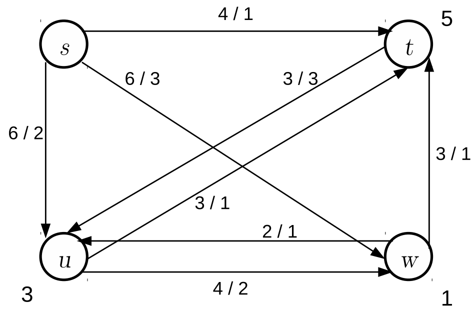
Problem 1: extended formulation for fixed charge network flow problem

Consider the network topology given by the graph $G = (V, E)$ in the figure, where $V = \{s, t, u, w\}$. Each node represents a router of the network, and each arc $(i, j) \in E$ with $i, j \in V$ represents a link. The numbers associated to each arc (i, j) correspond to u_{ij} / f_{ij}, where $u_{ij} > 0$ is the bandwith (arc capacity) in Mbps and $f_{ij} > 0$ is the fixed cost of using the arc.



There is a single source node s and a subset $K \subseteq V \setminus \{s\}$ of destinations (terminals). Next to each destination node $i \in K$ we report its flow (bandwith) demand b_i. For the destination nodes $i \in K$ we assume $b_i > 0$. For the source node s, b_s represents the total availability, that we assume equivalent to the sum of the demands of all nodes $i \in K$, that is, $b_s = -\sum_{i \in K} b_i$.

The aim is to determine the feasible minimum cost flow that satisfies the requirements of all destinations $i \in K$.

1) Write in AMPL the natural MILP formulation for fixed charge network flow problem described in class.
2) Solve its linear relaxation. Is the optimal solution integer?
3) Propose a stronger extended formulation. *Hint: consider for each arc $(i, j) \in E$, a specific flow variable for each destination node.* Write it in AMPL and solve its linear relaxation.
4) Compare the optimal solutions obtained in points 2) and 3). Is the solution of the linear relaxation of the natural formulation feasible for the linear relaxation of the extended formulation?

AMPL model (file network-i.mod)

```
# SETS

set V;
set E within {V,V};


# PARAMETERS

param source symbolic in V;
param u{(i,j) in E} >= 0;
param f{(i,j) in E} >= 0;
param b{i in V} default if i=source then (- sum{j in V: j<>source} b[j]);
```

AMPL runfile (file network.run)

```
reset;
model network.mod
data network.dat
option solver cplex;
solve;
display x,y;
option relax_integrality 1;
solve;
display{(i,j) in E} (x[i,j],y[i,j]);
```

Data (file network.dat), optimal value: 6

```
data;

set V := s t u w;

param source := s;


param: E: u :=
        s t 4
        s u 6
        s w 6
        t u 3
        u t 3
        u w 4
        w t 3
        w u 2
;

param: f :=
        s t 1
        s u 2
        s w 3
        t u 3
```

```
        u t 1
        u w 2
        w t 1
        w u 1
;

param: b :=
        t 5
        u 3
        w 1

;
```

SOLUTION

Formulation

Sets

- V: nodes
- $E \subseteq V \times V$: arcs
- $K \subseteq V$: destination nodes

Parameters

- $s \in V \setminus K$: source node
- u_{ij}: capacity of the arc (i, j), with $(i, j) \in E$
- f_{ij}: cost of the arc (i, j), with $(i, j) \in E$
- b_i: flow requirements in node i, with $b_i = \begin{cases} > 0 & i \in K \\ < 0 & i = s \\ 0 & otherwise \end{cases}$

Decision variables

- x_{ij}: flow sent over the arc (i, j), with $(i, j) \in E$
- y_{ij}: binary variable indicating if $(i, j) \in E$ is used

Model

For each node $i \in V$, let $\delta^+(i) = \{j \in V : (i, j) \in E\}$, $\delta^-(i) = \{j \in V : (j, i) \in E\}$.

$$\begin{array}{llll}
\min & \displaystyle\sum_{(i,j) \in E} f_{ij} y_{ij} & & \text{(value)} \\
\text{s.t.} & & & \\
& \displaystyle\sum_{j \in \delta^-(i)} x_{ji} - \sum_{j \in \delta^+(i)} x_{ij} = b_i & i \in V & \text{(balance)} \\
& x_{ij} \leq u_{ij} y_{ij} & (i, j) \in E & \text{(capacity)} \\
& x_{ij} \geq 0 & (i, j) \in E & \text{(nonnegative var.)} \\
& y_{ij} \in \{0, 1\} & (i, j) \in E & \text{(binary var.)}
\end{array}$$

AMPL model (file network.mod)

```
# SETS

set V;
set E within {V,V};


# PARAMETERS

param source symbolic in V;
param u{(i,j) in E}, >=0;
param f{(i,j) in E}, >= 0;

param b{i in V} default if i=source then (- sum{j in V: j<>source} b[j]);


# VARIABLES

var x{(i,j) in E}, >=0;
var y{(i,j) in E}, binary;


# OBJECTIVE FUNCTION

minimize fixed_costs:
        sum{(i,j) in E} f[i,j]*y[i,j];


# CONSTRAINTS

subject to balance{i in V}:
        sum{j in  V : (j,i) in E} x[j,i] - sum{j in  V : (i,j) in E} x[i,j] = b[i];

subject to capacity{(i,j) in E}:
        x[i,j] <= u[i,j]*y[i,j];
```

AMPL runfile (file network.run)

```
reset;
model network.mod
data network.dat
option solver cplex;
solve;
display x,y;
option relax_integrality 1;
solve;
display{(i,j) in E} (x[i,j],y[i,j]);
```

Data (file `network.dat`), optimal value: 6

```
data;

set V := s t u w;

param source := s;


param: E: u :=
        s t 4
        s u 6
        s w 6
        t u 3
        u t 3
        u w 4
        w t 3
        w u 2
;

param: f :=
        s t 1
        s u 2
        s w 3
        t u 3
        u t 1
        u w 2
        w t 1
        w u 1
;

param: b :=
        t 5
        u 3
        w 1

;
```

Soluzione

```
CPLEX 12.7.1.0: optimal integer solution; objective 6
11 MIP simplex iterations
0 branch-and-bound nodes
:      x   y    :=
s t    3   1
s u    6   1
s w    0   0
t u    0   0
u t    2   1
u w    1   1
w t    0   0
w u    0   0
```

```
;

CPLEX 12.7.1.0: optimal solution; objective 3.166666667
5 dual simplex iterations (0 in phase I)
:   x[i,j]    y[i,j]     :=
s t    4     1
s u    4     0.666667
s w    1     0.166667
t u    0     0
u t    1     0.333333
u w    0     0
w t    0     0
w u    0     0
;
```

Multi-commodity formulation

Sets

- V: nodes
- $E \subseteq V \times V$: arcs
- $K \subseteq V$: destination nodes

Parameters

- $s \in V \setminus K$: source node
- u_{ij}: capacity of the arc (i,j), with $(i,j) \in E$
- f_{ij}: cost of the arc (i,j), with $(i,j) \in E$
- b_i: flow requirement in node i, with $b_i = \begin{cases} > 0 & i \in K \\ < 0 & i = s \\ 0 & otherwise \end{cases}$
- d_i^k: flow requirement directed to k in node i, that is $d_i^k = \begin{cases} b_i & i = k \\ -b_k & i = s \\ 0 & otherwise \end{cases}$

Decision variables

- x_{ij}^k: flow sent on the arc (i,j) for the destination node $k \in K$
- y_{ij}: binary variable indicating if arc $(i,j) \in E$ is used

Model

$$
\begin{array}{llll}
\min & \sum\limits_{(i,j)\in E} f_{ij} y_{ij} & & \text{(value)} \\
\text{s.t.} & & & \\
& \sum\limits_{j\in\delta^-(i)} x_{ji}^k - \sum\limits_{j\in\delta^+(i)} x_{ij}^k = d_i^k & i \in V, k \in K & \text{(balance)} \\
& x_{ij}^k \le \min\{u_{ij}, b_k\} y_{ij} & (i,j) \in E, k \in K & \text{(capacity restricted to } k\text{)} \\
& \sum\limits_{k\in T} x_{ij}^k \le u_{ij} & (i,j) \in E & \text{(capacity)} \\
& x_{ij}^k \ge 0 & (i,j) \in E, k \in K & \text{(nonnegative var.)} \\
& y_{ij} \in \{0,1\} & (i,j) \in E & \text{(binary var.)}
\end{array}
$$

Since for each arc $(i, j) \in E$ the total flow quantity through it is $\sum_{k \in K} x_{ij}^k$, we clearly have that $x_{ij} = \sum_{k \in K} x_{ij}^k \quad \forall i, j \in E$, where x_{ij} are the flow variables of the previous, single-flow formulation. The multi-commodity formulation is obtained extending the space of the decision variables of the single-flow formulation.

AMPL model (file `network-2.mod`)

```
# SETS

set V;
set E within {V,V};
set K within V;


# PARAMETERS

param source symbolic in V;
param u{(i,j) in E}, >=0;
param f{(i,j) in E}, >= 0;

param b{i in V} default if i=source then (- sum{j in V: j<>source} b[j]);

param d{i in V, k in K} default if i<>source and i=k then b[i]
                                                else if i=source then -b[k]
                                                else 0;


# VARIABLES

var y{(i,j) in E}, binary;
var x{(i,j) in E, k in K}, >=0, <=u[i,j];


# OBJECTIVE FUNCTION

minimize fixed_costs:
        sum{(i,j) in E} f[i,j]*y[i,j];


# CONSTRAINTS

subject to balance{i in V, k in K}:
        sum{(j,i) in E} x[j,i,k] - sum{(i,j) in E} x[i,j,k] = d[i,k];

subject to capacity{(i,j) in E, k in K}:
        x[i,j,k] <= min(u[i,j],b[k])*y[i,j];

subject to total_flow{(i,j) in E}:
        sum{k in K} x[i,j,k] <= u[i,j];
```

AMPL runfile (file `network-2.run`)

```
reset;
model network-2.mod
data network-2.dat

option solver cplex;
solve;
```

```
display{k in K}:{(i,j) in E} x[i,j,k];
display{(i,j) in E} (sum{k in K} x[i,j,k],y[i,j]);

option relax_integrality 1;
solve;
display{k in K}:{(i,j) in E} x[i,j,k];
display{(i,j) in E} (sum{k in K} x[i,j,k],y[i,j]);
```

Data (file `network-2.dat`), optimal value: 6

```
data;

set V := s t u w;

set K := t u w;

param source := s;


param: E: u :=
        s t 4
        s u 6
        s w 6
        t u 3
        u t 3
        u w 4
        w t 3
        w u 2
;

param: f :=
        s t 1
        s u 2
        s w 3
        t u 3
        u t 1
        u w 2
        w t 1
        w u 1
;

param: b :=
        t 5
        u 3
        w 1

;
```

Soluzione

```
CPLEX 12.7.1.0: optimal integer solution; objective 6
12 MIP simplex iterations
0 branch-and-bound nodes
x[i,j,'t'] :=
s t   3
s u   2
s w   0
t u   0
u t   2
u w   0
w t   0
w u   0
;

x[i,j,'u'] :=
s t   0
s u   3
s w   0
t u   0
u t   0
u w   0
w t   0
w u   0
;

x[i,j,'w'] :=
s t   0
s u   1
s w   0
t u   0
u t   0
u w   1
w t   0
w u   0
;

:   sum{k in K} x[i,j,k] y[i,j]    :=
s t          3           1
s u          6           1
s w          0           0
t u          0           0
u t          2           1
u w          1           1
w t          0           0
w u          0           0
;

CPLEX 12.7.1.0: optimal solution; objective 5.333333333
16 dual simplex iterations (0 in phase I)
x[i,j,'t'] :=
s t   4
s u   1
```

```
s w   0
t u   0
u t   1
u w   0
w t   0
w u   0
;

x[i,j,'u'] :=
s t   0
s u   3
s w   0
t u   0
u t   0
u w   0
w t   0
w u   0
;

x[i,j,'w'] :=
s t   0
s u   1
s w   0
t u   0
u t   0
u w   1
w t   0
w u   0
;

:   sum{k in K} x[i,j,k]    y[i,j]     :=
s t           4            1
s u           5            1
s w           0            0
t u           0            0
u t           1            0.333333
u w           1            1
w t           0            0
w u           0            0
;
```

The multi-commodity formulation is stronger than the initial one, as it is possible to notice from the optimal value of the linear relaxation, which is significantly closer to the optimal value of the original problem.
The solution obtained with the relaxation of point 2) is not feasible for the new model, since the objective function value in that case is strictly smaller than the one of the multi-commodity formulation.
Considering the multi-commodity formulation, we can observe that the constraints

$$x_{ij}^k \le \min\{u_{ij}, b_k\} y_{ij} \quad (i,j) \in E, k \in K$$

force the variable y_{ij} to take value 1 corresponding to partial flows x_{ij}^k which are equivalent to the entire requirement b_k of the destination k. As an example, the flow directed to w on the arc (s, u) is equal to 1, that is $x_{su}^w = b_w$, therefore the capacity constraint implies $y_{su} = 1$. In the solution to question 2), on the other hand, we have $y_{su} = 2/3$, not feasible for the new formulation.