Advanced Algorithms

Flow Network

Applications

- fluid in pipes
- current in an electrical circuit
- traffic on roads
- data flow in a computer network
- money flow in an economy

Applications

Image Segmentation

Image segmentation.

- Central problem in image processing.
- Divide image into coherent regions.
  - three people standing in front of complex background scene
  - identify each of three people as coherent objects

Applications

Airline Scheduling

Airline scheduling.

- Complex computational problem faced by nation's airline carriers.
- Produces schedules for thousands of routes each day that are efficient in terms of:
  - equipment usage, crew allocation, customer satisfaction
  - in presence of unpredictable issues like weather, breakdowns
- One of largest consumers of high-powered algorithmic techniques.

"Toy problem."

- Manage flight crews by reusing them over multiple flights.
- Input: list of cities V.
- Travel time t(v, w) from city v to w.
- Flight i: (o_i, d_i, t_i) consists of origin and destination cities, and departure time.

Applications

Matrix Rounding

Feasible matrix rounding.

- Given a p x q matrix D = $\{d_{i\,j}\}$ of real numbers.
- Row i sum = a_i, column j sum b_j.
- Round each d_{ij}, a_i, b_j up or down to integer so that sum of rounded elements in each row (column) equal row (column) sum.
- Original application: publishing US Census data.

Theorem: for any matrix, there exists a feasible rounding.

3.14	6.8	7.3	17.24
9.6	2.4	0.7	12.7
3.6	1.2	6.5	11.3
16.34	10.4	14.5	

Original Data

3	7	7	17
10	2	1	13
3	1	7	11
16	10	15	

Possible Rounding

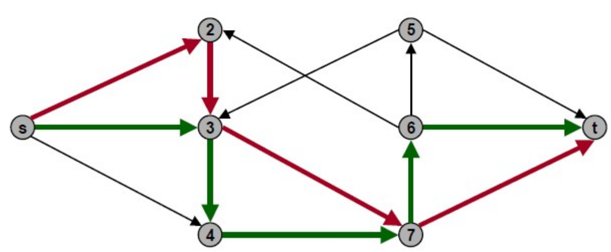
Applications

Disjoint Paths

Disjoint path network: G = (V, E, s, t).

- Directed graph (V, E), source s, sink t.
- Two paths are edge-disjoint if they have no arc in common.

Disjoint path problem: find max number of edge-disjoint s-t paths.



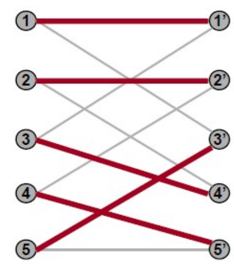
Applications

Dancing Problem (k-Regular Bipartite Graph)

Dancing problem.

- Exclusive Ivy league party attended by n men and n women.
- Each man knows exactly k women.
- Each woman knows exactly k men.
- Acquaintances are mutual.
- Is it possible to arrange a dance so that each man dances with a different woman that he knows?

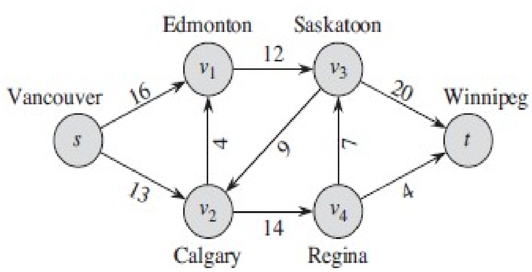
Mathematical reformulation: does every k-regular bipartite graph have a perfect matching?



What is Network Flow ?

- Each edge (u,v) has a non-negative capacity c(u,v).
- If (u,v) is not in E assume c(u,v)=0.
- We have source s and sink t.
- Assume that every vertex v in V is on some path from s to t.

Following is an illustration of a network flow:



c(s,v1)=16
c(v1,s)=0
c(v2,s)=0 …

Flow:

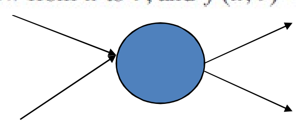
- G=(V,E): a flow network with capacity function c.
- s-- the source and t-- the sink.
- A flow in G: a real-valued function f:VxV → R satisfying the following two properties:

Capacity constraint: For all $u, v \in V$, we require $0 \leq f(u, v) \leq c(u, v)$.

Flow conservation: For all $u \in V - \{s, t\}$, we require

$$\sum_{v \in V} f(v, u) = \sum_{v \in V} f(u, v) .$$

When $(u, v) \notin E$, there can be no flow from u to v, and $f(u, v) = 0$.



9

Example of a flow

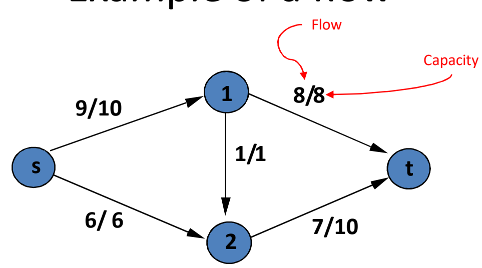


Table illustrating Flows and Capacity across different edges of graph above:

$f_{s,1}$ = 9 , $c_{s,1}$ = 10 (Valid flow since 10 > 9)

$f_{s,2}$ = 6 , $c_{s,2}$ = 6 (Valid flow since 6 ≥ 6)

$f_{1,2}$ = 1 , $c_{1,2}$ = 1 (Valid flow since 1 ≥ 1)

$f_{1,t}$ = 8 , $c_{1,t}$ = 8 (Valid flow since 8 ≥ 8)

$f_{2,t}$ = 7 , $c_{2,t}$ = 10 (Valid flow since 10 > 7)

The flow across nodes 1 and 2 are also conserved as flow into them = flow out.

Net flow and value of a flow f:

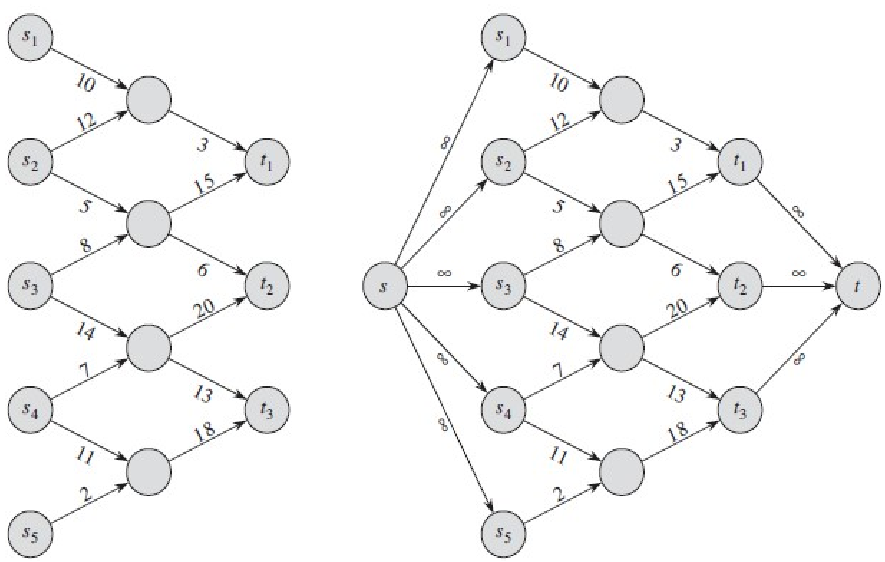
- The quantity f (u, v) is called the net flow from vertex u to vertex v.
- The value of a flow is defined as

$$| f | = \sum_{v \in V} f(s, v) = \sum_{v \in V} f(v, t)$$

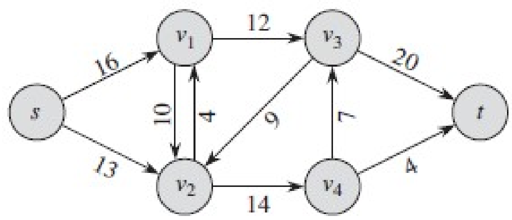
  - The total flow from source to any other vertices.
  - The same as the total flow from any vertices to **the sink.**

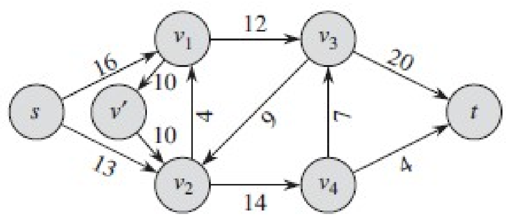
11

Multiple Source & Multiple Sink Nodes



How to handle anti-parallel edges ?





Maximum-flow problem:

- Given a flow network G with source s and sink t
- Find a flow of maximum value from s to t.

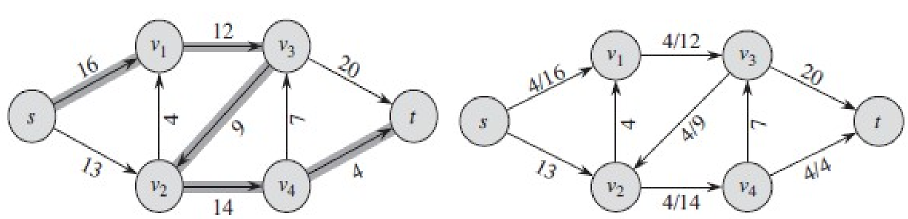
- How to solve it efficiently?

14

Residual networks:

- Given a flow network and a flow, the **residual network** consists of edges that can admit more net flow.
- G=(V,E) --a flow network with source s and sink t
- f: a flow in G.

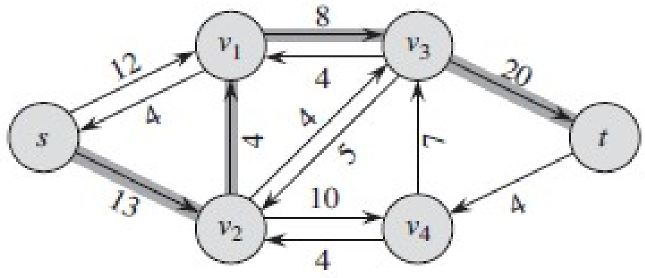
Residual Networks



The amount of additional net flow from u to v before exceeding the capacity c(u,v) is the residual capacity of (u,v), given by:
$c_f(u,v)=c(u,v)-f(u,v)$

in the other direction: $c_f(v, u)=c(v, u)+f(u, v).$

Example of Residual network (continued)



17