CUTTING PLANE ALGORITHM

Multicommodity network flow problem

Joaquim Girbau

Ignasi Mañé

22-03-2019

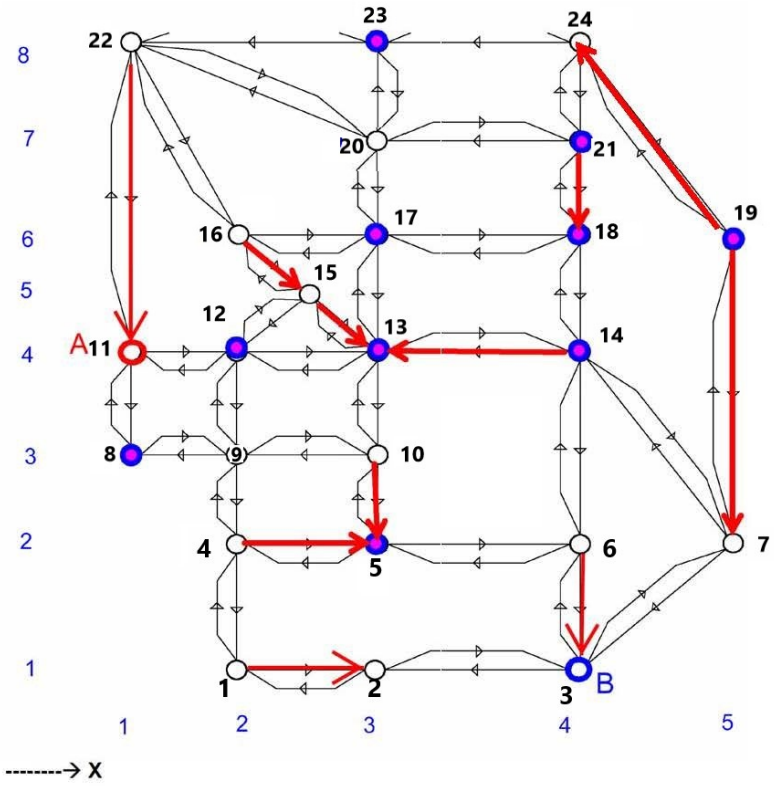
CONTENTS

INTRODUCTION

The following multicommodity network flow problem must be solved using the cutting plane decomposition algorithm by dualizing the joint capacity constraint.

$$
\begin{aligned}
Min_x \quad & \sum_{\ell \in O} c^T x^\ell \\
s.t: \quad & Bx^\ell = p^\ell, \quad \ell \in O \\
& x^\ell \geq 0 \\
& \sum_{\ell \in O} x^\ell \leq d
\end{aligned}
$$

The problem in a graphic way is:



All the blue (exterior) and pink (interior) nodes have to obtain 5 units of the product that starts in A (node 11) and B (node 3). For this assignment, the red links will be read as black links.

All the links have a exploitation cost for each unit of flow passes through them depending on the coordinates (x, y) of the nodes which form the link:

$$c_{i,j} = 10 + \left(\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 \right)^{\frac{3}{2}}, (i,j) \in A$$

All the nodes belong to the set N (from 1 to 24). The set O contains the origin nodes (A and B).

WITHOUT CAPACITIES

First, we solve the problem without capacities. And we will take four links with the maximum flows to fix them in a smaller capacity.

1. *Solve initially the problem without capacities on links, i.e.:*

$$\begin{aligned} Min_x \quad & \sum_{\ell \in O} c^T x^\ell \\ s.t: \quad & Bx^\ell = p^\ell, \quad \ell \in O \\ & x^\ell \geq 0 \end{aligned}$$

We implement it into the mod file. First, we define the variables and the links (if we define them as nodes and arcs, it is not necessary to put the second constraint):

```
var xx{A};
node I {i in N, l in O}: net_out=t[i,l];
arc xl {(i,j) in A, l in O}>=0: from I [i,l], to I [j,l];
```

Then, we define the objective function:

```
minimize COST: sum{l in O} (sum {(i,j) in A} c[i,j,l]*xl[i,j,l]);
```

Finally, we analyse the total flow for each link (we sum the flow per origin):

```
subject to total_flow {(i,j) in A}: xx[i,j] = sum{l in O} xl[i,j,l];
```

In the cuttingOGE.dat file, we insert the data: coordinates (xc, yc), the sets of nodes (N), origins (O) and arcs (A), and the total balance of flows (t).

2. *Report the solution and optimal function value. Then select four links and set on them a global capacity bound violated by the reported solution of the uncapacitated problem. Edit the file caps.dat with the selected values.*

```
CPLEX 12.8.0.0: optimal solution; objective 4785.043835
Network extractor found 93 nodes and 228 arcs.
31 network simplex iterations.
52 dual simplex iterations (0 in phase I)
```

From	To	Flow
2	1	5
1	4	5
3	2	5
4	9	5
6	5	5
3	7	5
7	19	5
9	8	5
11	8	5
12	9	5
9	10	5

From	To	Flow
11	12	45
12	13	35
13	12	5
13	14	5
14	18	10
17	18	5
17	20	20
18	21	5
20	23	10
20	21	10
21	24	5

From	To	Flow
24	19	5
10	5	5
3	6	40
14	13	20
13	17	35
6	14	35

WITH CAPACITIES

The 4 links with the largest flow values are: (11,12) with 45 units of flow, (3,6) with 40 units, (6,14) with 35 units, and (12,13) with 35 units.

Using this information we now set less units of flow than those of the solution gave us as maximum capacities for these four arcs, so that the total units of flow of each arc must not exceed: (11,12) 25 units of flow, (3,6) 20 units, (6,14) 15 units, and (12,13) 15 units.

SOLVER (CPLEX)

Now, we solve the problem with capacities using cplex to see whether it is feasible or not. Now, the problem to solve is:

$$\begin{aligned} Min_x \quad & \sum_{\ell \in O} c^T x^\ell \\ s.t: \quad & Bx^\ell = p^\ell, \quad \ell \in O \\ & x^\ell \geq 0 \\ & \sum_{\ell \in O} x^\ell \leq d \end{aligned}$$

So, in the .mod file in AMPL, we have to add the constraint caps:

```
subject to caps {(i,j) in A}: xx[i,j]<=y[i,j];
```

3. *With the joint capacities on the selected links solve the capacitated problem using the AMPL files. (cuttingOGE.mod, .dat, .run, caps.dat). If the capacitated problem results unfeasible, then try with new capacity values.*

```
CPLEX 12.8.0.0: optimal solution; objective 5234.150765
Network extractor found 93 nodes and 228 arcs.
89 network simplex iterations.
39 dual simplex iterations (0 in phase I)
```

From	To	Flow
2	1	10
1	4	10
3	2	10
4	9	10
9	4	5
6	5	5
3	7	20
7	14	15
7	19	5
8	9	20
9	8	5

From	To	Flow
11	8	25
9	12	5
9	10	15
10	13	15
11	12	25
12	15	5
12	13	15
13	14	5
14	18	20
15	16	5
16	17	5

From	To	Flow
17	18	5
18	17	10
17	20	20
18	21	5
20	23	10
20	21	10
21	24	5
24	19	5
4	5	5
3	6	20
14	13	5
13	17	20
6	14	15

It has been solved, so it is feasible (the capacities are not exceeded).

4. *Implement the cutting plane algorithm. Note that the algorithm requires an initial feasible solution of the capacitated problem. In order to determine one, use the network's transformation that uses an artificial node and artificial links, being the costs of associated to artificial links the constant "big-M"*

Mathematical Formulation to have a trivial initial basic solution

To solve the problem using the cutting plane algorithm we need a starting point, that is, an initial feasible solution from which start the successive iterations. However, finding an initial feasible solution in a capacity constrained problem might be a very difficult task. To overcome this issue, we will reformulate the problem to an equivalent problem (with the same final solution) in which finding an initial basic solution is rather trivial.

The well-known idea is to create an artificial node to which each injection node or consumption node is connected by a highly cost artificial arc. Since these artificial costs are high enough, at the final solution, the flow of the artificial arcs will be reallocated to real arcs if there exists a plausible solution for the initial problem, otherwise, at the optimal solution, artificial arcs will still have non-zero flow values. The mathematical formulation of this new problem can be written as follows:

Sets

N: Set of nodes

A: Set of arcs

O: Set of origins

$\widehat{N}$: Set of artificial nodes

Parameters

c_{ij}: exploitation costs per arc $(i,j) \in A$

x_{ij}: total flow per arc $(i,j) \in A$

p_{il}: demand of node $i \in N$ from the origin node $l \in O$

P: cost of artificial arcs

Variables

x_{ij}: total flow per arc $(i,j) \in A$

x_{ijl}: flow per arc $(i,j) \in A$ and origin $l \in O$

t_{lr}: flow from origin $l \in O$ to artificial node $r \in \widehat{N}$

t_{ril}: flow from artificial node $r \in \widehat{N}$ to node I $\in N$ per origin $l \in O$

Objective Function

$$Minimize_x \sum_{(i,j)\in A} c_{ij} x_{ij} + P \sum_{\substack{l\in O \\ r\in \widehat{N}}} t_{lr} + P \sum_{\substack{i\in N \\ l\in O\ r\in \widehat{N}}} t_{ril}$$

Constraints

(1) Balance Flow: $\sum_{(i,j)\in A} x_{ij} - \sum_{(i,j)\in A} x_{ij} = \sum_{l\in O} P_{il}\ , \forall\ i \in N$

(2) Total flow per arc: $x_{ij} = \sum_{l\in O} x_{ijl}\ , \forall\ (i,j) \in A$

(3) Balance flow in artifical arcs: $\sum_{\substack{l\in O \\ r\in \widehat{N}}} t_{lr} = \sum_{\substack{i\in N \\ l\in O\ r\in \widehat{N}}} t_{ril}$

(4) Capacities: $x_{ij} \leq y_{ij}\ , \forall\ (i,j) \in A$

Mathematical Formulation for Cutting Plane

Taking into account the previous formulation of the problem we can reformulate it in order to apply the Dantzig cutting plane algorithm.

Sets

N: Set of nodes

A: Set of arcs

O: Set of origins

$\widehat{N}$: Set of artificial nodes

Parameters

c_{ij}: exploitation costs per arc $(i,j) \in A$

x_{ij}: total flow per arc $(i,j) \in A$

p_{il}: demand of node $i \in N$ from the origin node $l \in O$

P: cost of artificial arcs

Variables

x_{ij}: total flow per arc $(i,j) \in A$

x_{ijl}: flow per arc $(i,j) \in A$ and origin $l \in O$

t_{lr}: flow from origin $l \in O$ to artificial node $r \in \widehat{N}$

t_{ril}: flow from artificial node $r \in \widehat{N}$ to node I $\in N$ per origin $l \in O$

μ_{ij}: Lagrange multiplier associated to capacity constraint in arc $(i,j) \in A$

Objective Function sub problem

$$minimize_{x,t} \left\{ \sum_{(\mathrm{i,j})\in A} \mathrm{c_{ij}} x_{ij} + \sum_{(i,j)\in A} \mu_{ij}(x_{ij} - y_{ij}) + \mathrm{P} \sum_{\substack{l\in O \\ r\in \widehat{N}}} t_{lr} + P \sum_{\substack{i\in N \\ l\in O\ r\in \widehat{N}}} t_{ril} \right\}$$

Constraints sub problem

(1) Feasibility: $x, t \in \mathrm{X}$

Objective Function master problem

$maximize_{z,\mu}\ z$

Constraints mater problem

(1) Cuts: $z \leq \sum_{(i,j)\in A} c_{ij}x_{ij} + \sum_{(i,j)\in A} \mu_{ij}(x_{ij} - y_{ij}) + P\sum_{\substack{l\in O \\ r\in \hat{N}}} t_{lr} + P \sum_{\substack{i\in N \\ l\in O\ r\in \hat{N}}} t_{ril}$

(2) Optimality: $\mu_{ij} \geq 0 \quad ,\forall\ (i,j) \in A$

5. *Report also the solution obtained (flows per origin and total flows) and the solution obtained at step 1 for the uncapacitated problem. Also, the value for the dual variables μ >=0 corresponding to the solution of the capacitated problem.*

Solutions of the Problem

Uncapacitated Problem

As requested, we first solve the uncapacitated problem using ampl and the straight forward formulation presented at the beginning of the document.

$$\begin{aligned} Min_x \quad & \sum_{\ell\in O} c^T x^\ell \\ s.t: \quad & Bx^\ell = p^\ell, \quad \ell \in O \\ & x^\ell \geq 0 \end{aligned}$$

The results at the optimal solution (total flows and total flows by origin) are presented in three different tables.

Total Flow results:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	5	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	5	.	.	.	40	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	0	.	.	.	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	5	.	.	.	.	.	.	.	.	35	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	5	.	.	.	.	.
8	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	0	.	.	.	5	.	5	.	0	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	5	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	5	.	.	.	45	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	5	.	0	.	35	.	0	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	5	.	5	0	.	35	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	20	.	.	.	.	10	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	0	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	5	.	20	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	.	.	5	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	10	0	10	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	5
22	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	5	.	0	.	0	.

Total flow from origin 3:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	5	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	5	.	.	.	40	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	0	.	.	.	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	5	.	.	.	.	.	.	.	.	35	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	5	.	.	.	.	.
8	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	0	.	.	.	5	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	0	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	0	.	0	.	0	.	0	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	5	.	0	0	.	10	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	20	.	.	.	.	10	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	0	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	0	.	5	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	.	.	5	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	0	0	5	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	0
22	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0	.

Total flow from origin 11:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	0	.	.	.	0	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	0	.	.	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.
8	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	0	.	.	.	0	.	5	.	0	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	5	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	5	.	.	.	45	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	5	.	0	.	35	.	0	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	0	.	5	0	.	25	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	0	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	5	.	15	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	.	.	0	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	10	0	5	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	5
22	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	5	.	0	.	0	.

Cutting Plane

Since we have reformulated the original problem using artificial arcs and an artificial node, finding an initial basic solution is very easy. In our case, one trivial initial feasible solution from which start the algorithm is: $t_{ril} = p_{il} \ \forall \ r \in \hat{N}, i \in N, l \in O$, $t_{lr} = 50 \ \forall \ r \in \hat{N}, l \in O$, and $x_{ij} = 0 \ \forall \ (i,j) \in A$. Notice that at the beginning, none of the real arcs have flow. However, we will see that in a few steps, the flow will have been reallocated form these arcs to the real ones.

Lagrange multipliers

In the following table there are represented all the lagrange multipliers of the optimal solution reached by the algorithm. We expect to have, as it is, no more than four lagrange multipliers greater or equal than 0 because there are only four possible active constraints related to capacities.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	0	.	.	.	5	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	0	.	.	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.
8	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	0	.	.	.	0	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	0	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	0	.	.	.	17	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	0	.	0	.	4	.	0	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	0	.	0	0	.	0	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	0	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	0	.	0	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	.	.	0	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	0	0	0	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	0
22	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0	.

Total Flow results

In the following table there is represented the total flow, regardless of the origin, for each arc at the optimal solution reached using the Dantzig cutting plane algorithm. However, notice that the solution provided by the algorithm is not feasible and, therefore, is not a valid solution for the primal problem. The flows marked in red color clearly exceed the maximum capacities defined in section 2.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	10	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	10	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	10	.	.	.	35	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	5	.	.	.	10	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	5	.	.	.	.	.	.	.	.	30	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	5	.	.	.	.	.
8	.	.	.	.	.	.	.	.	15	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	5	.	.	.	5	.	10	.	5	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	0	.	.	.	0	.	.	.	10	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	20	.	.	.	30	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	0	.	0	.	0	.	25	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	0	.	5	0	.	0	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	5	.	.	.	.	20	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	25	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	25	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	5	.	20	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	10	.	.	.	5	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	10	0	10	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	5
22	.	.	.	.	.	.	.	.	.	0	.	.	.	.	.	0	.	.	.	0	.	.	0	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	5	.	0	.	0	.

We present now, as before, the flows segmented by origin.

Total flow from origin 3:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	10	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	10	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	10	.	.	.	35	5	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	0	.	.	.	10	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	5	.	.	.	.	.	.	.	.	30	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	5	.	.	.	.	.
8	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	0	.	.	.	5	.	0	.	5	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	0	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	0	.	0	.	0	.	0	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	0	.	0	0	.	0	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	5	.	.	.	.	20	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	0	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	0	.	5	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	10	.	.	.	5	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	0	0	5	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	0
22	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0	.

Total flow from origin 11:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	.	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
2	0	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
3	.	0	.	.	.	0	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
4	0	.	.	.	5	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
5	.	.	.	0	.	0	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.
6	.	.	0	.	0	.	.	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.
7	.	.	0	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.
8	.	.	.	.	.	.	.	.	15	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.
9	.	.	.	5	.	.	.	0	.	10	.	0	.	.	.	.	.	.	.	.	.	.	.	.
10	.	.	.	.	0	.	.	.	0	.	.	.	10	.	.	.	.	.	.	.	.	.	.	.
11	.	.	.	.	.	.	.	20	.	.	.	30	.	.	.	.	.	.	.	.	.	0	.	.
12	.	.	.	.	.	.	.	.	0	.	0	.	0	.	25	.	.	.	.	.	.	.	.	.
13	.	.	.	.	.	.	.	.	.	0	.	0	.	5	0	.	0	.	.	.	.	.	.	.
14	.	.	.	.	.	0	0	.	.	.	.	.	0	.	.	.	.	0	.	.	.	.	.	.
15	.	.	.	.	.	.	.	.	.	.	.	0	0	.	.	25	.	.	.	.	.	.	.	.
16	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	25	.	.	.	.	0	.	.
17	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	5	.	15	.	.	.	.
18	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	0	.	.	.	0	.	.	.
19	.	.	.	.	.	.	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0
20	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	.	.	10	0	5	.
21	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	.	.	5
22	.	.	.	.	.	.	.	.	.	.	0	.	.	.	.	0	.	.	.	0	.	.	0	.
23	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	0	.	0	.	0
24	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	5	.	0	.	0	.

6. *The evolution of the algorithm reporting for each iteration:*
   - *Iteration number*
   - *z_k, $w(\mu_k)$, $c^T x_{k+1}$, feasibility or unfeasibility of x_{k+1} (x_{k+1} <= d ¿)*

We present the results obtained by the algorithm at each iteration. Notice that the majority of the solutions obtained by the algorithm are not feasible and that the upper and lower bound eventually converge at the same optimal value obtained in section 3.

Iteration	z	w	c	Feasible
1	1,00E+12	4785,04	4785,04	No
2	5,71E+11	-4,29E+11	5022,87	No
3	5,71E+11	-4,29E+11	5060,05	No
4	4,44E+11	-5,56E+11	5665,04	No
5	2,22E+11	-7,78E+11	5945,31	Yes
6	5340,26	5173,5	5075,18	No
7	5286,08	5151,07	4940,18	No
8	5272,78	5182,81	4955,31	No
9	5249,54	5170,41	4904,92	No
10	5245,3	5220	5575,04	No
11	5234,46	5230,14	4924,92	No
12	5234,15	5234,15	5144,92	No

7. *Although unnecessary, solve the corresponding generalized linear programming problem equivalent to the dual of the linear problem solved in step 1 of the cutting plane algorithm reporting the solution obtained by this problem and the corresponding primal objective function value of the capacitated problem.*

To solve the generalized linear programming problem, we have to solve the following problem:

$$z_k = Min_{\alpha} \quad \sum_{j=0}^{k} \alpha_j f(\hat{x}_j)$$

$$\left.\begin{array}{l} \sum_{j=0}^{k} \alpha_j g(\hat{x}_j) \geq 0 \\ \sum_{j=0}^{k} \alpha_j h(\hat{x}_j) = 0 \\ \sum_{j=0}^{k} \alpha_j = 1 \ , \ \alpha_j \geq 0 \end{array}\right\} \longrightarrow \quad \widetilde{x} = \sum_{j=0}^{k} \alpha^* \hat{x}_j$$

It consists on minimize the alphas (as well as number of iterations), which are weights of the solutions at each iteration. It allows us to find an optimal feasible solution. In this assignment, we only have dualized the inequality constraint of capacities. So, we will not have equality constraints in this problem. We want to minimize the sum of the objective functions of each iteration, obtaining a weighted solution $\tilde{x}$. We write the mod file in AMPL:

```
var alpha {1..nCUT0} >=0;
var xfin {(i,j) in A} = sum {k in {1..nCUT0}} alpha[k]*xxX[i,j,k];
minimize glpp: sum {l in 1..nCUT0} alpha[l]*(sum {(i,j) in A} c[i,j]*xxX[i,j,l]);
subject to
        glpp_cap {(i,j) in A}: sum {l in 1..nCUT0}  alpha[l]*(y[i,j]-xxX[i,j,l]) >= 0;
```

```
subject to
        sum_alpha: sum {l in 1..nCUT0} alpha[l] = 1;
```

The optimal alpha, will be the dual variables from the Master Problem. Each iteration has provided a cut, so for each cut there is an optimal alpha associated:

Iteration	Alpha*		Cut	Dual Variable
1	0,000		1	0,000
2	0,000		2	0,000
3	0,000		3	0,000
4	0,000		4	0,000
5	0,000		5	0,000
6	0,129		6	0,129
7	0,000		7	0,000
8	0,371		8	0,371
9	0,000		9	0,000
10	0,429		10	0,429
11	0,071		11	0,071
12	0,000		12	0,000

As it can be observed in the previous results table, the dual variables have the same value as the optimal alpha.

If we compute $\tilde{x}$ for each arc, we observe that the new solution is feasible and still optimal because the value of z (exploitation costs) has remained the same.

From Node	To Node	xfin[i,j]	y[i,j]
3	6	20	20
6	14	15	15
11	12	25	25
12	13	15	15