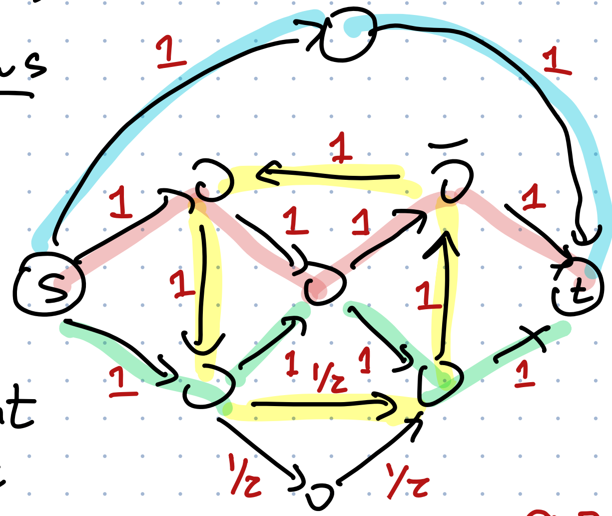
Applications and Extensions of Max Flows

Edge-disjoint paths

Given directed graph
two nodes s and t
Find max # edge-disjoint paths from s to t



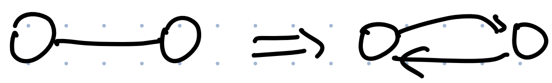
Solution: ① Give edge cap. 1

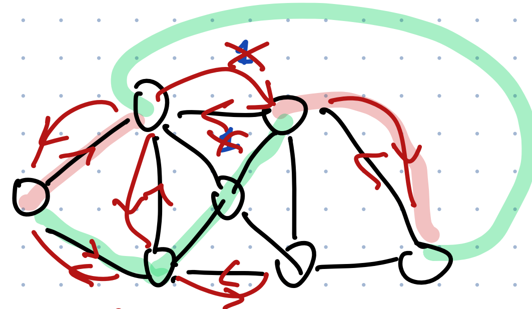
② Compute max flow f^* using FF (guarantees integer flow)

③ Decompose f^* into paths and cycles
discard cycles

Running time: $O(E \cdot |f^*|) = O(EV)$ time

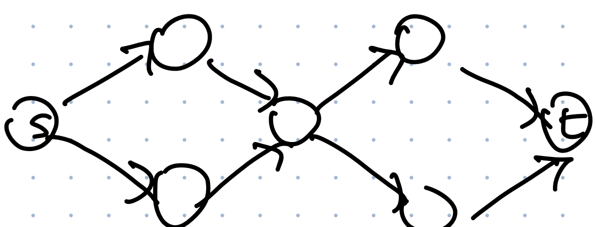
What if G is undirected?

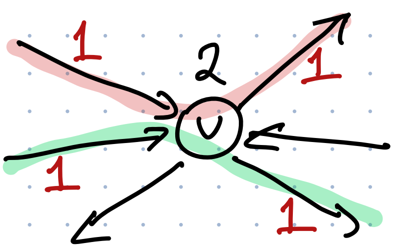


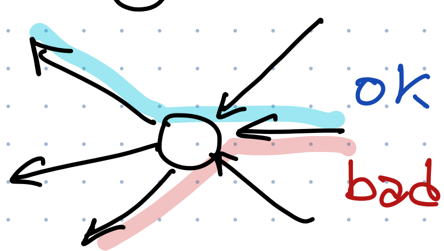


Split undir edges into dir ← $+O(V+E)$
Cancel 2-cycles before decompose ← $+O(E)$

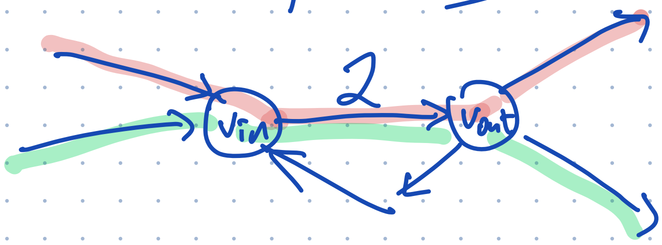
Vertex-disjoint paths?   Vertex capacities?

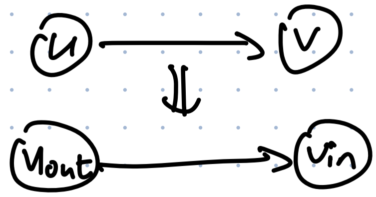






Vertex splitting



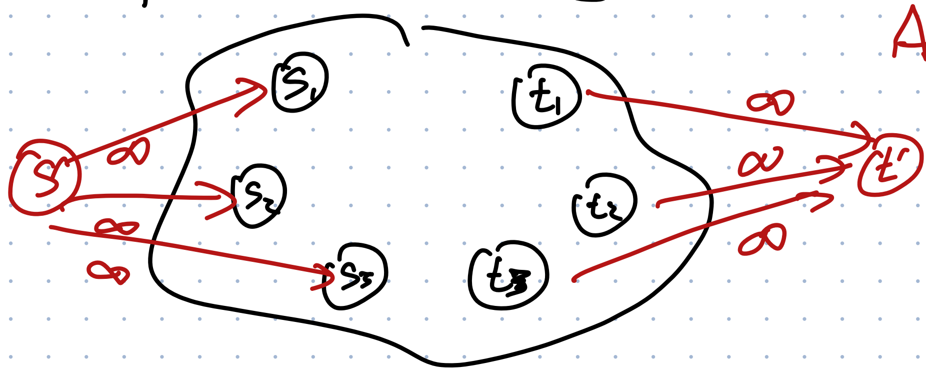


max flow from s_{out} to t_{in}

$O(VE)$ time to find vertex-disjoint paths.

$c'(v_{in} \to v_{out}) = c(v)$

Multiple sources + targets

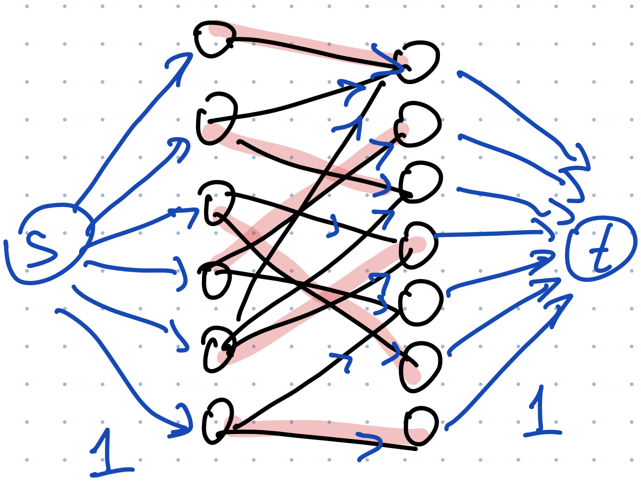


Add super source s'
edges $s' \to s_i$
super-target t'
edges $t_j \to t'$

Flow/paths can go from any s_i to any t_j

all new edges have ∞ capacity

Bipartite matching



Add source s edges $s \to l_i$
target t edges $r_j \to t$
direct $l_i \to t_j$

Find edge-disjoint paths
$O(VE)$ time