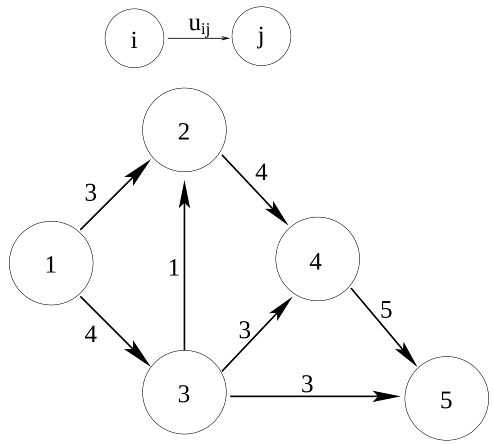
Slide 1



Slide 2

- Suppose that it possible to increase the capacity of the existing arcs in network by paying \$100 per extra unit of capacity per arc.

  For example, the capacity of arc $(1, 2)$ could be increased from 3 to 5 for a cost of \$200.

- Formulate a mathematical programming model to determine a set of minimum-cost arc-capacity increases to the network shown above so that 10 messages per minute may be transmitted from Node 1 to Node 5.

Slide 3

Let y_{ij} be the number of units of additional capacity purchased for arc $(i, j) \in A$. A correct mathematical programming formulation is shown below.

$$\begin{aligned}
\text{Minimize} \quad & 100 \sum_{(i,j)\in A} y_{ij} && (1)\\
\text{s.t.} \quad & \sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(j,i)\in A\}} x_{ji} = 0, \forall i \in N && (2)\\
& 0 \le x_{ij} \le u_{ij} + y_{ij}, \qquad (i,j)\forall \in A && (3)\\
& x_{51} \ge 10 && (4)\\
& y_{ij} \ge 0, \qquad \forall (i,j) \in A && (5)\\
& y_{ij} \text{ is integer }, \qquad \forall (i,j) \in A && (6)
\end{aligned}$$

Slide 4

The objective function (1) minimizes the cost of purchasing the additional capacity. The flow balance constraints (2) are unchanged. Constraint set (3) ensures that enough additional capacity is added to carry the messages routed on arc $(i, j) \in A$. Constraint (4) forces arc $(5, 1)$ to carry at least 10 messages. Constraints (5) and (6) force t he y_{ij}'s to be non-negative integers.

Alternatively, we can remove arc $(5, 1)$, set $b_1 = 10$ and $b_5 = -10$. This leads to the formul ation shown below.

Slide 5

$$\text{Minimize} \quad 100 \sum_{(i,j)\in A} y_{ij} \tag{7}$$

$$\text{s.t.} \quad \sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(j,i)\in A\}} x_{ji} = b_i, \forall i \in N \tag{8}$$

$$0 \le x_{ij} \le u_{ij} + y_{ij}, \qquad (i,j)\forall \in A \tag{9}$$

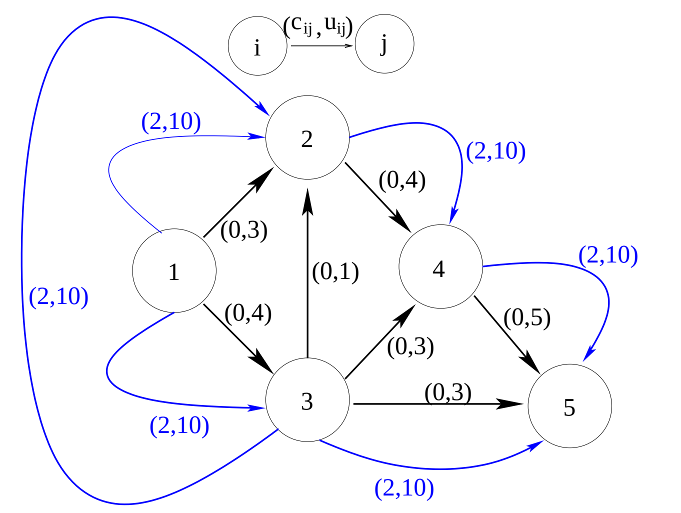
$$y_{ij} \ge 0, \qquad \forall (i,j) \in A \tag{10}$$

$$y_{ij} \text{ is integer }, \qquad \forall (i,j) \in A \tag{11}$$

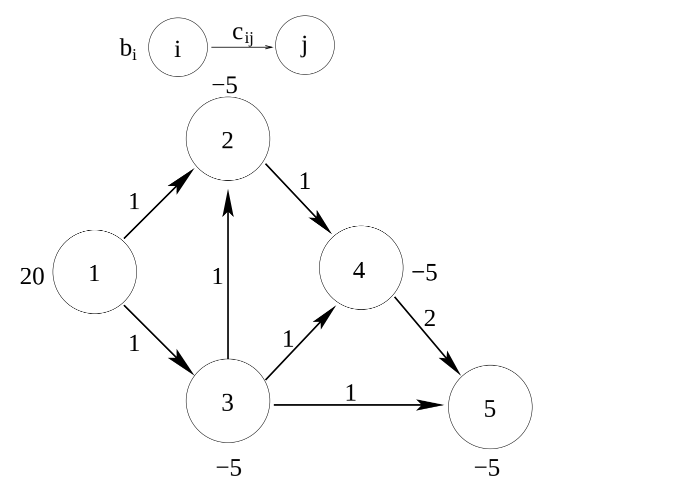
Slide 6

- Can we relax the integrality constraints on the y_{ij} variables?
  - It can be shown that the constraint matrix for the second formulation is TU, and so it follows that this problem can also be solved as an LP.
  - Alternatively, observe that the variable y_{ij} can be seen to represent the flow on an arc parallel to (i, j). Thus, this can be modeled as an instance of MCNFP.

Slide 7



Slide 8



Slide 9

$$\text{Minimize} \quad \sum_{(i,j)\in A} c_{ij}x_{ij} \tag{12}$$

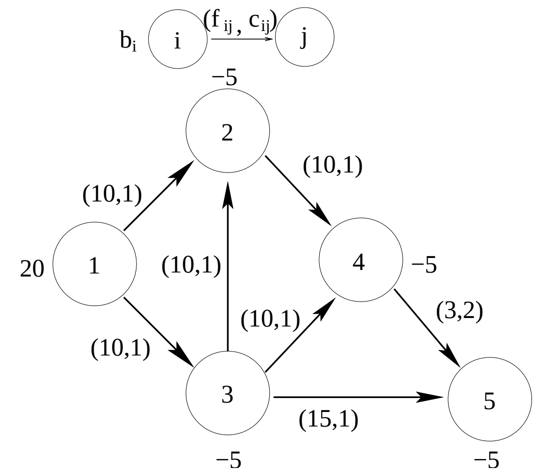
$$\text{s.t.} \quad \sum_{\{j:(i,j)\in A\}} x_{ij} - \sum_{\{j:(j,i)\in A\}} x_{ji} = b_i, \forall i \in N \tag{13}$$

$$x_{ij} \geq 0, \qquad \forall (i,j) \in A \tag{14}$$

Slide 10

```
CPLEX 8.0.0: optimal solution;
objective 30
x :=
1 2   10
1 3   10
2 4    5
3 2    0
3 4    0
3 5    5
4 5    0
;
```


Slide 11



Slide 12

For each arc (i,j), add a binary variable y_{ij} which is equal to 1 if and only if the arc carries flow (i.e. $x_{ij} > 0$). To ensure the correct relationship between x_{ij} and y_{ij}, we add the following set of constraints:

$$x_{ij} \leq 20y_{ij} \qquad \forall (i,j).$$

To account for the fixed cost of installing capacity on arc (i,j), let f_{ij} be first number given for arc (i,j) in the figure and change the objective function to

$$\min \sum_{(i,j)\in A} (c_{ij}x_{ij} + f_{ij}y_{ij}).$$

Slide 13

```
minimize cost:
        10*y[1,2] + 10*y[1,3] + 10*y[2,4] +
        10*y[3,2] + 10*y[3,4] + 15*y[3,5] +
         3*y[4,5] +    x[1,2] + x[1,3] +
           x[2,4] +    x[3,2] + x[3,4] +
           x[3,5] +  2*x[4,5];
```

Slide 14

```
s.t. flow_balance[1]:
x[1,2] + x[1,3] = 20;

s.t. flow_balance[2]:
-x[1,2] + x[2,4] - x[3,2] = -5;

s.t. flow_balance[3]:
-x[1,3] + x[3,2] + x[3,4] + x[3,5] = -5;

s.t. flow_balance[4]:
-x[2,4] - x[3,4] + x[4,5] = -5;

s.t. flow_balance[5]: -x[3,5] - x[4,5] = -5;
```


Slide 15

```
s.t. capacity[1,2]: x[1,2] - 20*y[1,2] <= 0;
s.t. capacity[1,3]: x[1,3] - 20*y[1,3] <= 0;
s.t. capacity[2,4]: x[2,4] - 20*y[2,4] <= 0;
s.t. capacity[3,2]: x[3,2] - 20*y[3,2] <= 0;
s.t. capacity[3,4]: x[3,4] - 20*y[3,4] <= 0;
s.t. capacity[3,5]: x[3,5] - 20*y[3,5] <= 0;
s.t. capacity[4,5]: x[4,5] - 20*y[4,5] <= 0;
```

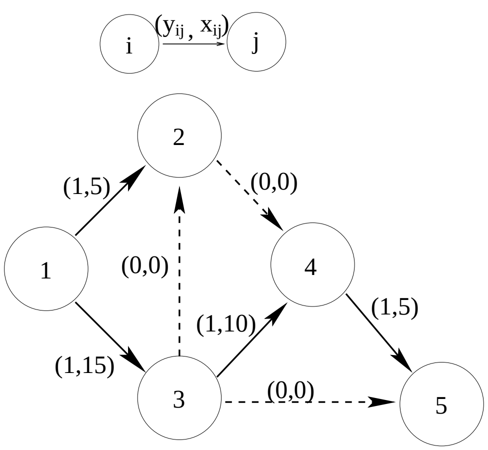
Slide 16

```
CPLEX 8.0.0: optimal integer solution;
objective 73
x :=
1 2    5
1 3   15
2 4    0
3 2    0
3 4   10
3 5    0
4 5    5
;
```

Slide 17

```
y :=
1 2   1
1 3   1
2 4   0
3 2   0
3 4   1
3 5   0
4 5   1
;
```

Slide 18



Slide 19

LP Relaxation

```
CPLEX 8.0.0: optimal solution;
objective 46.25
9 dual simplex iterations (1 in phase I)
x :=
1 2   10
1 3   10
2 4    5
3 2    0
3 4    0
3 5    5
4 5    0
;
```

Slide 20

```
y :=
1 2   0.5
1 3   0.5
2 4   0.25
3 2   0
3 4   0
3 5   0.25
4 5   0
;
```