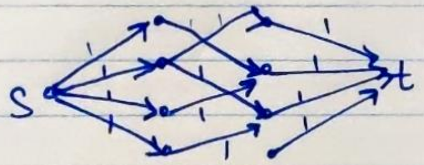
# Flows

① Naive : choose 2 vertices & find min-cut
$O(n^2) \times T_{max\,flow}$

Observe : Choosing 1 vertex & computing min cut for all remaining vertices wrt chosen vertex ~~also~~ suffices ~~gives~~ for global min-cut

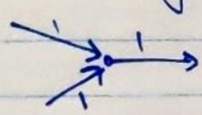
consider global min cut $A, A^c$. $v \in A$ or $v \in A^c$; where $v$ is the chosen vertex $\Rightarrow$ $n$ maxflow computa$^n$ are sufficient

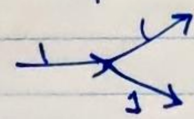
⑧ Hopcraft - Karp Alg




Add s, t
All Edges have wt 1

Run $\sqrt{V}$ phases of Dinic's $\Rightarrow E\sqrt{V}$
↳ Let the flow after this be f, residual graph be $G_f$
Max flow in this construc$^n$ has vertex disjoint flow paths
The following are not possible (integral flow) & flow cons$^n$





$dist_{G_f}(s,t) \geq \sqrt{V}$ {after $\sqrt{V}$ iter$^n$ of Dinic's}

At most $\frac{V}{\sqrt{V}} = \sqrt{V}$ vertex disjoint paths $\Rightarrow$ Max flow in $G_f \leq \sqrt{V}$

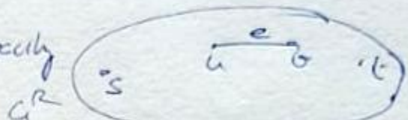
Run ford fulkerson from here (on $G_f$)

Q2 When we delete an edge e, the maximum flow might reduce by as much as the flow through the edge e. However it might reduce by a lesser amount.

The only way to solve this problem is to compute maximum s-t flow after deleting each edge in G. Thus the total time required would be $O(m T_{max\text{-}flow})$ where T is the time to compute a max s-t flow

Q3) No, statement is not true. In the example below increasing capacity of any one edge does not increase max-flow


Q4)(a) Let $e=(u,v)$ be the edge of whose capacity is increased by 1. We are given a maximum flow f. Compute the residual graph $G^R$ with respect to f. If in $G^R$ there exists a path from s to t then the maxflow would increase by 1 (note all edges in $G^R$ have integer capacities). Since the maxflow can only increase by atmost 1, we have computed the new maxflow. Time required for finding a path from s to t is $O(m+n)$ & time required for computing $G^R$ is also $O(m)$




Q4) (b). Let f be the maxflow & $G^R$ the residual graph w.r.t f. Let $e=(u,v)$ be the edge whose capacity is reduced by 1. If edge e is not saturated then there is no change to the maxflow f. However if e is saturated (residual capacity is 0) then we need to decrease the flow through e by 1 unit so as not to violate capacity constraints.

In $G^R$ find a path from u to v. If there is such a path we can increase flow along this path by 1 unit & decrease flow along u,v by 1 to get a flow of same value as before. This procedure takes $O(m+n)$ time.

If there is no path from u to v in $G^R$, then find a path from u to s & from t to v (these paths always exist since at least one unit of flow goes from s to u & from v to t. Reduce flow along these paths by 1 unit to obtain the new maxflow. This step also takes $O(m+n)$ time.

Q5) done in class on 21 Oct

Q6) A phase is the sequence of iterations in which the shortest path from s to t remains unchanged. Consider a BFS from s & the levels obtained. In phase i the shortest path from s to t is of length i & so t is at level i.

In this phase we only consider edges that go from one level to the ~~this~~ next since only these edges are on a shortest path from s to t.

E' is this set of edges

Remove all vertices which are at the same level as t (except t) & all vertices at levels beyond t.

repeat
- Keep the outdegree of each vertex in E'
- If some vertex has outdegree 0 remove it & its incoming edges

until all vertices have ~~zero~~ non-zero outdegree.

Start from s, follow any edge to reach a vertex at level 1, follow any edge out (this vertex has outdegree > 0) to reach a vertex at level 2 & so on till you reach t.
Remove all edges on this path & update outdegree of vertices, removing vertices with outdegree 0.

Repeat till no path left from s to t.

Running time: Once we touch an edge, we remove that edge. Hence total time spent in this phase is $O(m)$.

Q7) In a unit capacity graph we first run the above algorithm for each phase till the length of the shortest path from s to t in the residual graph is ~~atd~~ $> \sqrt{m}$.
The no of phases we run the above algorithm is $\sqrt{m}$ & so the required is $O(m\sqrt{m})$.

When length of shortest path from s to t is $\sqrt{m}$ then in ~~the~~ any maximum flow the length of each path carrying ~~every flow path~~ non-zero flow is $\geq \sqrt{m}$. Since total capacity of all edges is m the maximum flow that can be routed is atmost $m/\sqrt{m} = \sqrt{m}$.
Hence by running $\sqrt{m}$ iterations of ford fulkerson we can send the remaining flow. This takes $O(m\sqrt{m})$ time.