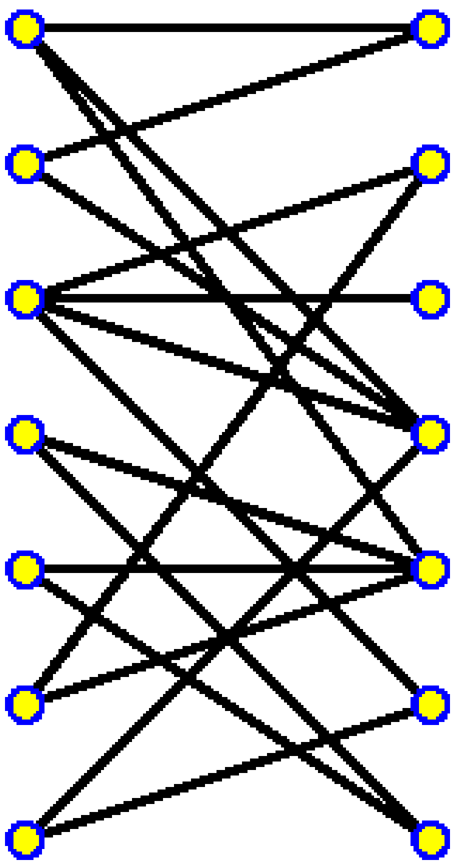
For each pink edge in the cut, its corresponding vertices are in the vertex cut of the original graph.


s
t
P
P̄
1
1
1
1
1
1
1
1
1

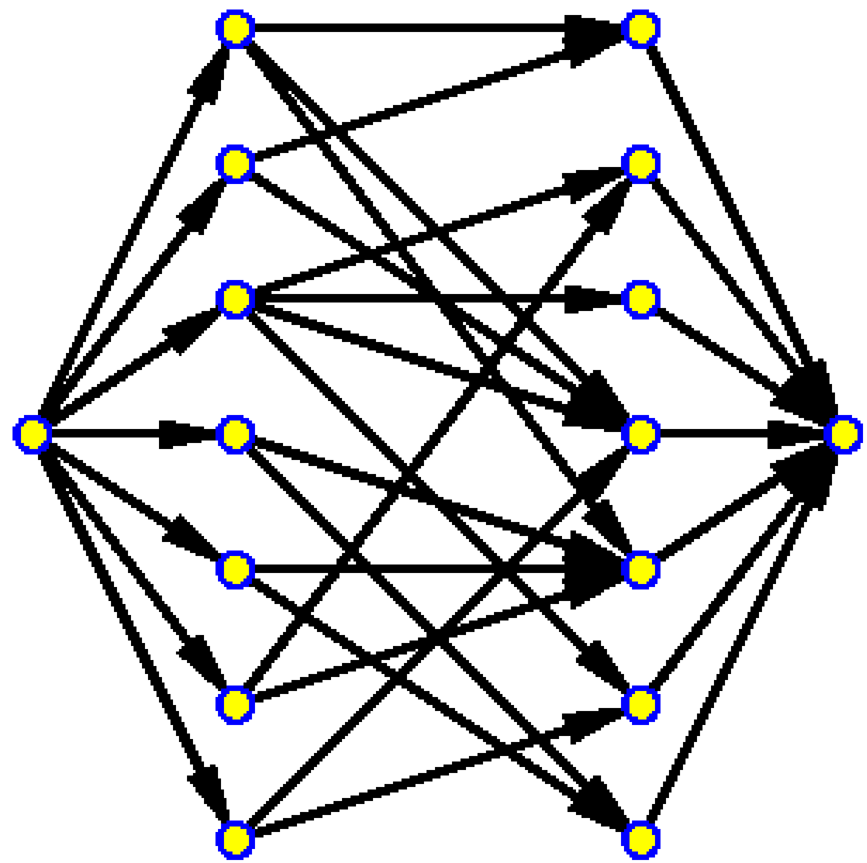

s
t

USING MAXFLOW for BIPARTITE MATCHING

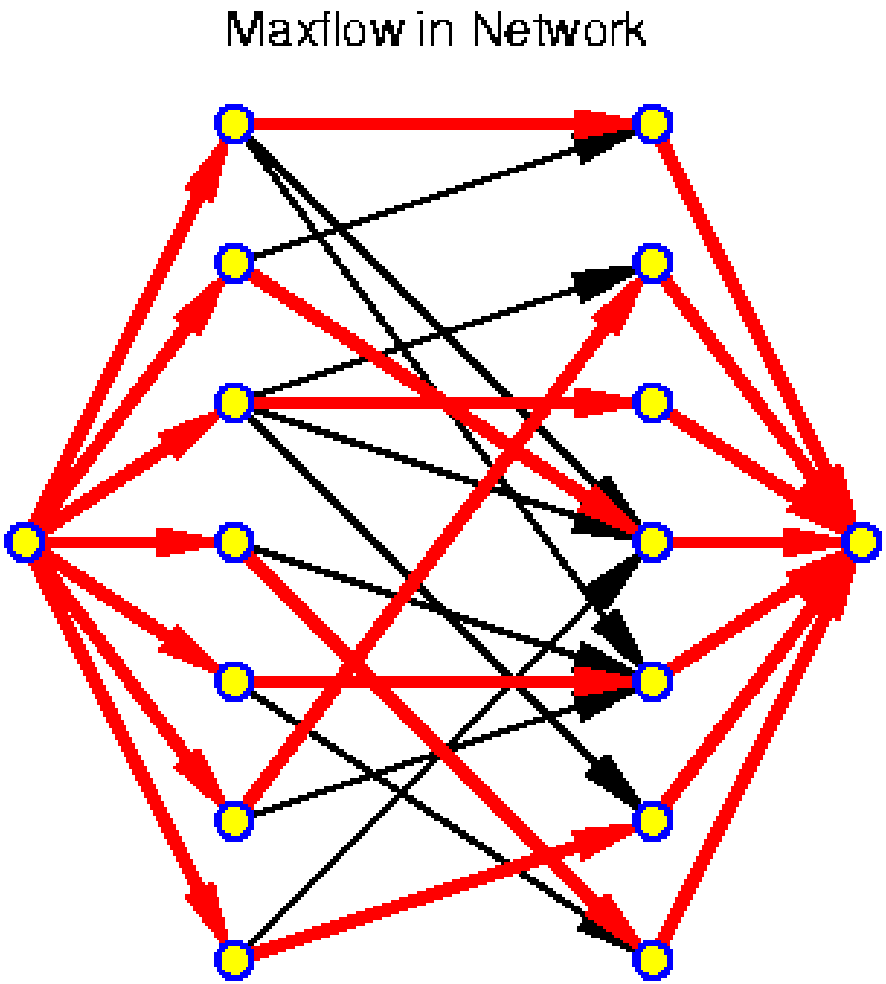
Bipartite Graph

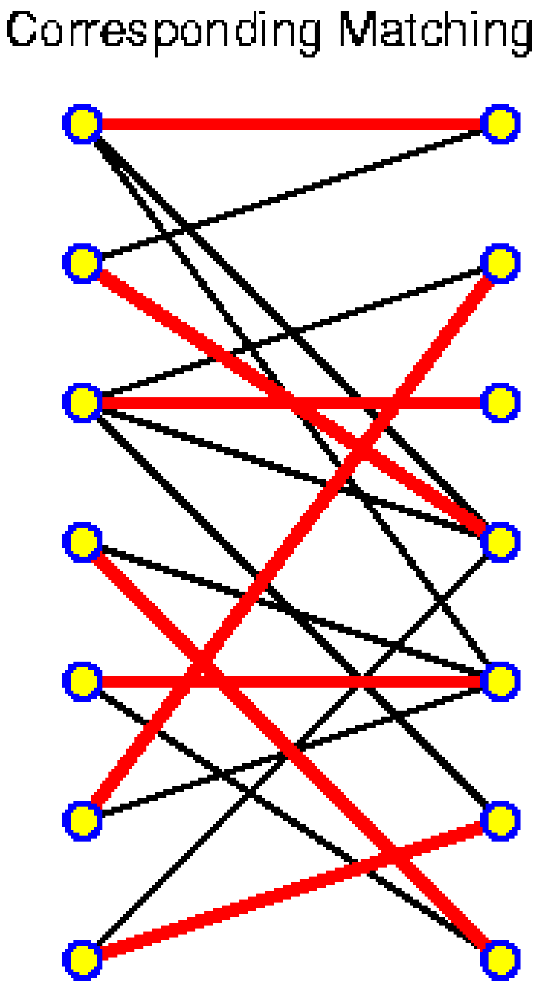


Corresponding Unit–Capacity Network



http://www8.cs.umu.se/~jopsi/dinf504/chap14.shtml





http://www8.cs.umu.se/~jopsi/dinf504/chap14.shtml

Ralph Gomery

T.C. Hu

The Gomory-Hu tree paper remains the most significant paper on multi-terminal flows since its publication in 1961.