And... not done!

Technique	Direct	With dynamic trees	Source(s)
Blocking flow	$O(V^2E)$	$O(VE\log V)$	[Dinitz; Karzanov; Even and Itai; Sleator and Tarjan]
Network simplex	$O(V^2E)$	$O(VE\log V)$	[Dantzig; Goldfarb and Hao; Goldberg, Grigoriadis, and Tarjan]
Push-relabel (generic)	$O(V^2E)$	—	[Goldberg and Tarjan]
Push-relabel (FIFO)	$O(V^3)$	$O(VE\log(V^2/E))$	[Goldberg and Tarjan]
Push-relabel (highest label)	$O(V^2\sqrt{E})$	—	[Cheriyan and Maheshwari; Tunçel]
Push-relabel-add games	—	$O(VE\log_{E/(V\log V)} V)$	[Cheriyan and Hagerup; King, Rao, and Tarjan]
Pseudoflow	$O(V^2E)$	$O(VE\log V)$	[Hochbaum]
Pseudoflow (highest label)	$O(V^3)$	$O(VE\log(V^2/E))$	[Hochbaum and Orlin]
Incremental BFS	$O(V^2E)$	$O(VE\log(V^2/E))$	[Goldberg, Held, Kaplan, Tarjan, and Werneck]
Compact networks	—	$O(VE)$	[Orlin]

Figure 10.10. Several purely combinatorial maximum-flow algorithms and their running times.

Even more known about "special" graphs:

Computer Science > Data Structures and Algorithms

Minimum Cuts in Surface Graphs

Erin W. Chambers, Jeff Erickson, Kyle Fox, Amir Nayyeri

(Submitted on 9 Oct 2019)

We describe algorithms to efficiently compute minimum (s,t)-cuts and global minimum cuts of undirected surface-embedded graphs. Given an edge-weighted undirected graph G with n vertices embedded on an orientable surface of genus g, our algorithms can solve either problem in $g^{O(g)}n\log\log n$ or $2^{O(g)}n\log n$ time, whichever is better. When g is a constant, our $g^{O(g)}n\log\log n$ time algorithms match the best running times known for computing minimum cuts in planar graphs.
Our algorithms for minimum cuts rely on reductions to the problem of finding a minimum-weight subgraph in a given $\mathbb{Z}_2$-homology class, and we give efficient algorithms for this latter problem as well. If G is embedded on a surface with b boundary components, these algorithms run in $(g+b)^{O(g+b)}n\log\log n$ and $2^{O(g+b)}n\log n$ time. We also prove that finding a minimum-weight subgraph homologous to a single input cycle is NP-hard, showing it is likely impossible to improve upon the exponential dependencies on g for this latter problem.