3E4: Modelling Choice

Lecture 5

Network Flows Problems

1

Announcements

Coursework

- To be handed in:
  - on Mon 28 Feb, 4pm, Lecture Room 1, Eng. Dept.
- Filling in the coversheet:
  - MODULE: **3E4 MODELLING CHOICE**
  - MODULE LEADER: **GG278**
  - TITLE OF COURSEWORK: **Production planning and optimal decisions: The SteelPro case**
- Cover sheet also available online

Solutions to Lecture 1-3 Homework

- now available from http://www.eng.cam.ac.uk/~dr241/3E4

2

3E4 : Lecture Outline

Lecture 1. Management Science & Optimisation Modelling: Linear Programming

Lecture 2. LP: Spreadsheets and the Simplex Method

Lecture 3. LP: Sensitivity & shadow prices
Reduced cost & shadow price formulae

Lecture 4. Integer LP: branch & bound

Lecture 5. Network flows problems

Lecture 6. Multiobjective LP

Lecture 7 – 8. Introduction to nonlinear programming

3

Introduction

- A number of business problems can be represented graphically as networks.
- Several types of network flow problems:
  - Transshipment Problems
  - Shortest Path Problems
  - Transportation/Assignment Problems
  - Minimal Spanning Tree Problems
  - Maximal Flow Problems
  - Generalized Network Flow Problems

4

Characteristics of Network Flow Problems

- Network flow problems can be represented as “graphs”, i.e. a collection of nodes connected by arcs.
- There are three types of nodes:
  - “Supply” or “Source” (less flow goes in than comes out)
  - “Demand” or “Sink” (more flow goes in than comes out)
  - “Transshipment” (inflow = outflow)
- We’ll use
  - **Net inflow** (=inflow – outflow) to model the amount of flow passing through a node, hence we’ll use negative numbers to represent supplies and positive numbers to represent demand.

5

Characteristics of Network Flow Problems

- Arcs can be directed or undirected
- Arcs can have a limited capacity
- A “cost” (or a “cost” function) can be associated to each arc
- A directed graph whose nodes and/or arcs have associated numerical values (costs, capacities, supplies, demands) is named “directed **network**”.

6

Important *properties of Network LPs*

- The relatively simple structure of Network LPs means there is very **efficient** (fast) **software** specialised for these problems.
- If a Network LP with <u>**INTEGER**</u> data has any solutions then
  - it has at least one solution with all integer flows
  - the Simplex method will find an integral solution!

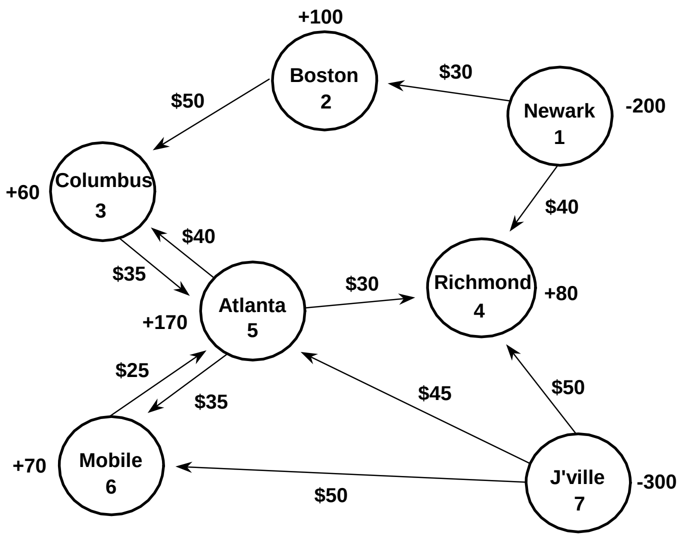
7

A Transshipment Problem: The Bavarian Motor Company

A luxury car importer, BMC, ships vehicles from Hamburg, Germany to Jacksonville (FLA) and Newark (NJ) in the USA, and then must distribute them as cheaply as possible to 5 other cities.

8

A Transshipment Problem: The Bavarian Motor Company



9

Defining the Decision Variables

For each arc in a network flow model
we define a decision variable as:

X_{ij} = the amount being shipped (or flowing) <u>from</u> node *i* <u>to</u> node *j*

For example,

X_{12} = the number of cars shipped *from* node 1 (Newark) *to* node 2 (Boston)

X_{56} = the number of cars shipped *from* node 5 (Atlanta) *to* node 6 (Mobile)

Note: The number of arcs determine the number of variables in a network flow problem!

10

Defining the Objective Function

Minimize total shipping costs.

$$\text{MIN: } 30X_{12} + 40X_{14} + 50X_{23} + 35X_{35} + 40X_{53} + 30X_{54} + 35X_{56} + 25X_{65} + 50X_{74} + 45X_{75} + 50X_{76}$$

11

Constraints for Network Flow Problems:
The Balance-of-Flow Rules

For Minimum Cost Network Flow Problems Where:	Apply This Balance-of-Flow Rule At Each Node:
Total Supply > Total Demand	Inflow-Outflow >= Supply or Demand
Total Supply < Total Demand	Inflow-Outflow <=Supply or Demand
Total Supply = Total Demand	Inflow-Outflow = Supply or Demand

12

Defining the Constraints

- In the BMC problem:

  Total Supply = 500 cars

  Total Demand = 480 cars

- So for each node we need a constraint of the form:

  Inflow - Outflow >= Supply or Demand

- Constraint for node 1:

  $-X_{12} - X_{14} >= -200$ **(there is no inflow for node 1!)**

- This is equivalent to:

  $+X_{12} + X_{14} <= 200$

13

Defining the Constraints

- Flow constraints

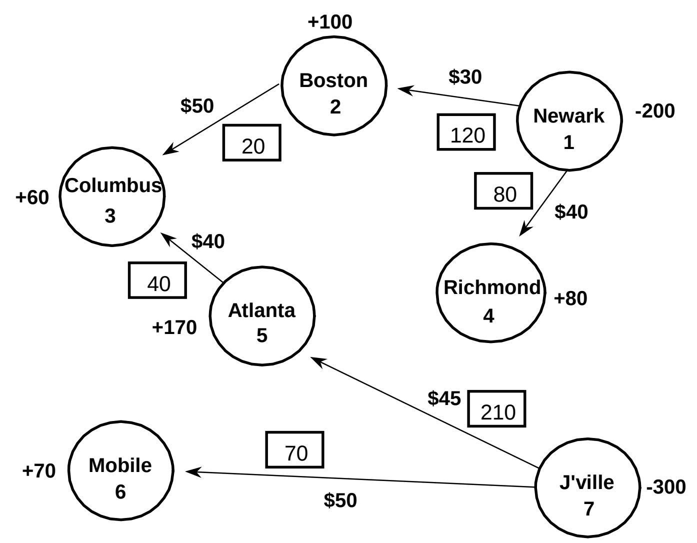
$-X_{12} - X_{14} >= -200$	} node 1
$+X_{12} - X_{23} >= +100$	} node 2
$+X_{23} + X_{53} - X_{35} >= +60$	} node 3
$+ X_{14} + X_{54} + X_{74} >= +80$	} node 4
$+ X_{35} + X_{65} + X_{75} - X_{53} - X_{54} - X_{56} >= +170$	} node 5
$+ X_{56} + X_{76} - X_{65} >= +70$	} node 6
$-X_{74} - X_{75} - X_{76} >= -300$	} node 7

- Nonnegativity conditions

  $X_{ij} >= 0$ for all (i,j)

14

Optimal Solution to the BMC Problem



15

The Shortest Path Problem

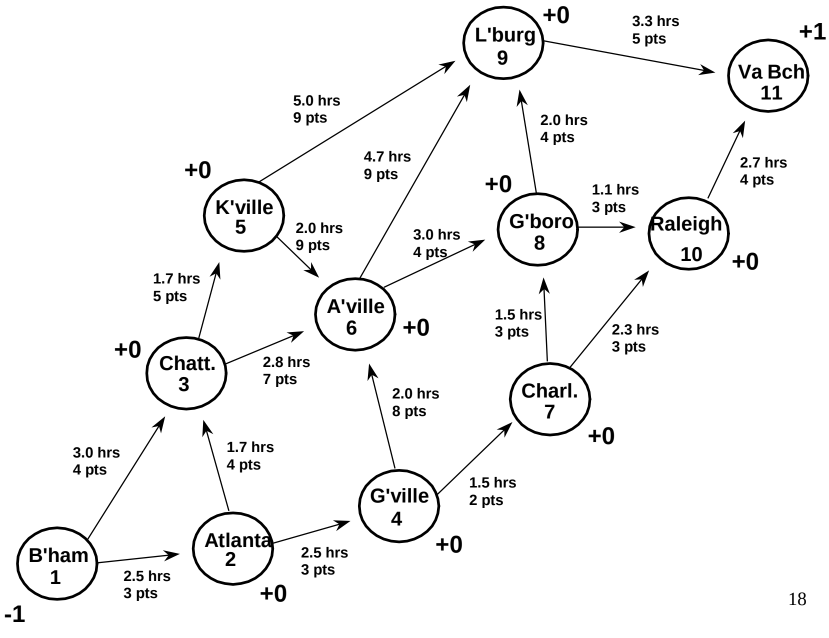
- Many decision problems boil down to determining the shortest (or least costly) route or path through a network.
  - Ex. Emergency Vehicle Routing
- This is a special case of a transshipment problem where:
  - there is a supply node with a supply of –1
  - there is a demand node with a demand of +1
  - all other nodes have supply/demand of +0

16

The American Car Association

The ACA provides a service to its clients: you provide your origin and destination cities, then it determines the quickest way (shortest path) for you to drive there.

The American Car Association



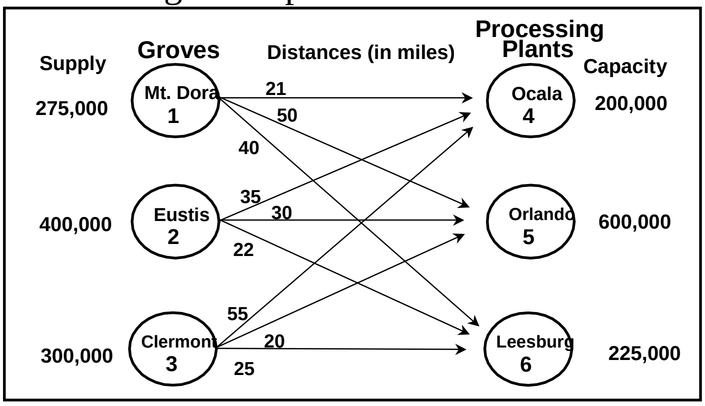
Solving the Problem

- There are two possible objectives for this problem
  - Finding the quickest route (minimizing travel time)
  - Finding the most scenic route (maximizing the scenic rating points)
- Model & solve (using Excel) either of these shortest path problems for **homework**.

19

Transportation & Assignment Problems

- Some network flow problems don’t have trans-shipment nodes; only supply and demand nodes.
- Groves assignment problem:



- These problems are implemented more effectively using the LP technique described in Lectures 2-3

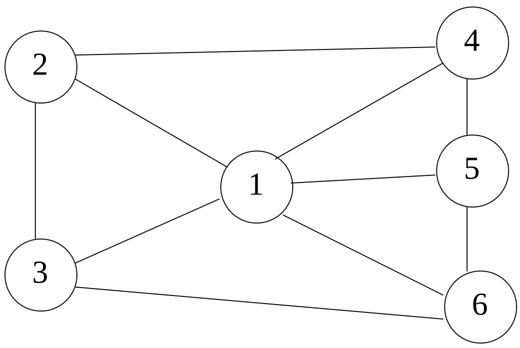
20

Minimal spanning tree problems

- For a network with *n* nodes, a spanning tree is a set of *n–1* arcs that connects all the nodes and contain no loops.
- A minimum spanning tree problem involves determining the spanning tree with minimal total length (or cost) of the selected arcs.

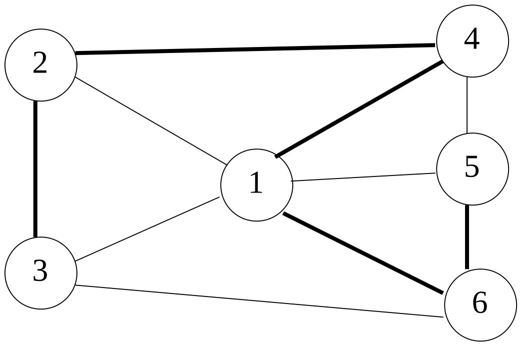
21

Spanning tree

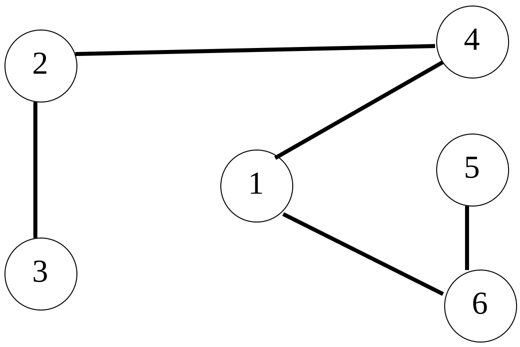


22

Spanning tree

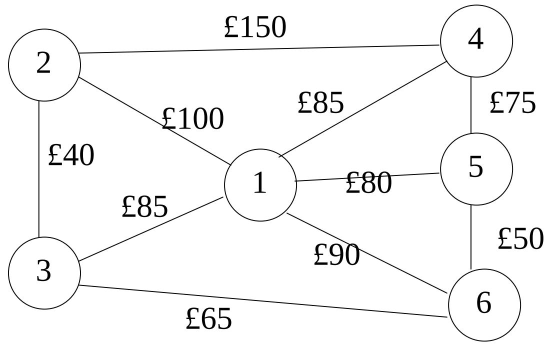


Spanning tree



Minimal spanning tree problems

- Suppose you are requested to design a computer LAN, where all computers are connected together, not necessarily directly one to each other. The network in the picture represents all the possible links that can be activated, and the £ amount on each link represents the cost of making the connection:

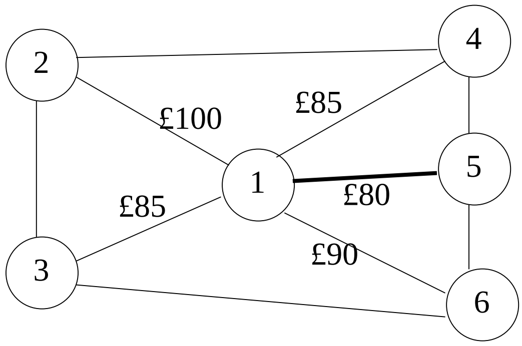


An algorithm for the Minimal spanning tree problem

1. Select any node and call this the current subnetwork.
2. Add to the current subnetwork the cheapest arc, and the corresponding node, that connect any node within the network to any node not in the current subnetwork. Call this the current subnetwork.
3. If all nodes are in the subnetwork STOP, this is the optimal solution. Otherwise return to step 2.

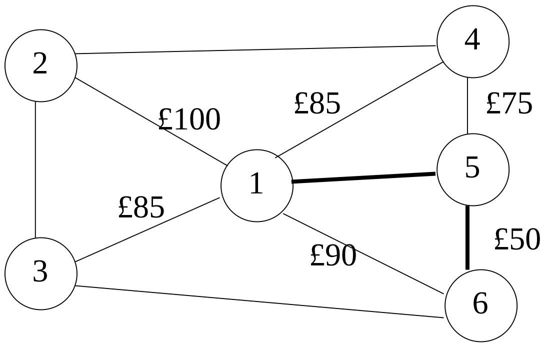
An algorithm for the Minimal spanning tree problem

1. Select 1 as a starting subnetwork.
2. The cheapest arc is (1,5). The new subnetwork is {1,5}.
3. Four nodes remain unconnected. Go to step 2.



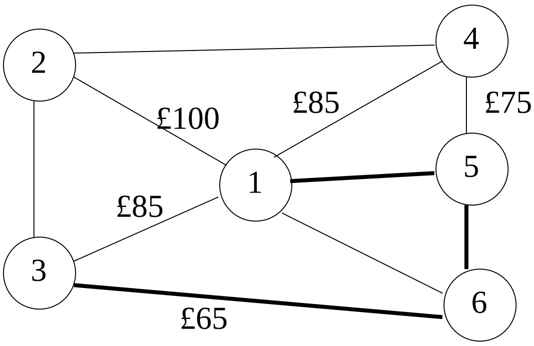
An algorithm for the Minimal spanning tree problem

2. The cheapest arc is (5,6). The new subnetwork is {1,5,6}.
3. Three nodes remain unconnected. Go to step 2.



An algorithm for the Minimal spanning tree problem

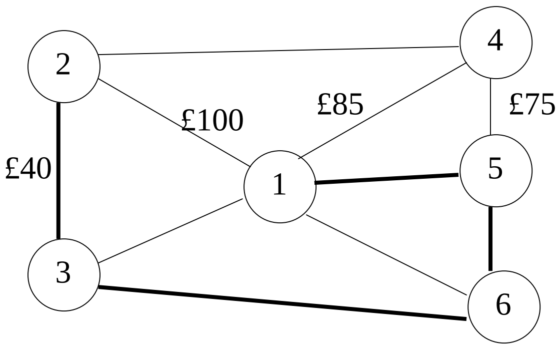
2. The cheapest arc is (3,6). The new subnetwork is {1,3,5,6}.
3. Two nodes remain unconnected. Go to step 2.



29

An algorithm for the Minimal spanning tree problem

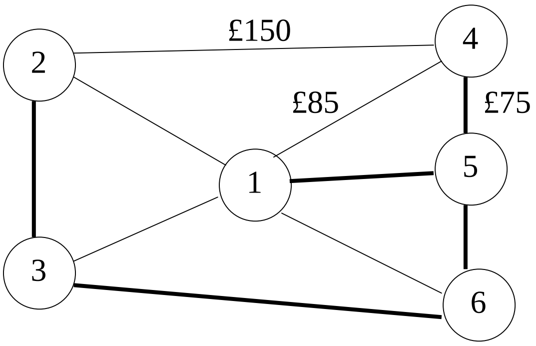
2. The cheapest arc is (2,3). The new subnetwork is {1,2,3,5,6}.
3. One node remains unconnected. Go to step 2.



30

An algorithm for the Minimal spanning tree problem

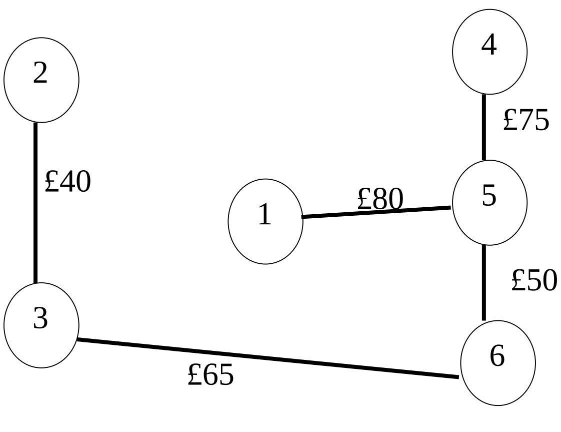
2. The cheapest arc is (4,5). The new subnetwork is {1,2,3,4,5,6}.
3. No nodes remain unconnected. STOP.



31

An algorithm for the Minimal spanning tree problem

The optimal solution:



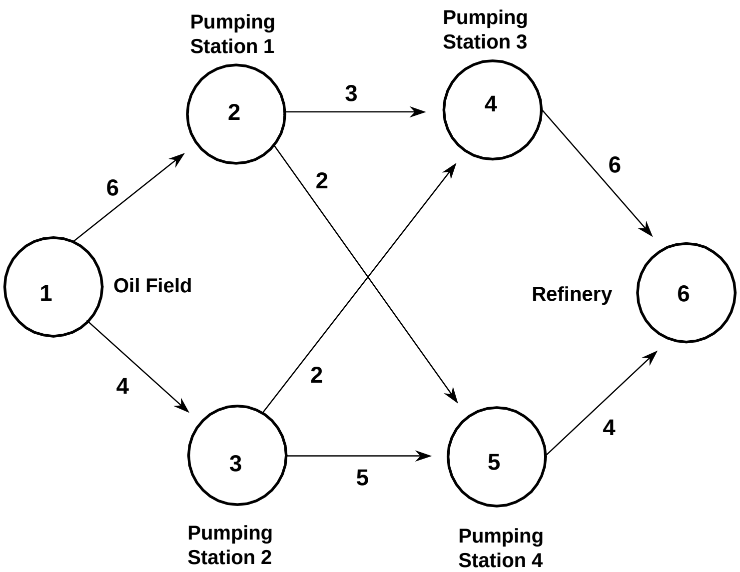
32

The Maximal Flow Problem

- In some network problems, the objective is to determine the maximum amount of flow that can occur through a network.
- The arcs in these problems have upper and lower flow limits.
- Examples
  - How much water can flow through a network of pipes?
  - How many cars can travel through a network of streets?

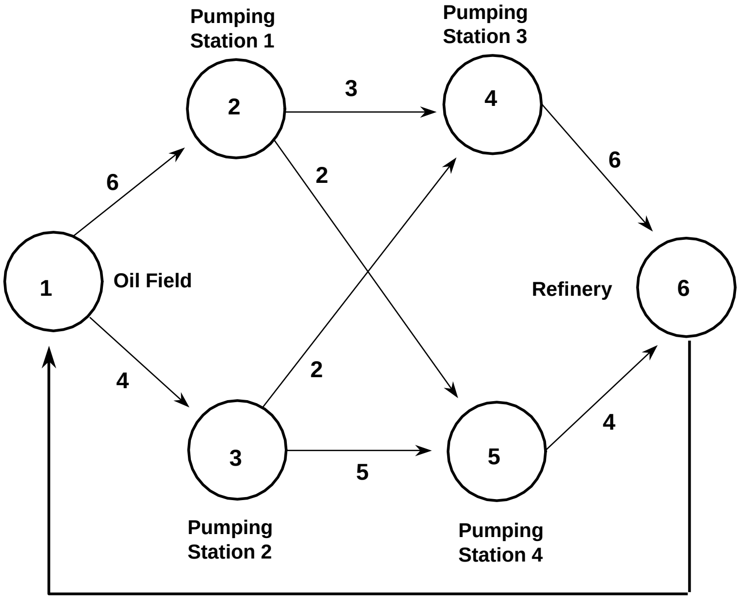
33

The Northwest Petroleum Company



34

The Northwest Petroleum Company



35

Formulation of the Max Flow Problem

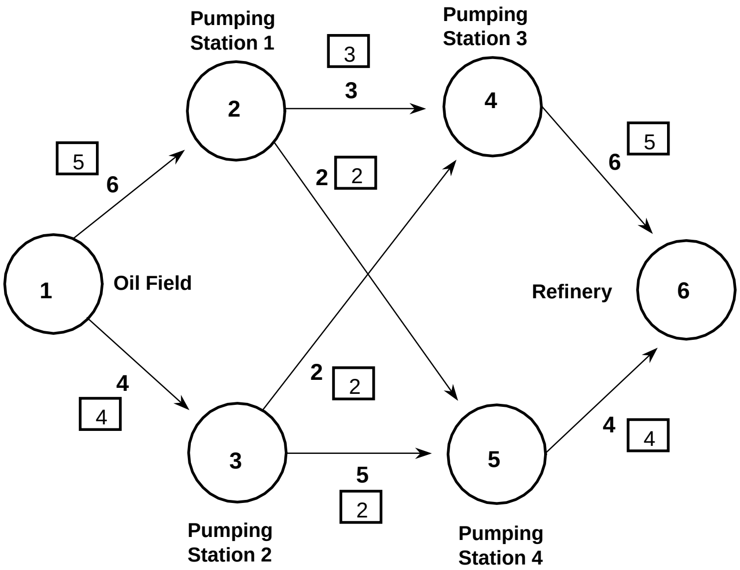
MAX: X_{61}

Subject to:
$$
\begin{aligned}
&+X_{61} - X_{12} - X_{13} = 0\\
&+X_{12} - X_{24} - X_{25} = 0\\
&+X_{13} - X_{34} - X_{35} = 0\\
&+X_{24} + X_{34} - X_{46} = 0\\
&+X_{25} + X_{35} - X_{56} = 0\\
&+X_{46} + X_{56} - X_{61} = 0
\end{aligned}
$$

with the following bounds on the decision variables:

$0 \le X_{12} \le 6$	$0 \le X_{25} \le 2$	$0 \le X_{46} \le 6$
$0 \le X_{13} \le 4$	$0 \le X_{34} \le 2$	$0 \le X_{56} \le 4$
$0 \le X_{24} \le 3$	$0 \le X_{35} \le 5$	$0 \le X_{61} \le \text{inf}$

Optimal Solution



37

Max Flow general formulation

Network *G=(N,A)*

N=set of nodes, A=set of (directed) arcs

s: source node, t: sink node, c_{ij}: capacity of arc (i,j)

f: value of an s-t flow

$$
\begin{aligned}
\max \quad & f \\
s.t. \quad & f - \sum_{j:(s,j)\in A} x_{sj} = 0 \\
& -f + \sum_{j:(j,t)\in A} x_{jt} = 0 \\
& \sum_{j:(i,j)\in A} x_{ij} - \sum_{j:(j,i)\in A} x_{ji} = 0 \qquad \forall i \in N - \{s,t\} \\
& 0 \le x_{ij} \le c_{ij} \qquad \forall (i,j) \in A
\end{aligned}
$$

38

Notation

- An *s*-*t* cut is a partition (*V,W*) of the nodes of *N* into sets *V* and *W* such that

$$s\in V,\ t\in W,\ V\cup W=N,\ V\cap W=\emptyset$$

- The *capacity* of an *s*-*t* cut is

$$C(V,W) = \sum_{\substack{(i,j)\in A \\ i\in V,\, j\in W}} c_{ij}$$

- We would expect that the value of an *s*-*t* flow cannot exceed the capacity of an *s*-*t* cut

39

Max Flow – Min Cut Theorem

- The value *f* of any *s*-*t* flow is no greater than the capacity *C(V,W)* of any *s*-*t* cut. Furthermore, the value of the maximal flow equals the capacity of the minimum cut, and a flow *f* and a cut *(V,W)* are jointly optimal if and only if

$$x_{ij} = 0 \quad \forall (i,j)\in A : i\in W,\ j\in V$$
$$x_{ij} = c_{ij} \quad \forall (i,j)\in A : i\in V,\ j\in W$$

40

Augmenting path

- Given a feasible s-t flow x over the network $G=(N,A)$, an **augmenting path** P is a path from s to t in the undirected graph resulting from G by ignoring arc directions with the following properties:
  - For every arc $(i,j)\in A$ that is traversed by P in the forward direction (forward arc), we have $x_{ij}<c_{ij}$. That is forward arcs of P are unsaturated.
  - For every arc $(j,i)\in A$ that is traversed by P in the backward direction (forward arc), we have $x_{ji}>0$.

41

Augmenting path (cont.)

- **Remark:** Given a feasible s-t flow x, we can increase the flow from s to t (maintaining flow conservation) by increasing the flow on every forward arc of P, and decreasing it along every backward arc.
- The maximum amount of flow augmentation along P is given by

$$\delta = \min_{(i,j)\in P} \begin{cases} c_{ij} - x_{ij} & \text{along a forward arc} \\ x_{ji} & \text{along a backward arc} \end{cases}$$

- **Theorem**: Given a feasible s-t flow x, with value f, over the network $G=(N,A)$, if there exists in G an augmenting path from s to t, then the corresponding flow value f is not maximal.

42

Finding an augmenting path
The labeling algorithm

- To each node i is assigned a two part label $\lambda(i)=(P(i),F(i))$, where $P(i)$ denotes the node from where i was labeled and $F(i)$ the amount of extra flow that can be brought from s to i.
- There are two cases:
  - If node j is unlabeled and succeeds i, then we may label j if $x_{ij}<c_{ij}$, in which case we set
    - $P(j):=i$ and $F(j):=\min\{F(i),c_{ij}-x_{ij}\}$
  - If node j is unlabeled and precedes i, then we may label j if $x_{ij}>0$, in which case we set
    - $P(j):=-i$ and $F(j):=\min\{F(i),x_{ij}\}$

43

Finding an augmenting path
The labeling algorithm

- The process of labeling outward from a given node i is called **scanning** i.
- The labeling algorithm amounts to look for an augmenting path by scanning the nodes of the network, starting from node s.
- In particular, a list containing all labeled but unscanned nodes is kept.
- The list is initialized by adding s to it.
- At each iteration an element i is selected from the list and scanned, and all nodes labeled from i are added to the list.

44

Finding an augmenting path
The labeling algorithm

- The process terminates in one of two ways:
  - Either *t* gets labeled, in which case we can reconstruct an augmenting path backwards from *t* using *P(i)*’s, and increase the current flow along the augmenting path of *F(t)*;
  - Or the list is empty, then the flow is maximal (to be proved later).

45

Solving the Max Flow Problem
The Ford-Fulkerson algorithm

- Set *x:=0*; (Initialize the flow)
- Find, if any, an augmenting path *P* and increase the flow of

$$\delta = \min_{(i,j)\in P} \begin{cases} c_{ij} - x_{ij} & \text{along a forward arc} \\ x_{ji} & \text{along a backward arc} \end{cases}$$

- If there does not exist any augmenting path, then terminate, the current flow is maximal.

46

Solving the Max Flow Problem
The Ford-Fulkerson algorithm

- Set $x:=0$; (Initialize the flow)
- Repeat
  - Set all labels to zero;
  - Set LIST:={s};
  - While (LIST≠∅) and (t is unlabeled) do
    - Let i be any node in LIST,
    - Remove i from LIST and Scan i.
    - If t is labeled then increase the flow along the augmenting path of $F(t)$

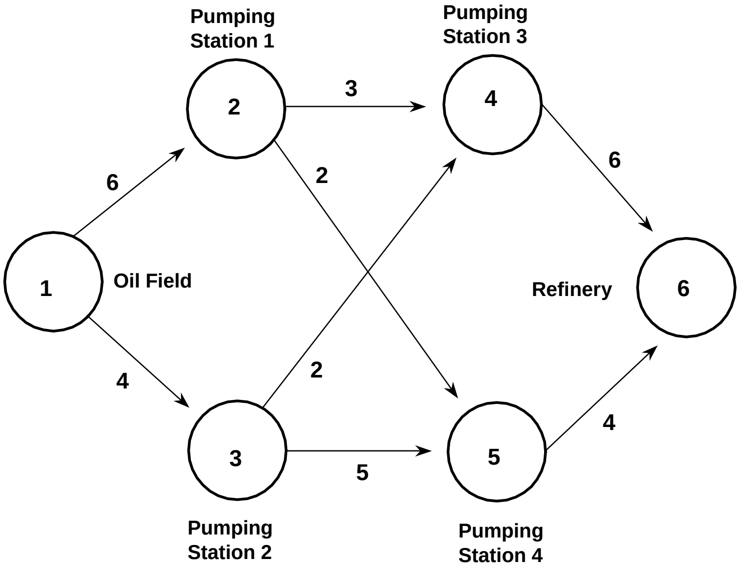
    Endwhile

  Until t is unlabeled

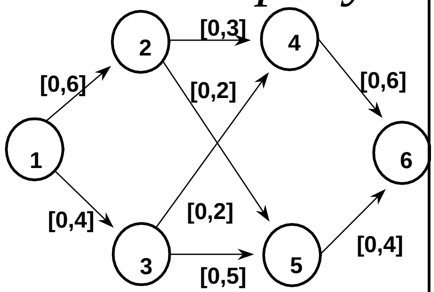
The Ford-Fulkerson algorithm
Termination and optimality

- When the capacities are integers or rational numbers, the Ford-Fulkerson labeling algorithm terminates after finitely many iterations.
- **Theorem:** When the Ford-Fulkerson labeling algorithm terminates, it does so at optimal flow.
- **Proof:** At termination of the algorithm, some nodes are labeled and some are not (at least the sink node). Call the set of labeled nodes V and the set of unlabeled nodes W. (V,W) is an s-t cut for the network G. All arcs (i,j), with $i \in V$ and $j \in W$, must be saturated, otherwise j would have been labeled when i was scanned. Similarly, all arcs (j,i), with $i \in V$ and $j \in W$, must be empty, otherwise j would have been labeled when i was scanned. Therefore, (V,W) is a min-cut, and the flow is optimal due to the max flow-min cut theorem.

The Northwest Petroleum Company



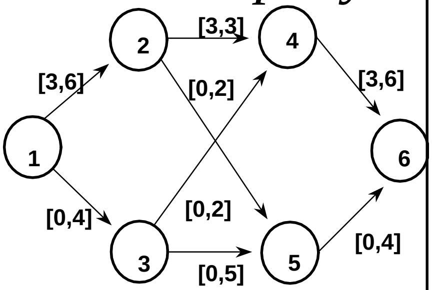
The Northwest Petroleum Company



- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=6,
  - P(3)=1,F(3)=4;
  - L:={2,3};
- Extract 2; L:={3};
- Scan 2:
  - P(4)=2,F(4)=min{6,3}=3,
  - P(5)=2,F(5)=min{6,2}=2;
  - L:={3,4,5};
- Extract 4; L:={3,5};
- Scan 4:
  - P(6)=4,F(6)=min{3,6}=3;
- 6 is labeled, increase of F(6)=3 the flow along the augmenting path **6**←P(6)=**4** ← P(4) = **2** ← P(2)=**1**.

The Northwest Petroleum Company

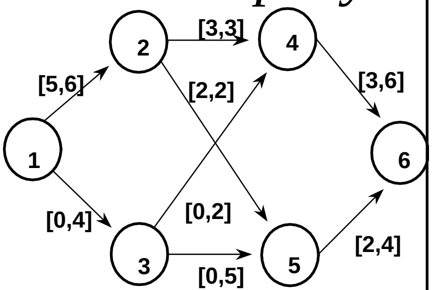
- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=3,
  - P(3)=1,F(3)=4;
  - L:={2,3};
- Extract 2; L:={3};
- Scan 2:
  - P(5)=2,F(5)=min{3,2}=2;
  - L:={3,5};
- Extract 5; L:={3};
- Scan 5:
  - P(6)=5,F(6)=min{2,6}=2;
- 6 is labeled, increase of F(6)=2 the flow along the augmenting path **6**←P(6)=**5** ← P(5) = **2** ← P(2)=**1**.



51

The Northwest Petroleum Company

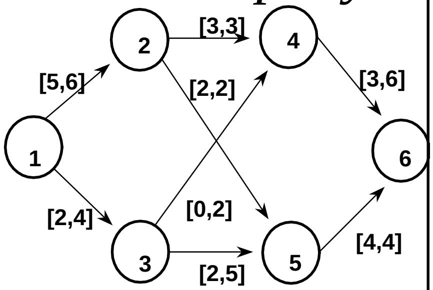
- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=1, P(3)=1,F(3)=4;
  - L:={2,3};
- Extract 2; L:={3}; Scan 2: No label.
- Extract 3; L:= ∅;
- Scan 3:
  - P(4)=3,F(4)=min{4,2}=2; P(5)=3,F(5)=min{4,5}=5;
  - L:={4,5}
- Extract 5; L:={4}
- Scan 5:
  - P(6)=5,F(6)=min{5,4–2}=2;
- 6 is labeled, increase of F(6)=2 the flow along the augmenting path **6**←P(6)=**5** ← P(5) = **3** ← P(3)=**1**.



52

The Northwest Petroleum Company

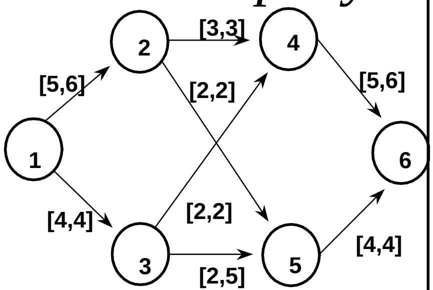
- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=1, P(3)=1,F(3)=2;
  - L:={2,3};
- Extract 3; L:= {2};
- Scan 3:
  - P(4)=3,F(4)=2;
  - P(5)=3,F(5)=min{2,3}=2;
  - L:={2,4,5}
- Extract 4; L:={2,5}
- Scan 4:
  - P(6)=4,F(6)=min{2,6–3}=2;
- 6 is labeled, increase of F(6)=2 the flow along the augmenting path **6**←P(6)=**4** ← P(4) = **3** ← P(3)=**1**.



53

The Northwest Petroleum Company

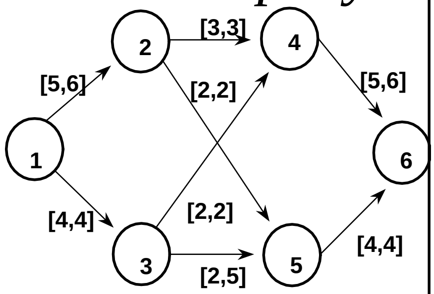
- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=1, P(3)=1,F(3)=2;
  - L:={2,3};
- Extract 3; L:= {2};
- Scan 3:
  - P(4)=3,F(4)=2;
  - P(5)=3,F(5)=min{2,3}=2;
  - L:={2,4,5}
- Extract 4; L:={2,5}
- Scan 4:
  - P(6)=4,F(6)=min{2,6–3}=2;
- 6 is labeled, increase of F(6)=2 the flow along the augmenting path **6**←P(6)=**4** ← P(4) = **3** ← P(3)=**1**.



54

The Northwest Petroleum Company

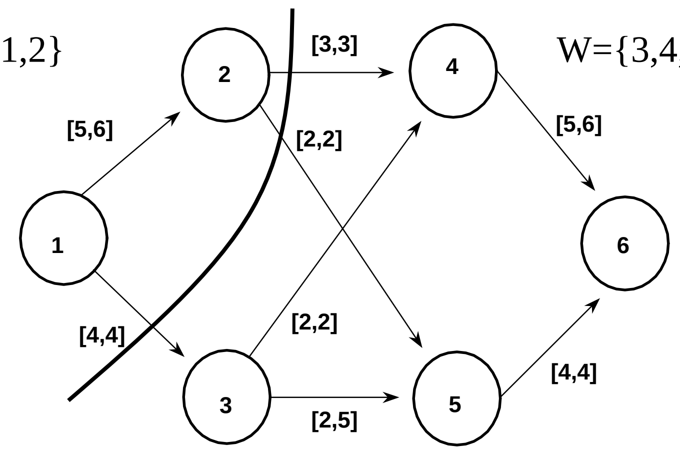
- L:={1}; Extract 1; L:=∅;
- Scan 1:
  - P(2)=1,F(2)=1;
  - L:={2};
- Extract 2; L:= ∅;
- Scan 2: No label.
- List empty!!!



The Northwest Petroleum Company

V={1,2}

W={3,4,5,6}



$C^*(V,W)=c_{13}+c_{24}+c_{25}=9$

$f^*=x_{12}+x_{13}=9$

Lecture 5 *3E4 Homework*

1.a. Model the ACA's shortest path problem to find the quickest route (minimizing travel time).

1.b Model the ACA's shortest path problem to find the the most scenic route (maximizing the scenic rating points)

1.c Solve (using Excel) either of these shortest path problems.

57

Lecture 5 *3E4 Homework*

2 *The Equipment Replacement Problem*

The problem of determining when to replace equipment is a common business problem. It can also be modeled as a shortest path problem.

2.a Model the following Compu-Train replacement problem as a pair of shortest path problems, comparing the two solutions.

58

The Compu-Train Company

- Compu-Train provides hands-on software training.
- Computers must be replaced at least every two years.
- Two lease contracts are being considered:
  - Each required $62,000 initially
  - Contract 1:
    - Prices increase 6% per year
    - 60% trade-in for 1 year old equipment
    - 15% trade-in for 2 year old equipment
  - Contract 2:
    - Prices increase 2% per year
    - 30% trade-in for 1 year old equipment
    - 10% trade-in for 2 year old equipment
- Want to determine which contract would allow to minimize the remaining leasing cost over the next five years and when, under the selected contract, the equipment should be replaced.

Lecture 5 *3E4 Homework*

3.a

Given the network in the next slide, determine the maximum flow that is possible to send from *s* to *t* using the Ford-Fulkerson algorithm. Arc capacities are showed in the picture along each arc.

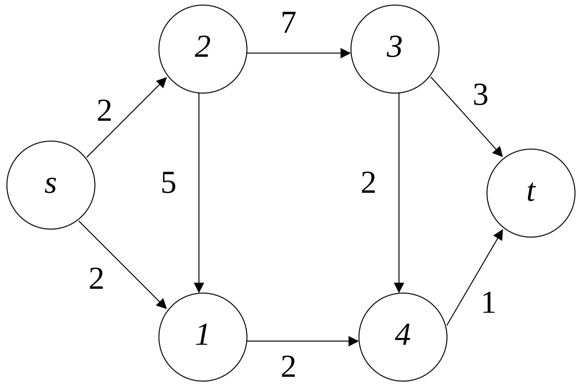
3.b

Check the optimality of the solution using the max flow – min cut theorem.

3.c

Solve the problem using Excel.

Lecture 5 *3E4 Homework* **3**



61