Network Design and Optimization course

Lecture 4

Alberto Ceselli
alberto.ceselli@unimi.it

Dipartimento di Tecnologie dell'Informazione
Università degli Studi di Milano

October 24, 2011

The problem

Given

- a set of nodes,
- a set of links connecting them

(that is, an existing network).
I want to analyze the overall performance of my network in terms of bandwidth:

- assess the bandwidth available while sending data from a particular node to another,

The problem

Given

- a set of nodes,
- a set of links connecting them

(that is, an existing network).
I want to analyze the overall performance of my network in terms of bandwidth:

- assess the bandwidth available while sending data from a particular node to another,
- how to fully exploit this bandwidth while sending packets,

The problem

Given

- a set of nodes,
- a set of links connecting them

(that is, an existing network).
I want to analyze the overall performance of my network in terms of bandwidth:

- assess the bandwidth available while sending data from a particular node to another,
- how to fully exploit this bandwidth while sending packets,
- which links form the bottleneck,

The problem

Given

- a set of nodes,
- a set of links connecting them

(that is, an existing network).
I want to analyze the overall performance of my network in terms of bandwidth:

- assess the bandwidth available while sending data from a particular node to another,
- how to fully exploit this bandwidth while sending packets,
- which links form the bottleneck,
- ...

Assumptions

Some assumptions:

1. no *costs* involved: packets can also follow non-shortest paths,
2. in general, packets will be routed on many (different) links,
3. *capacity* plays a central role: we want to push as many packets as possible,
4. no packet is loss in the transmission: every packet entering a node (resp. link) is assumed to leave the node (resp. link) as well.

Recognizing a known problem ...

We are entering the realm of *Maximum Flow problems*!

Modeling the capacity

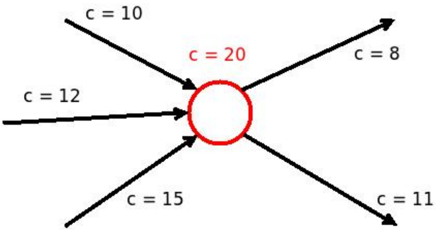
Step 1: estimating capacities.

- link capacities: several times explicit in network optimization (i.e. link bandwidth),
- node capacities:
  - several times, nodes are orders of magnitude faster than links;
  - otherwise: we estimate node behaviour and ...

Modeling the capacity

Step 1: estimating capacities.

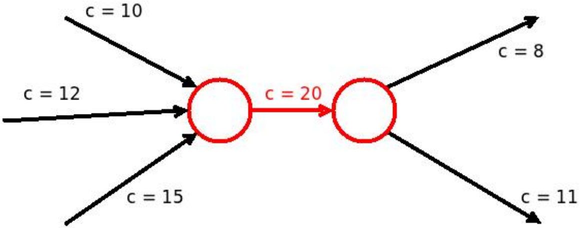
- link capacities: several times explicit in network optimization (i.e. link bandwidth),
- node capacities:
  - several times, nodes are orders of magnitude faster than links;
  - otherwise: we estimate node behaviour and ...



Modeling the capacity

Step 1: estimating capacities.

- link capacities: several times explicit in network optimization (i.e. link bandwidth),
- node capacities:
  - several times, nodes are orders of magnitude faster than links;
  - otherwise: we estimate node behaviour and ...



Graph model

Given a network, build a *directed* graph $G = (V, A)$ having

- one vertex $i \in V$ for each node of the network
- one arc $a \in A \subseteq V \times V$ for each link of the network
- capacities $u_{(i,j)}$ on each arc $(i, j) \in A$
- a special vertex $s \in V$ representing the origin of packets
- a special vertex $t \in V$ representing the destination of packets

Mathematical Programming model

Let $x_{(i,j)}$ be *decision variables* representing the *amount of flow* sent on arc (i, j). Let v represent the total amount of flow when $x_{(i,j)}$ units are sent on arcs.

$$
\begin{aligned}
\text{maximize } & v \\
\text{subject to } & \sum_{j \in V} x_{(i,j)} = \sum_{k \in V} x_{(k,i)} && \forall i \in V, i \neq s, t \\
& \sum_{j \in V} x_{(s,j)} = v \\
& 0 \leq x_{(i,j)} \leq u_{(i,j)} && \forall (i,j) \in A
\end{aligned}
$$

Question 1

Question 1: assuming no other packet is routed on the network, what is the maximum transmission rate between s and t?

Question 1

Question 1: assuming no other packet is routed on the network, what is the maximum transmission rate between s and t?
$\rightarrow$ find a *Maximum s-t flow* on G (i.e. give a value to variables $x_{(i,j)}$ maximizing v).

MaxFlow algorithms

A first basic approach: augmenting paths

- Ford Fulkerson algorithm,
- correctness and termination,
- cut duality theory,
- some thoughts on complexity.

(See Orlin's slides).

MaxFlow algorithm implementation

Lab session: implementing

- Ford Fulkerson

in AMPL.