CSE 421
Algorithms

Lecture 21

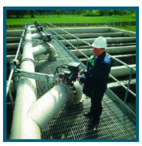
Network Flow, Part 1

Announcements

- Sample finals posted
- No questions planned on NP-completeness

Network Flow





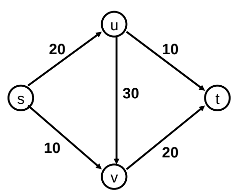

Outline

- Network flow definitions
- Flow examples
- Augmenting Paths
- Residual Graph
- Ford Fulkerson Algorithm
- Cuts
- Maxflow-MinCut Theorem

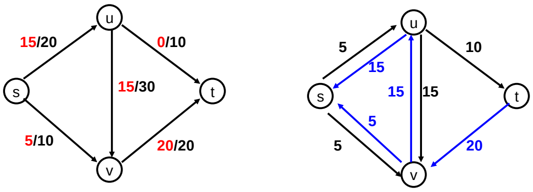
Network Flow Definitions

- Capacity
- Source, Sink
- Capacity Condition
- Conservation Condition
- Value of a flow

Flow Example



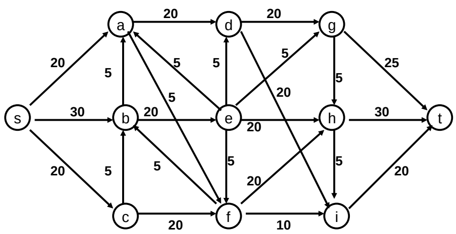
Flow assignment and the residual graph



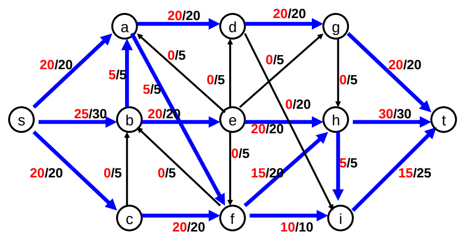
Network Flow Definitions

- Flowgraph: Directed graph with distinguished vertices s (source) and t (sink)
- Capacities on the edges, c(e) >= 0
- Problem, assign flows f(e) to the edges such that:
  - 0 <= f(e) <= c(e)
  - Flow is conserved at vertices other than s and t
    - Flow conservation: flow going into a vertex equals the flow going out
  - The flow leaving the source is a large as possible

Flow Example



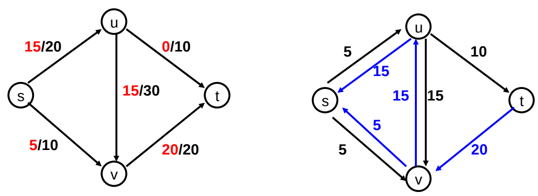
Find a maximum flow



Residual Graph

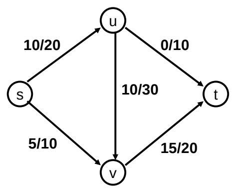
- Flow graph showing the remaining capacity
- Flow graph G, Residual Graph G_R
  - G: edge e from u to v with capacity c and flow f
  - G_R: edge e' from u to v with capacity c – f
  - G_R: edge e'' from v to u with capacity f

Flow assignment and the residual graph

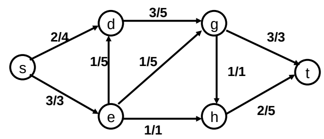


Augmenting Path Algorithm

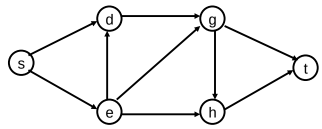
- Augmenting path
  - Vertices $v_1, v_2, \ldots, v_k$
    - $v_1 = s$, $v_k = t$
    - Possible to add b units of flow between v_j and v_{j+1} for j = 1 … k-1



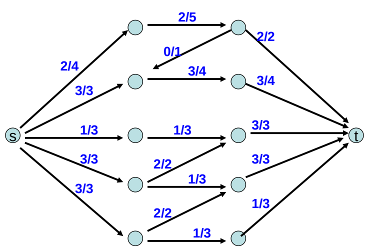
Build the residual graph



Residual graph:

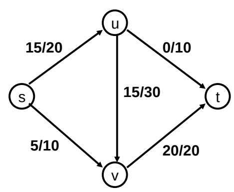


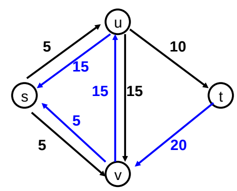
Find two augmenting paths



Augmenting Path Lemma

- Let $P = v_1, v_2, \ldots, v_k$ be a path from s to t with minimum capacity b in the residual graph.
- b units of flow can be added along the path P in the flow graph.





Proof

- Add b units of flow along the path P
- What do we need to verify to show we have a valid flow after we do this?
  -
  -

Ford-Fulkerson Algorithm (1956)

while not done

 Construct residual graph G_R

 Find an s-t path P in G_R with capacity b > 0

 Add b units along in G

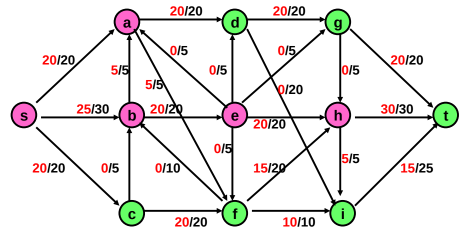
If the sum of the capacities of edges leaving S is at most C, then the algorithm takes at most C iterations

Cuts in a graph

- Cut: Partition of V into disjoint sets S, T with s in S and t in T.
- Cap(S,T): sum of the capacities of edges from S to T
- Flow(S,T): net flow out of S
  - Sum of flows out of S minus sum of flows into S
- Flow(S,T) <= Cap(S,T)

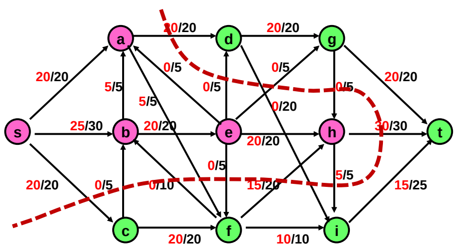
What is Cap(S,T) and Flow(S,T)

S={s, a, b, e, h}, T = {c, f, i, d, g, t}



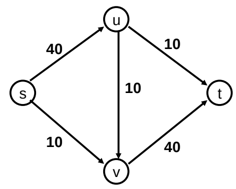
What is Cap(S,T) and Flow(S,T)

S={s, a, b, e, h}, T = {c, f, i, d, g, t}

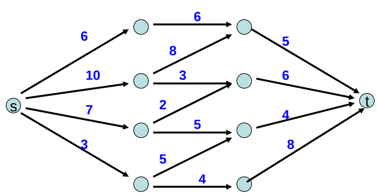


Cap(S,T) = 95, Flow(S,T) = 80 – 15 = 65

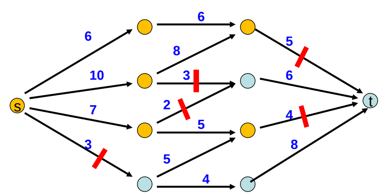
Minimum value cut



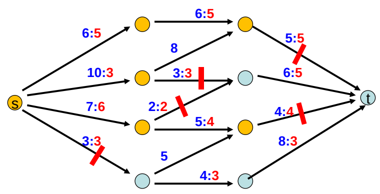
Find a minimum value cut



Find a minimum value cut

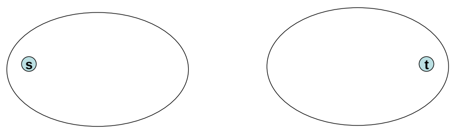


Find a minimum value cut

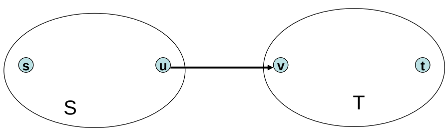


MaxFlow – MinCut Theorem

- There exists a flow which has the same value of the minimum cut
- Proof: Consider a flow where the residual graph has no s-t path with positive capacity
- Let S be the set of vertices in G_R reachable from s with paths of positive capacity



Let S be the set of vertices in G_R reachable from s with paths of positive capacity



What can we say about the flows and capacity between u and v?

Max Flow - Min Cut Theorem

- Ford-Fulkerson algorithm finds a flow where the residual graph is disconnected, hence FF finds a maximum flow.

- If we want to find a minimum cut, we begin by looking for a maximum flow.