Introduction to network flows
Overview

- > Transport network and flows
- > Complete flows.
- > Maximum flows.
- > Ford-Fulkerson algorithm.
- > Multiflows and dynamic flows.

Definition of a transport network and an *s*-*t*-flow

> *A transport Network* $R=(G,u,s,t)$ consists in :

- *A digraph* $G=(V,A)$.
- A capacity function $u : A \to \mathbb{R}^+$.
- Two particular vertices *s and t in V* (resp. called *source* and *sink*).

> Let $R=(G,u,s,t)$ be a transport network, we call *s*-*t*-flow on *R* function $f : A \to \mathbb{R}^+$ such that :

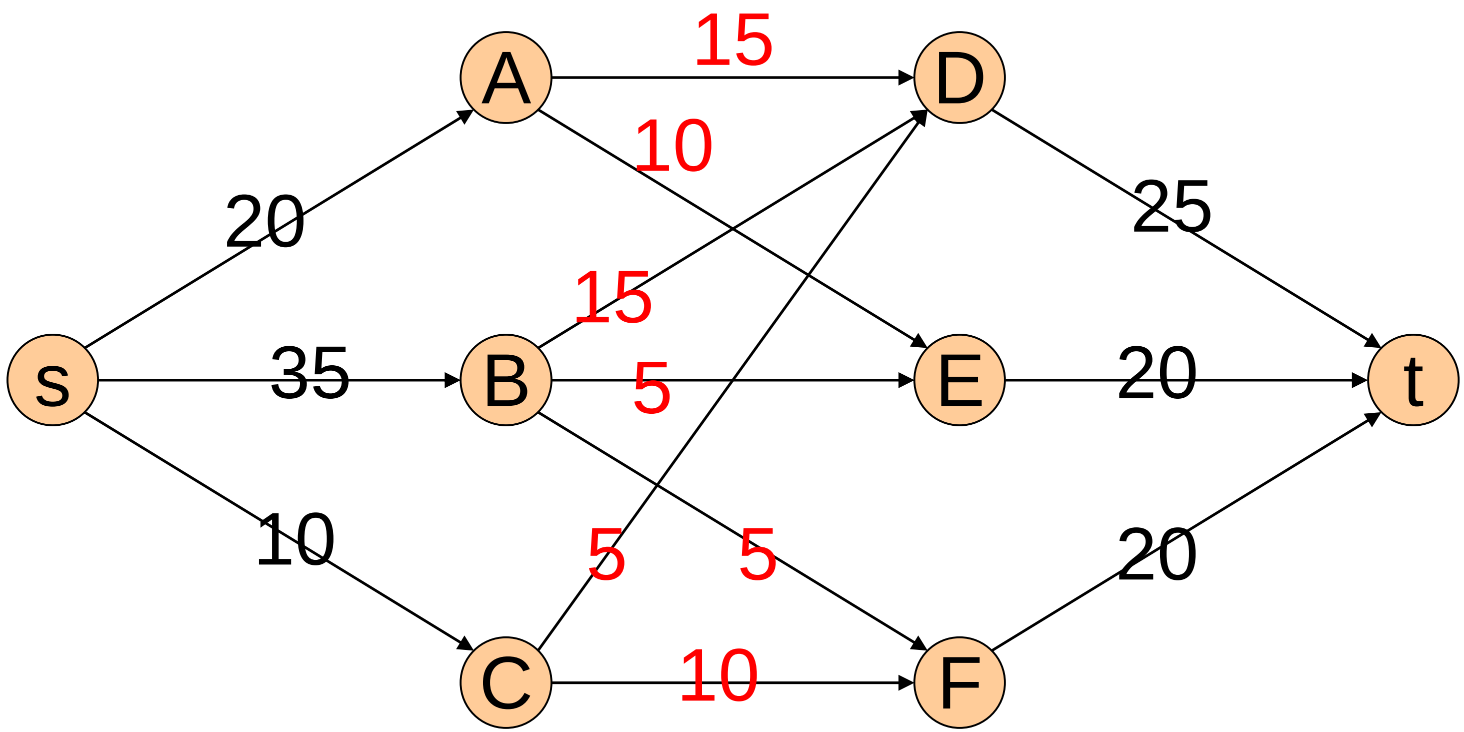
- $f(a) \leq u(a)$, for all $a \in A$.
- $\sum_{a \in \delta^-(v)} f(a) = \sum_{a \in \delta^+(v)} f(a)$, for all $v \in V \setminus \{s,t\}$ (Kirchoff constraint).

Example

> Three warehouses A, B, C, contain respectively 20, 35 et 10 tons of goods. There are demands of 25, 20 et 20 tons to destinations D, E and F. The unitary transportation costs are given in the following matrix. What would be the minimal cost transport plan ?

	D	E	F
A	15	10	0
B	15	5	5
C	5	0	10

The associated transport network



For arcs in the middle, unitary cost (in red) and capacity are taken as infinite
For other arcs, costs are taken as zeros.

Definition of the complete s-t-flow

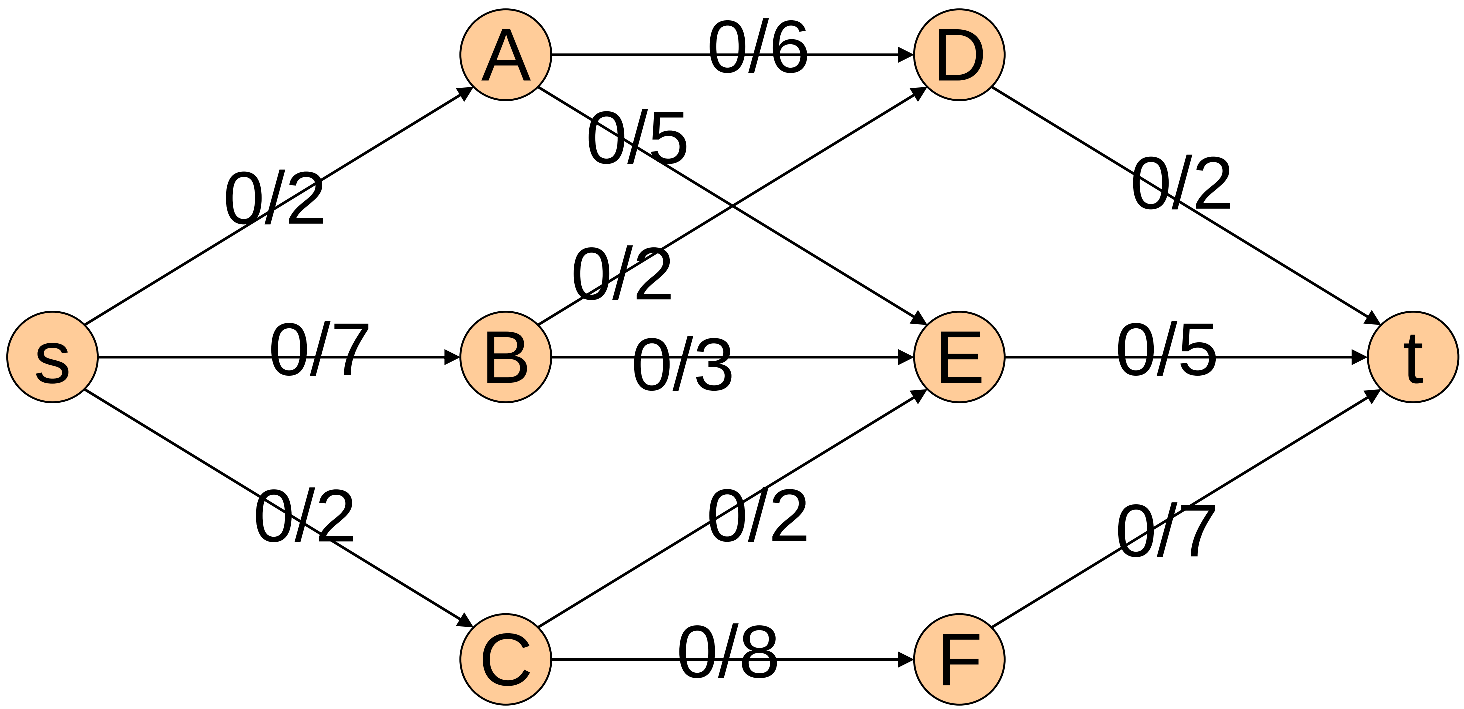
> Let $R=(G,u,s,t)$ be a transport network, s-t-flow f on R is called *complete* if for any s-t-path $P=((s,v_1),(v_1,v_2),\ldots,(v_{n-1},v_n),(v_n,t))$, there exists $a \in P$ such that $f(a) = u(a)$.

- Some complete flow is not necessarily maximum.

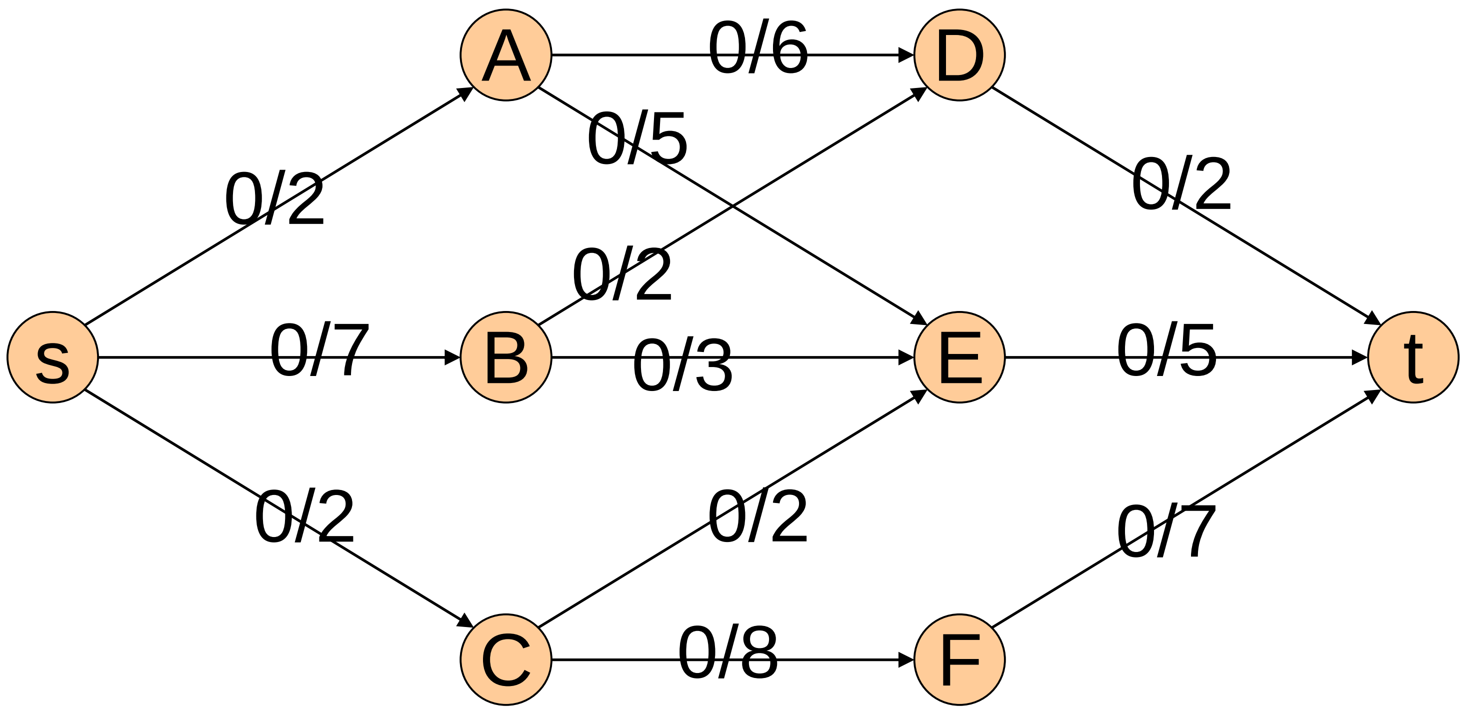
> Construction of a complete s-t-flow :

- Set $f := 0$.
- Repeat
  - Choose some s-t-path P such that $\forall a \in P$ we have $f(a)<u(a)$.
  - If such path doesn't exist, stop.
  - Otherwise, for all $a \in P$ do $f(a) := f(a) + \min_{a \in P} u(a)-f(a)$.

Example

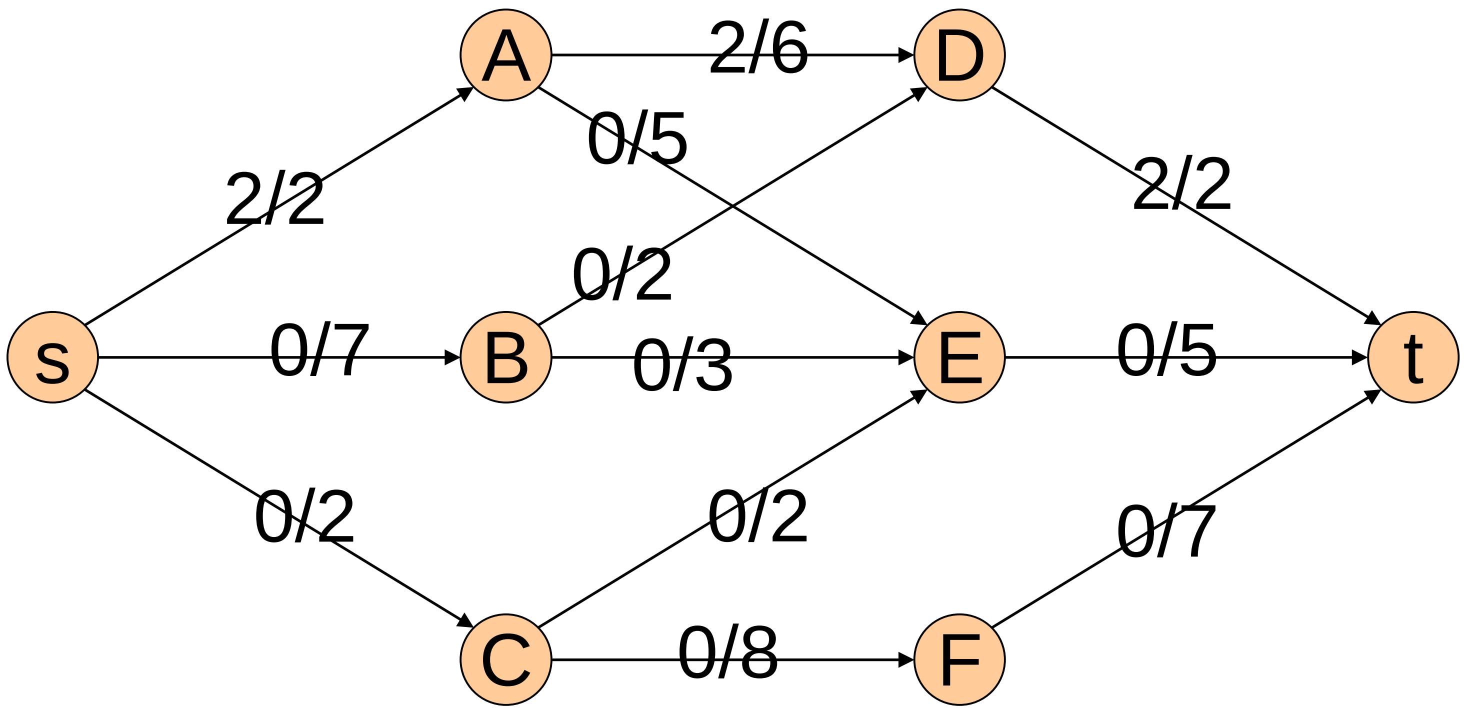


Example

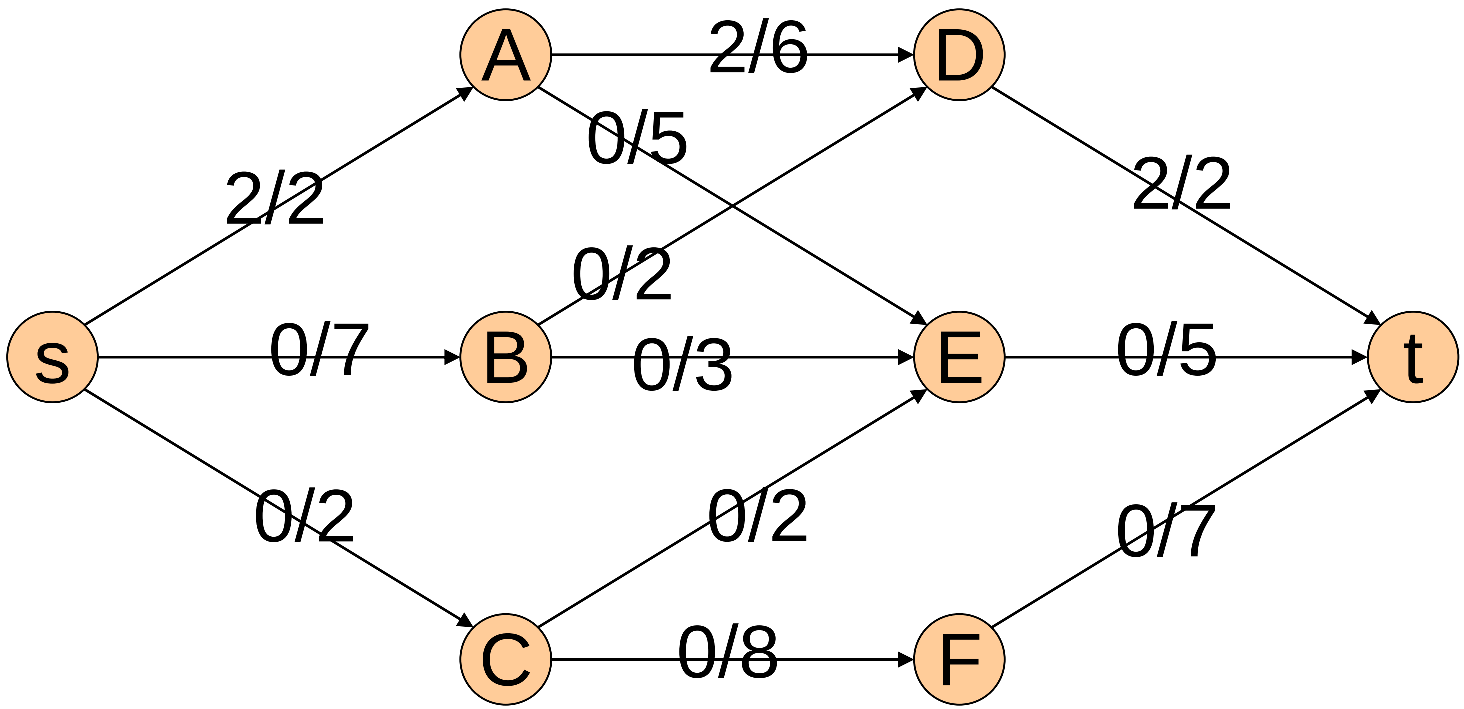


P = ((s,A),(A,D),(D,t)), $\min_{a \in P} u(a)-f(a)=2$

Example

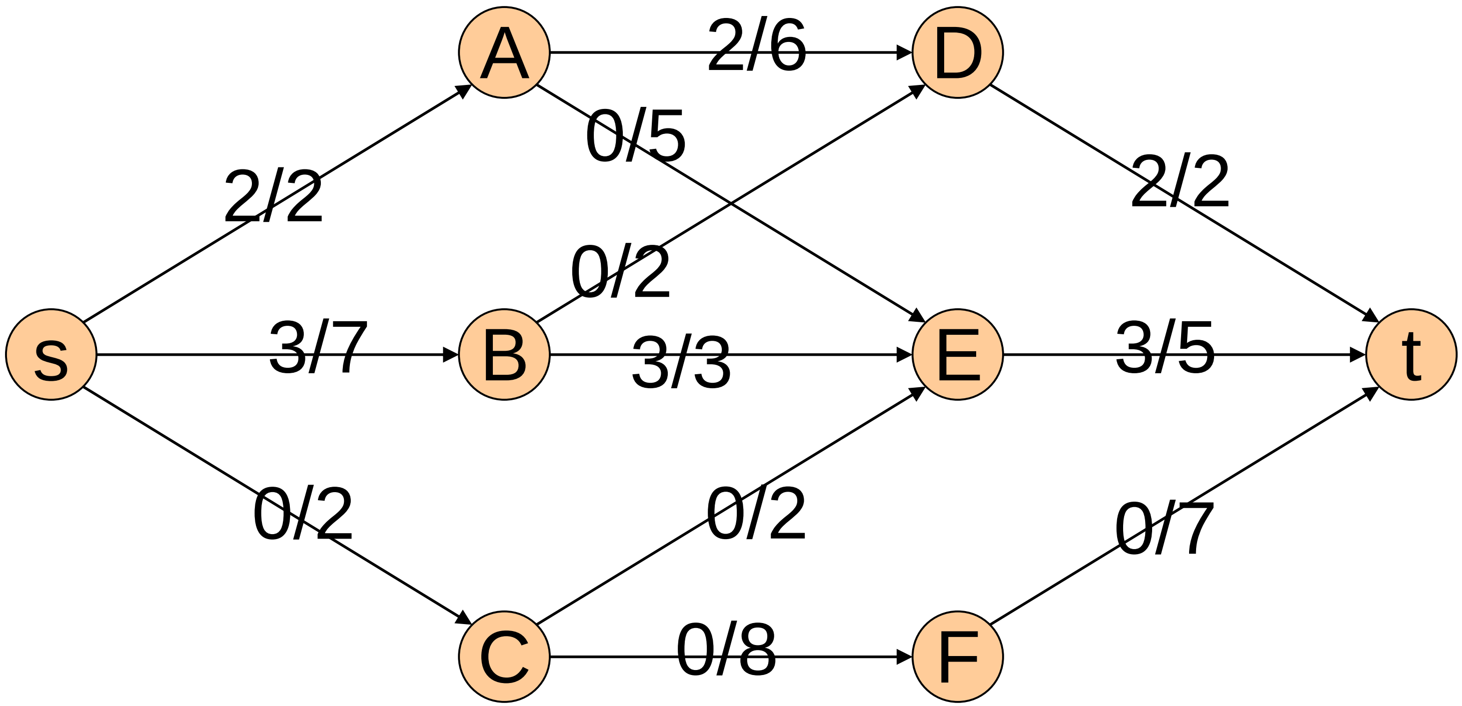


Example

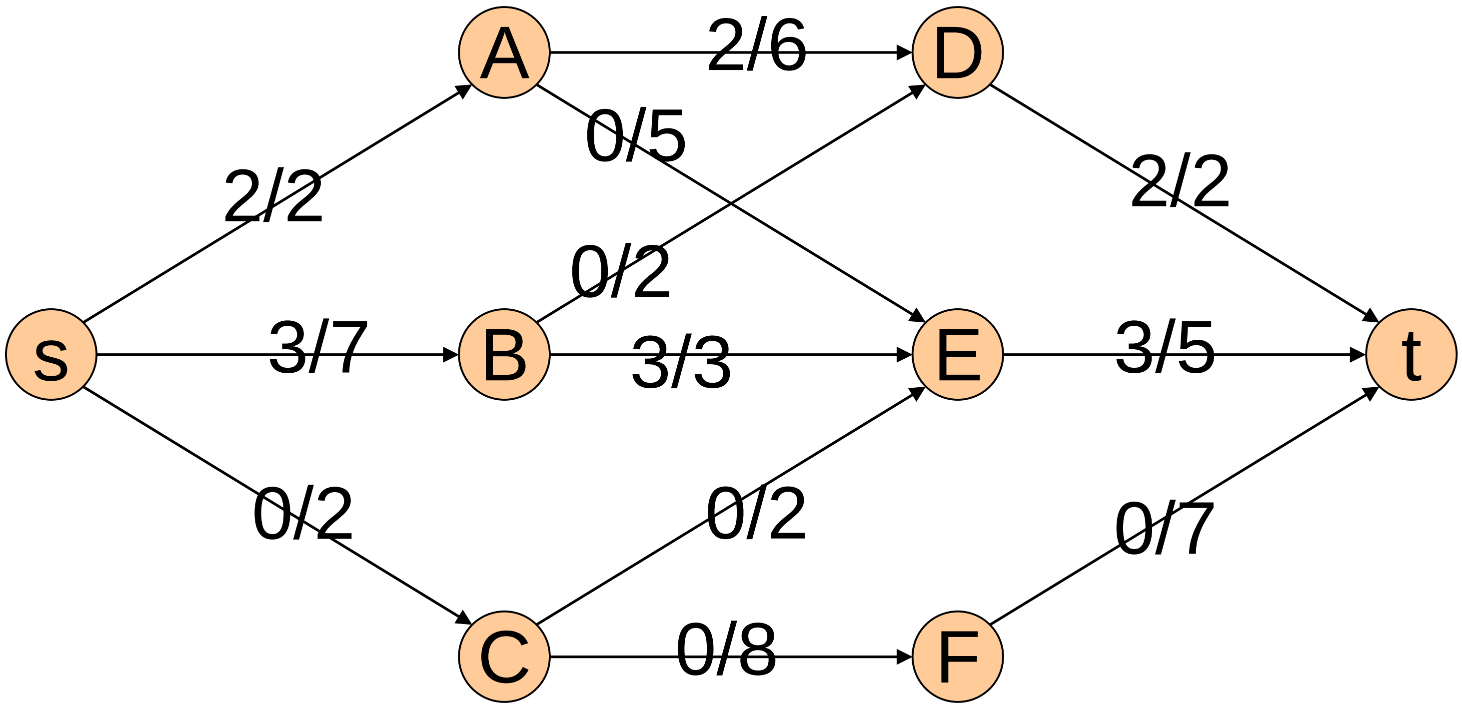


P = ((s,B),(B,E),(E,t)), $\min_{a \in P} u(a)-f(a)=3$

Example

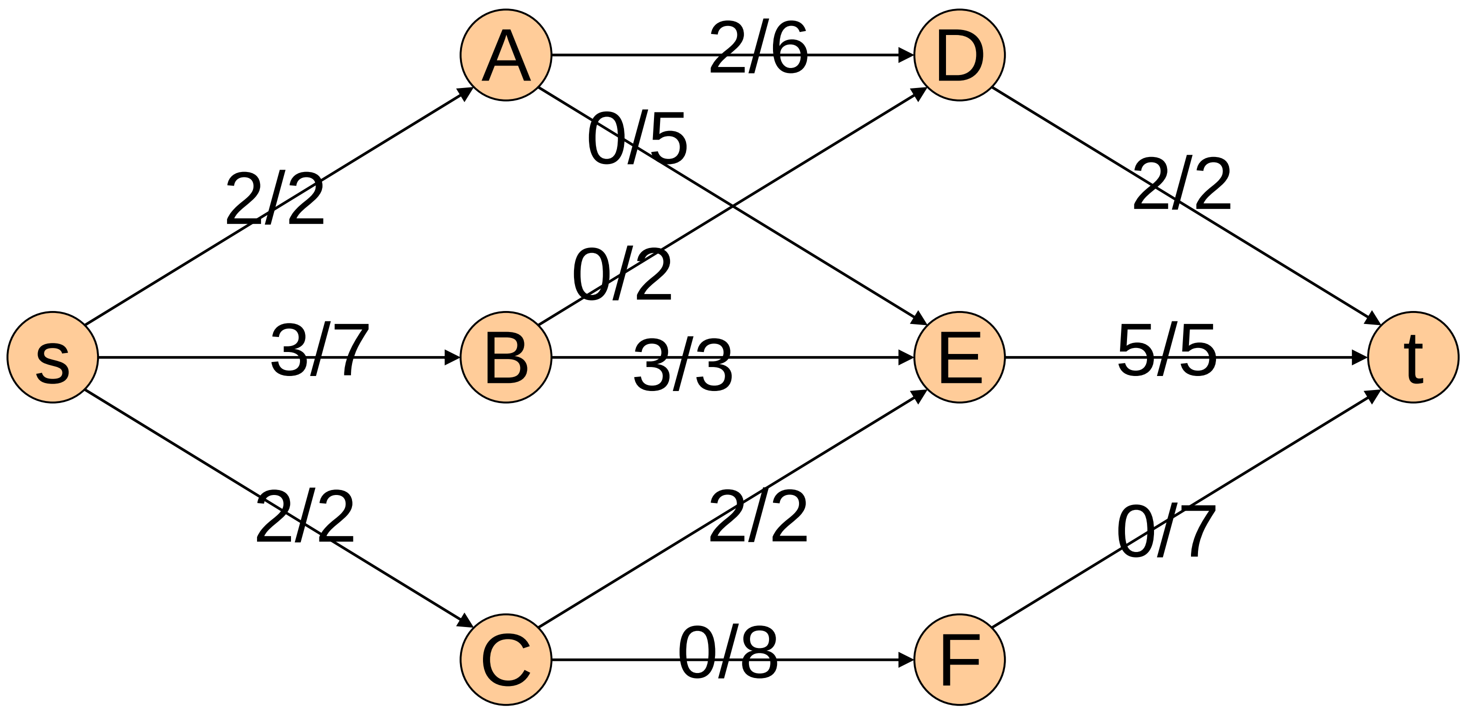


Example



$P = ((s,C),(C,E),(E,t)),\ \min_{a \in P} u(a)\text{-}f(a)=2$

Example



value(f) = 7

The maximum *s*-*t*-flow problem

> A complete flow is not necessarily a maximum one.

> Let $R=(G,u,s,t)$ be a transport network, the maximum *s*-*t*-flow problem consists in finding some *s*-*t*-flow f^* such that value(f^*) is maximum.

- I.e., find f^* s. t. for all *s*-*t*-flow f on R we have value(f) ≤ value(f^*).

The residual graph

> Given $G=(V,A)$ a digraph, we let $G'=(V,A\cup A')$ such that if $a=(v,w)\in A$ then $a'=(w,v)\in A'$.

- Remark : G' can contain parallel arcs even when G doesn't have any.

> Given a transport network $R=(G,u,s,t)$ and an s-t-flow f on R, $u_f : A\cup A'\rightarrow\mathbb{R}^+$ is called the residual capacity function s. t.:

- $u_f(a) = u(a) - f(a)$, for all $a\in A$.
- $u_f(a') = f(a)$, for all $a=(v,w)\in A$ and $a'=(w,v)\in A'$.

> We call *residual graph associated to f* the graph

$$G_f = (V,\{a\in A\cup A' : u_f(a) > 0\}).$$

Notion of *f*-augmenting path

> Let $R=(G,u,s,t)$ be a transport network, f an s-t-flow on R, P a path on G_f and $\gamma > 0$, *augment f* with γ over P means:
- $f(a) := f(a) + \gamma$ when $a \in A$.
- $f(a) := f(a) - \gamma$ when $a' \in A'$.

> We call *f-augmenting path*, any s-t-path in G_f.

Ford-Fulkerson algorithm

> Input: R=(G,u,s,t) a transport network.

> Output: an s-t-flow f of maximum value.

> Description:

- Set f := 0.
- Repeat:
  - Choose an f-augmenting path P.
  - If such path doesn’ t exist, then end (f is maximum).
  - Augment f with $\gamma = \min_{a \in P} u_f(a)$ over P.

Ford-Fulkerson algorithm

> A practical method:
  - Step 1. Set f :=0; mark S;
  - Step 2. While there is a non examinated marked node i and t is not marked, do:

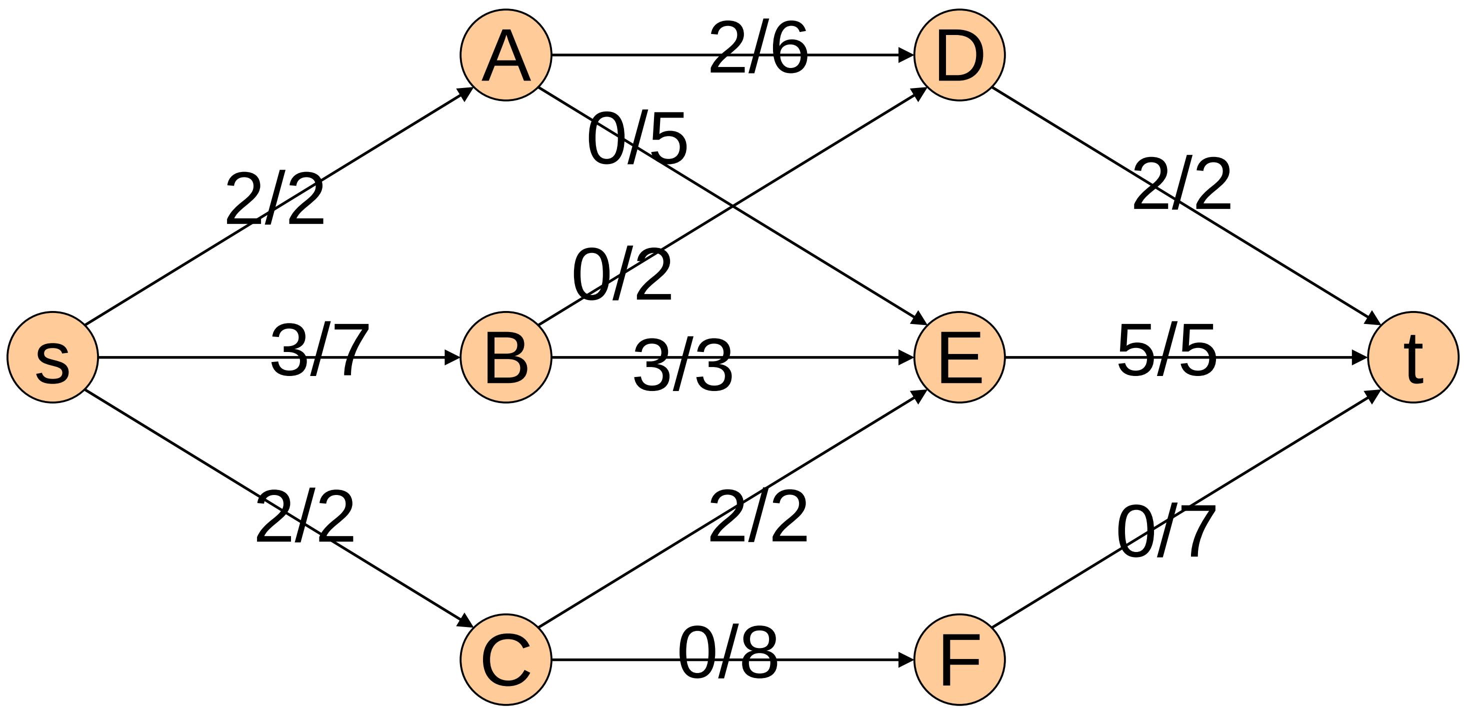
    Mark all successors with residual capacities of i with i+;

    Mark all predecessors with positive incoming flow of i with i−;

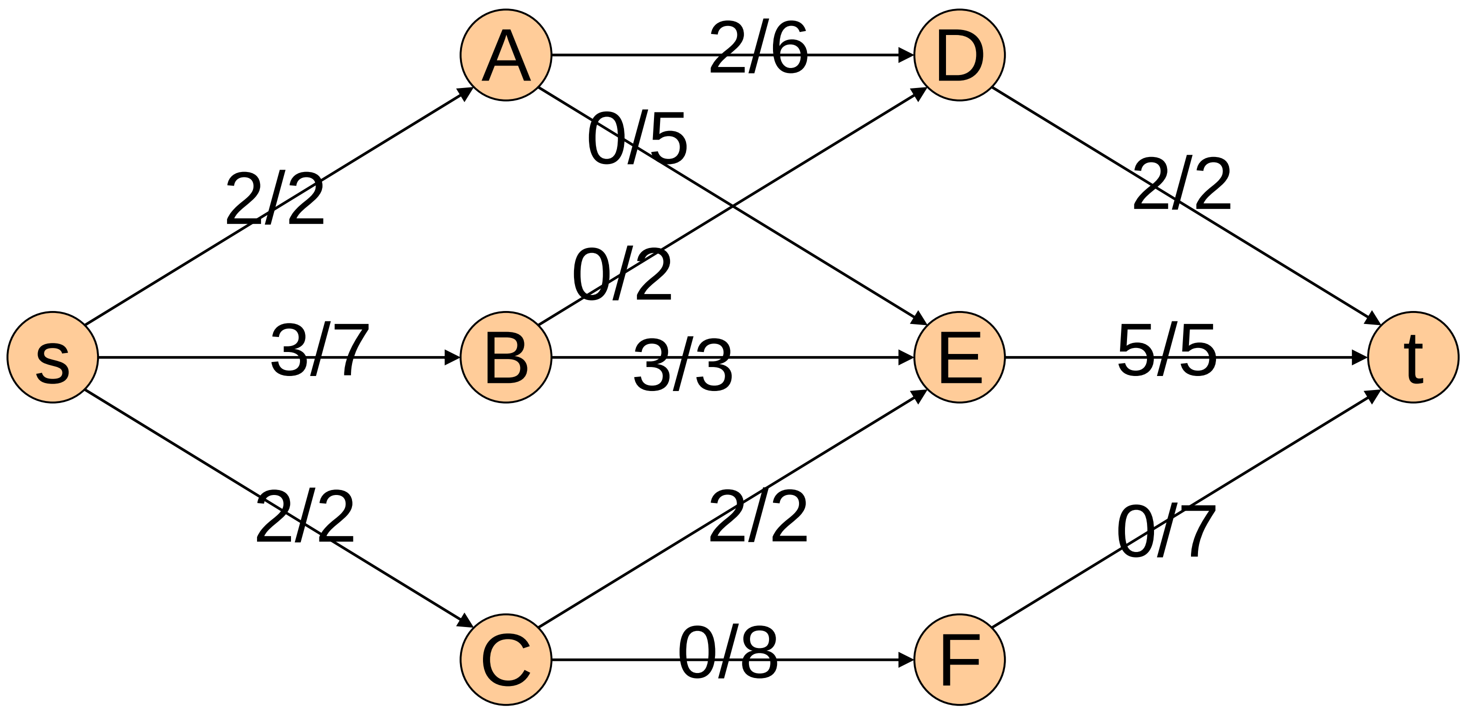
    End while;
  - Step 3. If t is marked, determine the augmenting path P and augment f with $\gamma = \min_{a \in P} u_f(a)$ over P. Mark S and go back to step 2.

    Otherwise, the flow is maximal;

Example (previous)

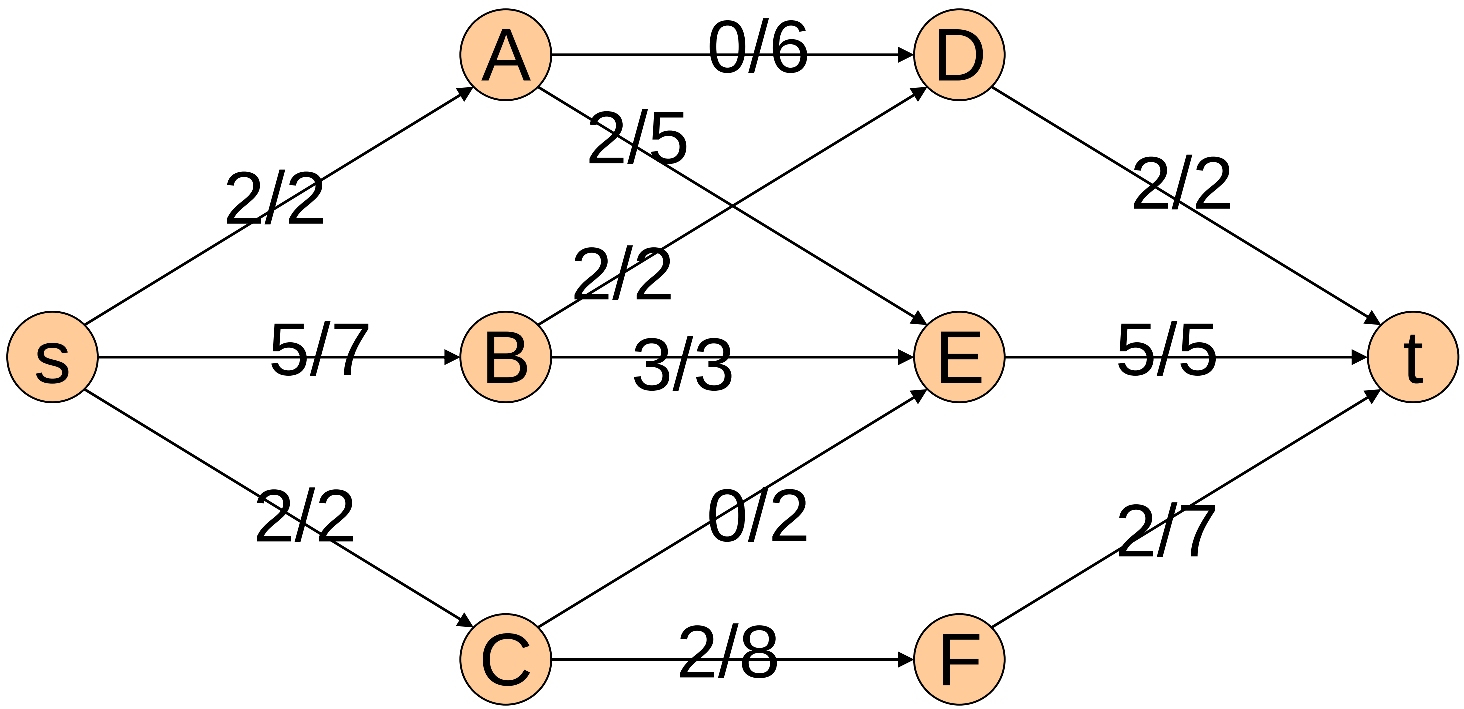


Example (previous)



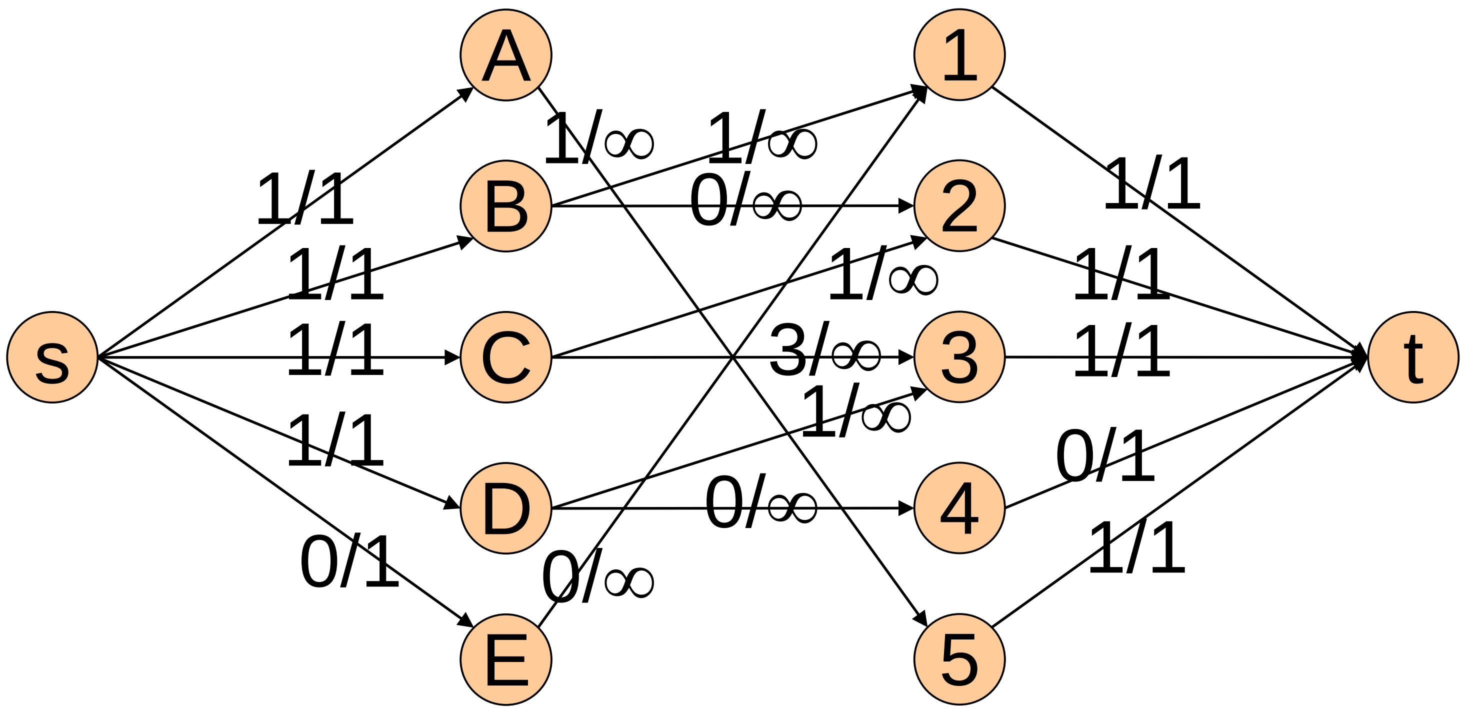
P = ((s,B),(B,D),(D,A),(A,E),(E,C),(C,F),(F,t)),
$\min_{a \in P} u_f(a) = 2$

Example (previous)



value(f) = 9, f is maximum

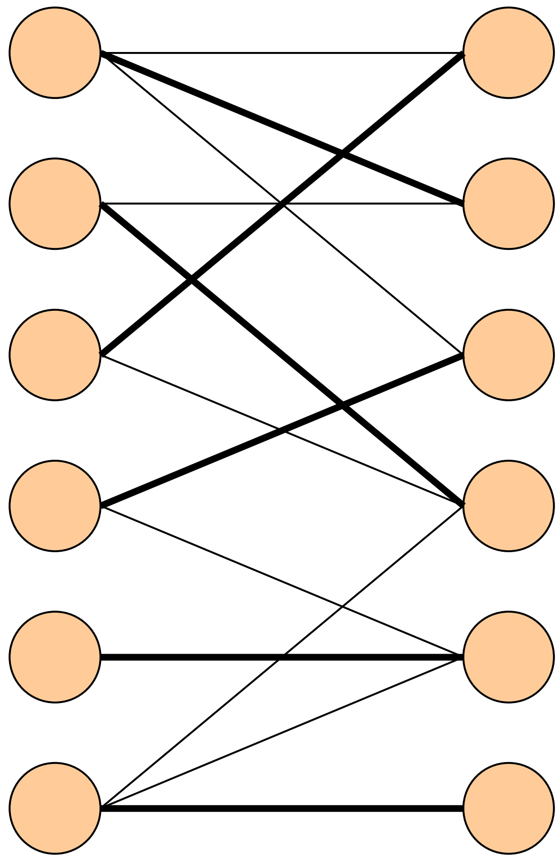
Go on …



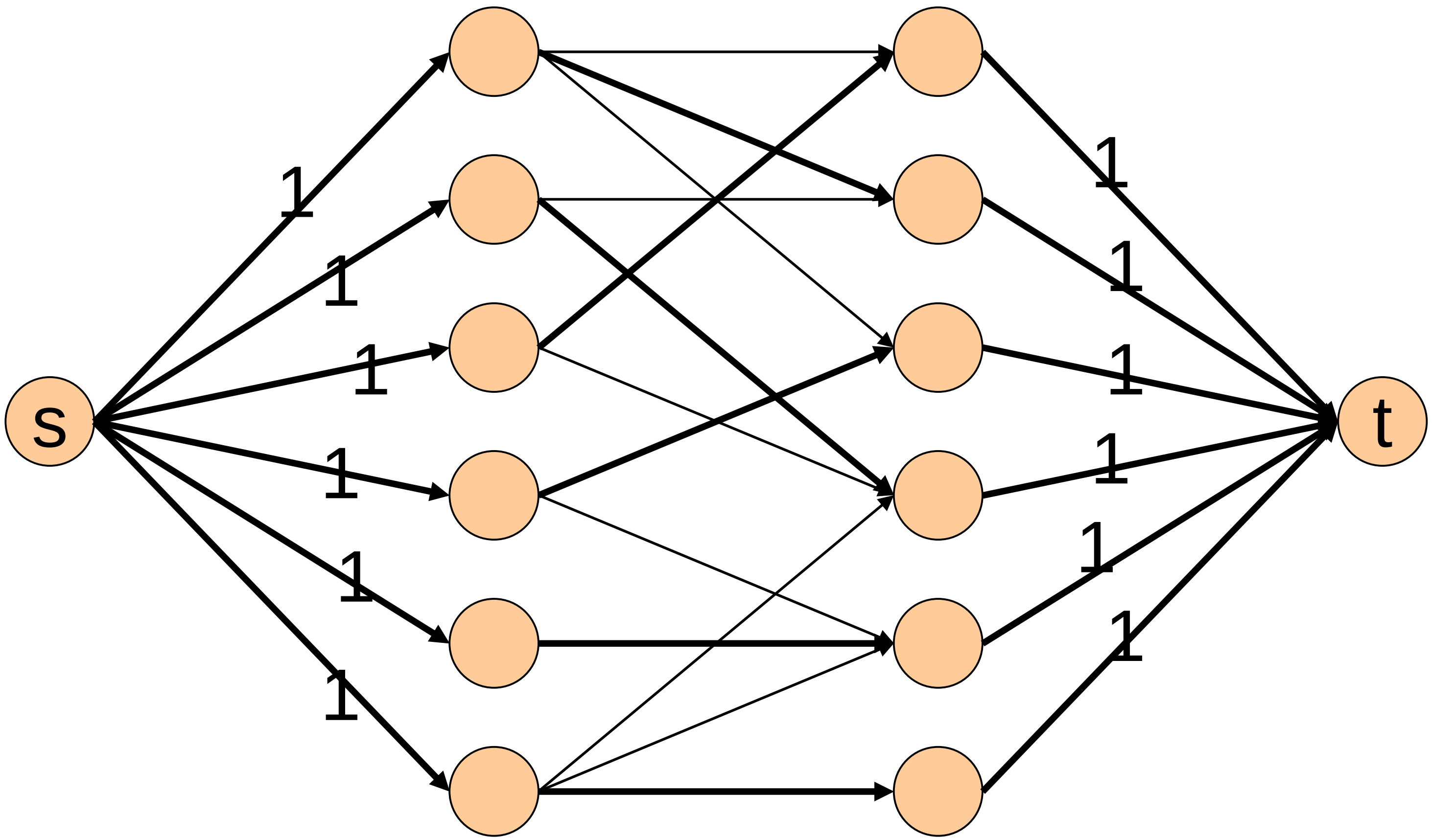
Transform the above complet flow to a maximum one

The maximum bipartite matching

> Transform this problem to a maximum flow one.



The maximum bipartite matching



Lemma on generalized Kirchoff constraints

> Let $R=(G,u,s,t)$ be a transport network and f an s-t-flow in R then for all $X \subseteq V \setminus \{s,t\}$ we have :

$$\sum_{a \in \delta^{-}(X)} f(a) = \sum_{a \in \delta^{+}(X)} f(a).$$

$\square$

Lemma

- Let $R=(G,u,s,t)$ be a transport network and f an s-t-flow in R then for all $X \subseteq V$ s. t. $s \in X$ and $t \notin X$ we have :
  - (a) value(f) = $\sum_{a \in \delta^+(X)} f(a) - \sum_{a \in \delta^-(X)} f(a)$.
  - (b) value(f) ≤ $\sum_{a \in \delta^+(X)} u(a)$.
- Proof :
  - (a) : similar to previous lemma.
  - (b) : trivial implication of (a) and of positivity of flow function ($0 \leq f(a) \leq u(a)$).
  - □

Theorem

> Let $R=(G,u,s,t)$ be a transport network, an s-t-flow f in R is maximum iff it doesn't exist an f-augmenting path.

> Proof :

Necessity : if such path exist, it would conduct to larger flow value, hence f is not maximum.

Sufficiency : if there is no f-augmenting path, t is not reachable (by definition) from s in G_f. Let Y be the set of reachable vertices from s in G_f. By definition of G_f we have $f(a)=u(a)$ (because $u_f(a)=u(a)-f(a)=0$) for all a of $\delta^+_G(Y)$ and $f(a)=0$ (because $u_f(a')=f(a)=0$) for all a in $\delta^-_G(Y)$. The previous lemma (part (a)) implies

$$\text{value}(f) = \sum_{a \in \delta^+_G(Y)} u(a),$$

and part(b) allows to conclude.

□

Connection to the minimum cut problem

> Let $G=(V,A)$ be a digraph, $X \subseteq V$ is called cut in G if $s \in X$ and $t \notin X$.

- We define capacity(X) = $\sum_{a \in \delta^+(X)} u(a)$.

> Theorem : Let $R=(G,u,s,t)$ be a transport network, the maximum value of s-t-flow in R is equal to the minimum capacity of cuts in G.

> Proof (hint) :

For any cut X, we have capacity($\delta^+(X)$) = $\sum_{a \in \delta^+(X)} u(a) \geq$ value(f).

On the other hand, for a maximum flow we can build a cut with the same capacity, which proves the validity of the theorem.

□

Remarks on the integrality

- Corollary : let $R=(G,u,s,t)$ be a transport network with $u : A \rightarrow \mathbb{Z}^+$, then there exists a maximum *s*-*t*-flow with integral values and Ford-Fulkerson algorithm achieves such a flow.
- Proof :

  This is because the capacity function has integer values $\gamma=\min_{a\in P} u_f(a)$.

  □
- Remark : the corresponding LP yields as well integral values because the associated matrix is TU (i.e., all their sub-determinants are 0, +1 or −1).

Remarks on algorithmic aspects

> To achieve better performances, it is convenient not to choose *f*-augmenting paths arbitrarily.

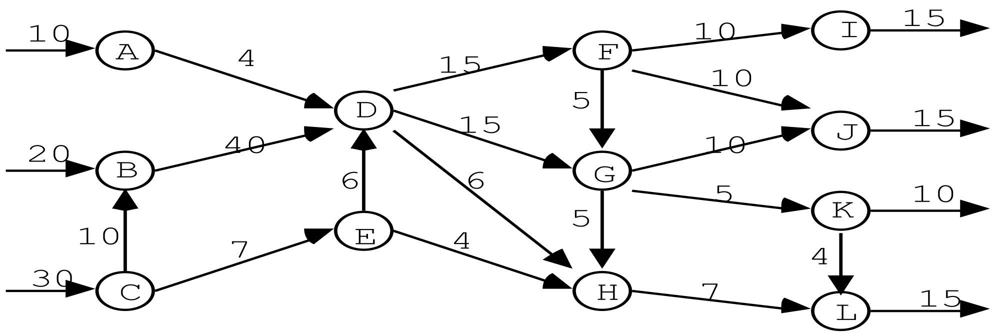
> Par ex. the Edmonds-Karp algorithm suggests to choose the shortest augmenting path.

- Theorem : The Edmonds-Karp algorithm solve the s-t-flow maximum problem in $O(m^2 n)$.

> Other algorithms exist, for instance the Golberg-Tarjan algorithm, which runs in $O(n^3)$.

A PROBLEM OF URBAN CIRCULATION

The road network which Fulkerson City must maintain is illustrated below (the capacities of the arcs represent the maximum flow in cars per hour).



1) Let N be the maximum number of cars per hour which can circulate through this network.

1.1) Calculate N.

1.2) Characterize the routes that can be created in order to increase N. By how much can we increase N?

2) The local officials want to improve the circulation, increasing the number N by creating new routes. In urban areas, only the creation of a route HK and the widening of the street EG previously reserved for pedestrians are feasible. The choice will therefore have to be made between:

a) creating HK; b) widening EG; c) creating HK and widening EG.

END

Multicommodity flows

Un multi-commodity flow Φ is composed of several distinct flows Φ^d for any couples *(s^d,t^d).* Any node in the graph can be source of flow destination.

Similarly to simple flows, a multi-flow defines a function Φ from *U* to *R+ satisfying:*

- *Capacity constraints(!)*
- *Flow conservation (!!)*
- *Flow value to satisfy/maximize(!!!)*

Dynamic flow

> Until now we have seen static flows, no temporal considerations have been taken into account;

> Some flows are dynamic in time, ATM, transport…

> Ford-Fulkerson (1958) have shown how we can transform dynamic flows to static ones.

Transformation of a dynamic flow to a static one an example