42136 - Assignment 2

Spring 22, 42136
Assignment 2
May 11, 2022

Anton Ruby Larsen, s174356

Danmarks Tekniske Universitet

1 | Question 1

In this assignment we are given the company OptiGas who wants to expand into a new region in Denmark. They consider buying a already existing gas distributor in the region with the following network.

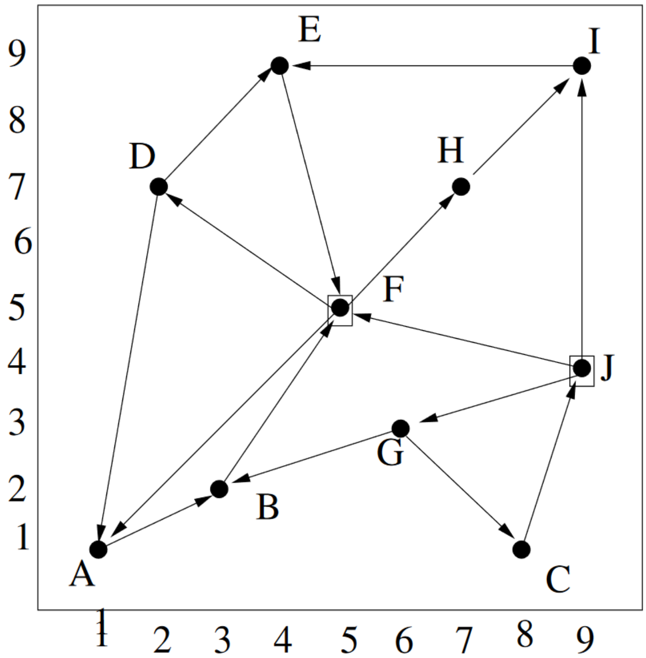


Figure 1.1 – The existing gas distributors pipe network

Gas is fed into the network through node F and J. The amount of gas pumped into the system in given in the column "Production" in table 1.1 while the demand is given in the column "Demand".

Node	Demand	Production
A	5	0
B	35	0
C	25	0
D	0	0
E	50	0
F	0	95
G	20	0
H	30	0
I	25	0
J	0	95

Table 1.1 – Demand and production of gas in the network.

The cost of transporting gas is given by the euclidean distance between two nodes and gas can only be sent in the direction of the edge. Hence can we for example only send gas from A to B but not B to A in the already existing network in 1.1. We can though build new pipes at a construction cost of 10 times the euclidean distance between two nodes. We will represent the transportation cost as

$$c_{ij} = \|z_i - z_j\|_2 \tag{1.0.1}$$

where z denotes the position of a node. Similarly we define the cost of building a new pipe as

$$f_{ij} = \begin{cases} 0 & \text{If the edge is already build} \\ 10c_{ij} & \text{Otherwise} \end{cases} \tag{1.0.2}$$

Set up of the problem

We are now to minimize cost where we assume the OptiGas buys the local distributor. We will define two decision variables. x_{ij} which decides how much gas to send from node i to node j at the cost of c_{ij} and y_{ij} decides if we build a pipe from node i to node j at the cost of f_{ij}. Hence we can now setup the objective function in the problem.

$$\min_{x,y} \quad \sum_i \sum_j c_{ij}x_{ij} + \sum_i \sum_j f_{ij}y_{ij} \tag{1.0.3}$$

where $x \in \mathbb{R}_+^{N\times N}$, $y \in \{0,1\}^{N\times N}$ and N denotes the number of nodes in the system. Then we have the constraints. First we must ensure that gas can only be sent through existing pipes. We hence introduce the constraint

$$My_{ij} - x_{ij} \geq 0, \quad \forall i, j, \tag{1.0.4}$$

where M is a large number. This is called a big-M formulation and if we had specific capacities of the pipes we could introduce that by M. Lastly we must make sure demand is met. This can be ensured by the constraint

$$\sum_j x_{ji} - \sum_j x_{ij} \geq D_i - P_i \quad \forall i. \tag{1.0.5}$$

Here $\sum_j x_{ji}$ is net inflow to node i and $\sum_j x_{ij}$ is net outflow of node i. This sum must be larger than the demand subtracted the production at all nodes.

We can hence now write up the full problem as

$$\begin{aligned} \min_{x,y} \quad & \sum_i \sum_j c_{ij}x_{ij} + \sum_i \sum_j f_{ij}y_{ij} \\ \text{s.t.} \quad & My_{ij} - x_{ij} \geq 0, \quad \forall i, j \\ & \sum_j x_{ji} - \sum_j x_{ij} \geq D_i - P_i \quad \forall i \\ & x \in \mathbb{R}_+^{N\times N}, \; y \in \{0,1\}^{N\times N} \end{aligned} \tag{1.0.6}$$

We are though to work with Benders decomposition for the rest of this assignment and hence it is much easier if we can get the problem into the form

$$\begin{aligned} \min_{x,y} \quad & c^T x + f^T y \\ \text{s.t.} \quad & Ax + By \geq b \\ & x \in \mathbb{R}_+^{N^2}, \; y \in \{0,1\}^{N^2} \end{aligned} \tag{1.0.7}$$

To do this we will vectorize the matrices c, f, x and y. We give an example of how we vectorize a 3×3 matrix.

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} \\ x_{21} & x_{22} & x_{23} \\ x_{31} & x_{32} & x_{33} \end{bmatrix} \xrightarrow{\text{vectorize}} \begin{bmatrix} x_{11} \\ x_{12} \\ x_{13} \\ x_{21} \\ x_{22} \\ x_{23} \\ x_{31} \\ x_{32} \\ x_{33} \end{bmatrix} \tag{1.0.8}$$

The B matrix only contains coefficients relevant for y in the constraints of 1.0.6. Hence it becomes

$$B = \begin{bmatrix} M \cdot I^{(N^2 \times N^2)} \\ \mathbf{0}^{(N \times N^2)} \end{bmatrix}, \tag{1.0.9}$$

where I is the identity matrix. Similarly the A matrix only contains coefficients relevant for x in the constraints of 1.0.6. The bottom constraint is a bit tricky so we examplify with a 3 by 3 example. The bottom constraint in 1.0.6 becomes

$$\begin{aligned} \sum_{j=1}^{3} x_{j1} - \sum_{j=1}^{3} x_{1j} &= x_{21} + x_{31} - x_{12} - x_{13} \\ \sum_{j=1}^{3} x_{j2} - \sum_{j=1}^{3} x_{2j} &= x_{12} + x_{32} - x_{21} - x_{23} \\ \sum_{j=1}^{3} x_{31} - \sum_{j=1}^{3} x_{3j} &= x_{13} + x_{23} - x_{31} - x_{32} \end{aligned} \tag{1.0.10}$$

which together with the first constraint in 1.0.6 for a 3 by 3 problem gives the A matrix

$$A^{(N=3)} = \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & -1 & -1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & -1 & -1 & 0 \end{bmatrix} \tag{1.0.11}$$

For $N = 10$ the same pattern holds. It is just enlarged to a $(N^2 + N) \times (N^2)$ matrix as the B matrix. Lastly the vector b becomes

$$b = \begin{bmatrix} \mathbf{0}^{(N^2 \times 1)} \\ D - P \end{bmatrix} \tag{1.0.12}$$

hence the dimension is $b \in \mathbb{R}^{N^2+N}$.

Solving the problem

We have setup the problem in the Julia file "Q1.jl". We obtain the objective value 805 and the following flow in the network.

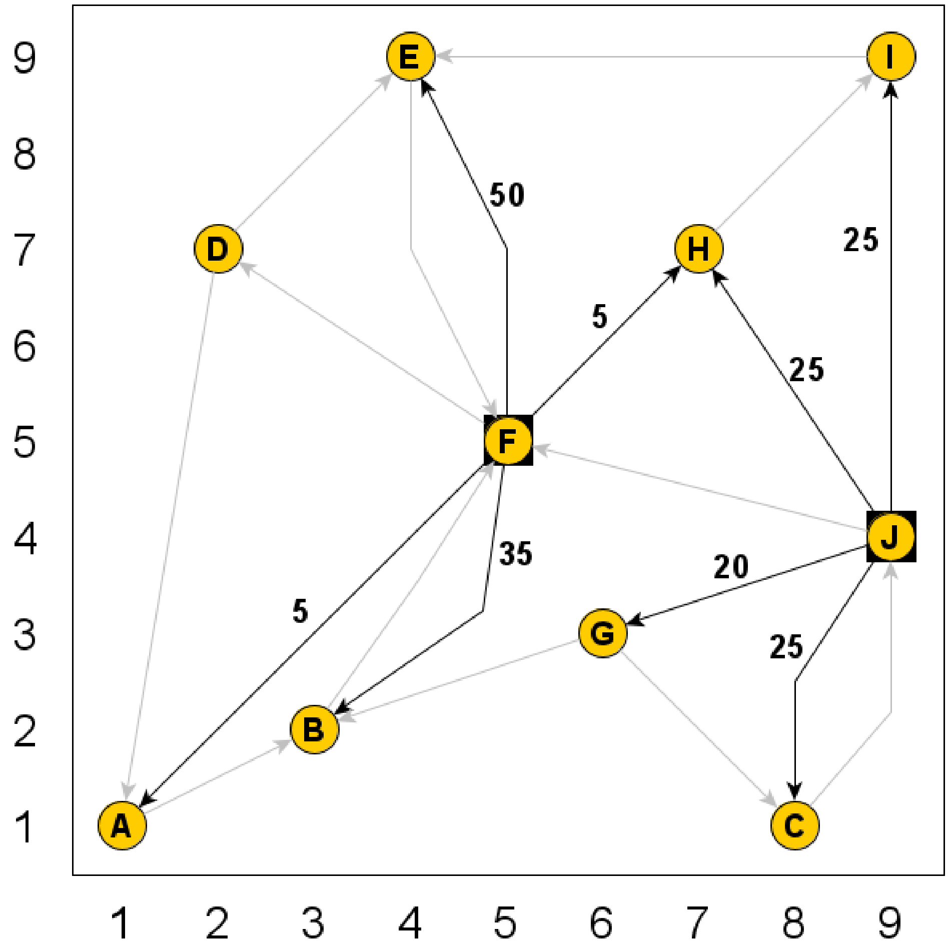


Figure 1.2

2 | Question 2

In the formulation 1.0.7 we see that the hard part of the problem is $y \in \{0, 1\}$. We hence try to fix y to a value $\bar{y}$ which gives the new problem

$$\begin{aligned} \min_{x} \quad & c^T x \\ \text{s.t.} \quad & Ax \geq b - B\bar{y} \\ & x \in \mathbb{R}_+^{N^2} \end{aligned} \tag{2.0.1}$$

We call this LP the primal Benders subproblem. We can now take the dual and obtain the dual Benders subproblem.

$$\begin{aligned} g(\bar{y}) = \max_{u} \quad & (b - B\bar{y})^T u \\ \text{s.t.} \quad & A^T u \leq c \\ & u \in \mathbb{R}_+^{(N^2+N)} \end{aligned} \tag{2.0.2}$$

In the rest of the assignment we will only work with the dual of the subproblem and hence just reference it as Benders subproblem or subproblem.

3 | Question 3

From [1] we know that by use of the Minkowski-Weyl theorem we can rewrite the dual subproblem using its extreme points. We can now use it to obtain the master problem from the original problem 1.0.7.

$$
\begin{aligned}
h(\{\bar{u}^p\},\{\bar{u}^r\}) = \min_{y,q} \quad & f^T y + q \\
\text{s.t.} \quad & (b - By)^T u^p \leq q, \quad \forall p \in \mathcal{I}_P \\
& (b - By)^T u^r \leq 0, \quad \forall r \in \mathcal{I}_R \\
& y \in \{0,1\}^{N^2}
\end{aligned}
\tag{3.0.1}
$$

where $q \in \mathbb{R}^1$, u^p are dual values for all extreme points $\mathcal{I}_P$ and u^r are dual values for all extreme rays $\mathcal{I}_R$. It is though impractical to generate all extreme points and rays. We will hence make a reduced version as in the Danzig Wolfe decomposition. We hence generate 1 at a time which corresponds to adding a constraint to the master problem. This corresponds to the delayed column generation in Danzig Wolfe decomposition and is also similarly called delayed constraint generation. Before we describe the delayed constraint generation we need to define a upper and lower bound for the procedure. The bounds are from [1].

$$
\mathrm{LB} = h(\{\bar{u}^p\},\{\bar{u}^r\}) \tag{3.0.2}
$$

$$
\mathrm{UB} = g(\bar{y}) + f^T \bar{y} \tag{3.0.3}
$$

where $h(\{\bar{u}^p\},\{\bar{u}^r\})$ is the objective value of the master problem and $g(\bar{y})$ is the objective value of the subproblem.
The procedure is now as follows

1. If the dual subproblem is unbounded we know the primal subproblem is infeasible. Hence we need to add a constraint to the master problem such it do not produce $\bar{y}$ solutions that are unbounded in the subproblem. Hence we have the extreme ray u^r and add the following constraint to the master problem.
$$(b - By)^T u^r \leq 0$$
This type of constraint is known as a Benders feasibility cut.

2. If $\mathrm{UB} - \mathrm{LB} \geq \varepsilon$ and the subproblem is optimal, then we have the extreme point u^p and add the following constraint to the master problem.
$$(b - By)^T u^r \leq q$$
This type of constraint is known as a Benders optimality cut.

3. If $\mathrm{UB} - \mathrm{LB} < \varepsilon$ then the algorithm stops and the found solution is optimal.

4 | Question 4

In our first implementation of the Benders decompoition we will only work with optimality cuts, and thus we must ensure our solver stays within the feasible domain. We therefore constraint our master problem with the initial network given in 1.1. This results in the new master problem 4.0.1.

$$
\begin{aligned}
h(\{\bar{u}^p\}, \{\bar{u}^r\}) = \min_{y,q} \quad & f^T y + q \\
\text{s.t.} \quad & (b - By)^T u^p \leq q, \quad \forall p \in \mathcal{I}_P \\
& y \geq y_{initial} \\
& y \in \{0,1\}^{N^2}
\end{aligned}
\tag{4.0.1}
$$

We have implemented the problem in the file "Q4.jl" where we initilize $\bar{y} = y_{initial}$. From table 6.1 we see that the solver uses 12 iterations to obtain the optimum and the optimal value is found to be 805. The same information is also given in the log file "Q4.log".

Iteration	Upper bound	Master objective	Subproblem objective
1.0	845.0	-4055.0	845.0
2.0	825.0	-1185.0	725.0
3.0	815.0	-175.0	725.0
4.0	815.0	-175.0	725.0
5.0	815.0	-145.0	745.0
6.0	805.0	755.0	675.0
7.0	805.0	775.0	825.0
8.0	805.0	785.0	795.0
9.0	805.0	785.0	775.0
10.0	805.0	785.0	775.0
11.0	805.0	795.0	795.0
12.0	805.0	805.0	775.0

Table 4.1 – Table showing the iterations of our solver only using optimality cuts and initialized with $\bar{y} = y_{initial}$.

We obtain the network flow given in figure 4.1 and the upper and lower bound is plotted in figure 4.2.

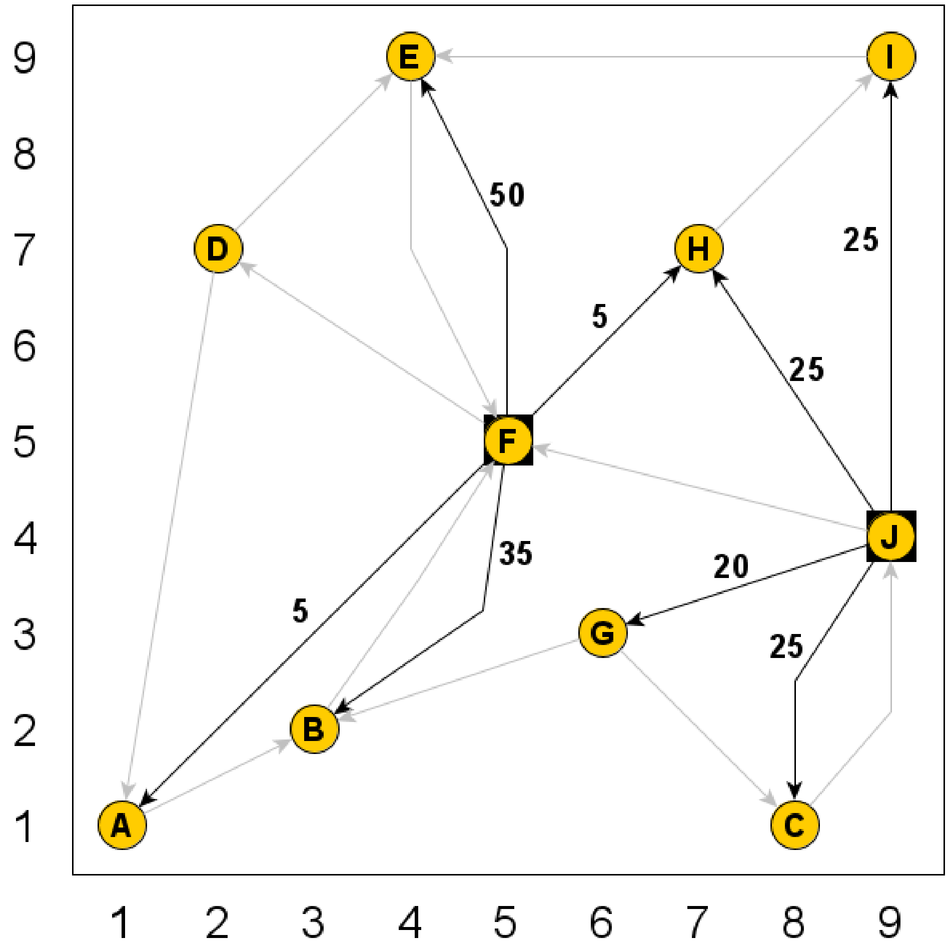


Figure 4.1

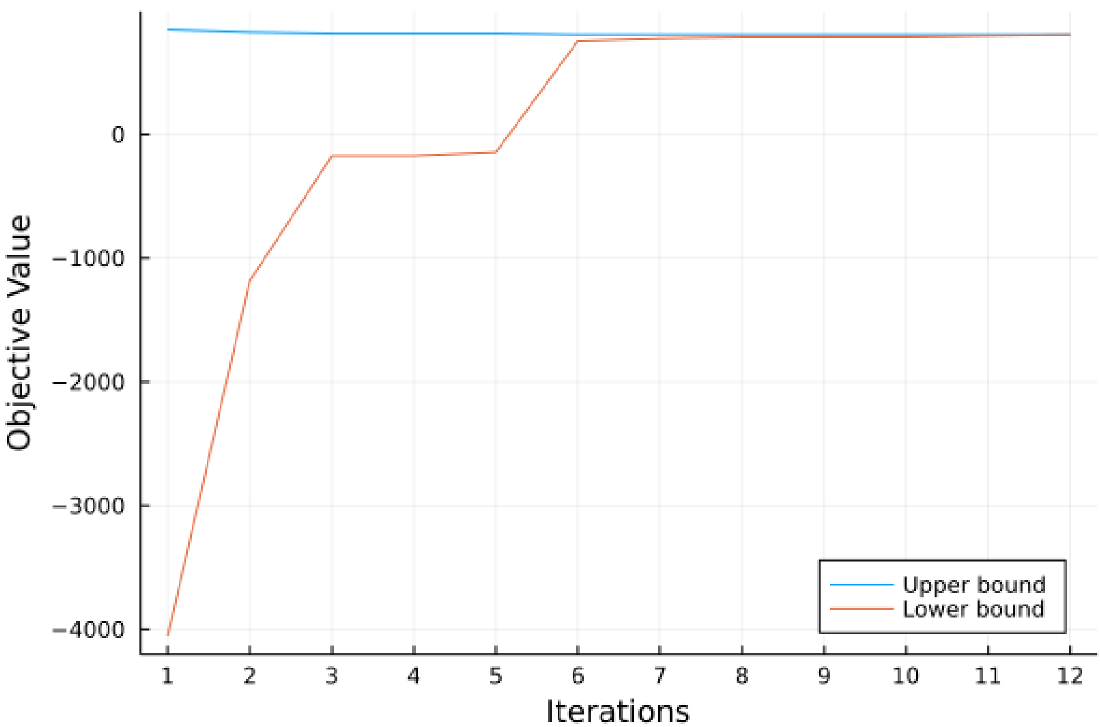


Figure 4.2

5 Question 5

The Benders problem described in Question 2 and 3 is already with rays so we will only reference these Questions here. We though still need to touch upon one thing before we can implement the ray version of the problem in Julia and that is the ray problem. When we have found that the dual of the Benders subproblem is unbounded then we know an extreme ray exists but we also need to find it. To do this we will use the Ray Problem given on page 10 in [2] and here given in 5.0.1.

$$\begin{aligned} \max_{u} \quad & 1 \\ \text{s.t.} \quad & (b - B\bar{y})^T u = 1 \\ & A^T u \leq 0 \\ & u \in \mathbb{R}_+^{(N^2+N)} \end{aligned} \tag{5.0.1}$$

6 | Question 6

We can now stitch exercise 2, 3 and 5 together and obtain the Julia program "Q6.jl". Here we initialize the problem with a vector of zeros to investigate if OptiGas should just build its own network instead of buying the already existing one. The program found a solution after 58 iterations and hence a table with all iterations as in exercise 4 would take up to much space. We will therefore only show a summary in table 6.1 and if one wants to see the full information on each iteration we refer to the log file "Q6.log".

Iterations	Found solution	Optimality cuts	Feasibility cuts
58	895	25	33

Table 6.1 – Table showing summary statistics for the ray version of Benders decomposition.

We hence see that the cost of building a network up from scratch is a more expensive solution than just buying the existing network. We obtain the network flow given in figure 6.1 and the upper and lower bound it plotted in figure 6.2.

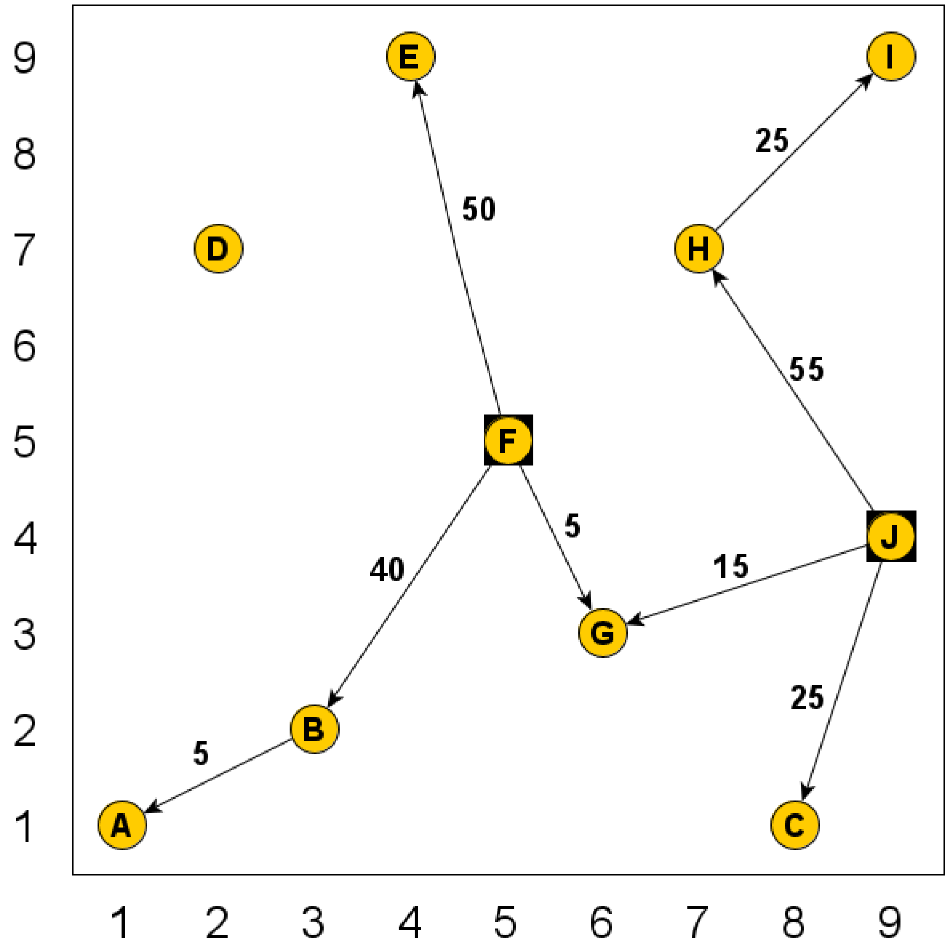


Figure 6.1

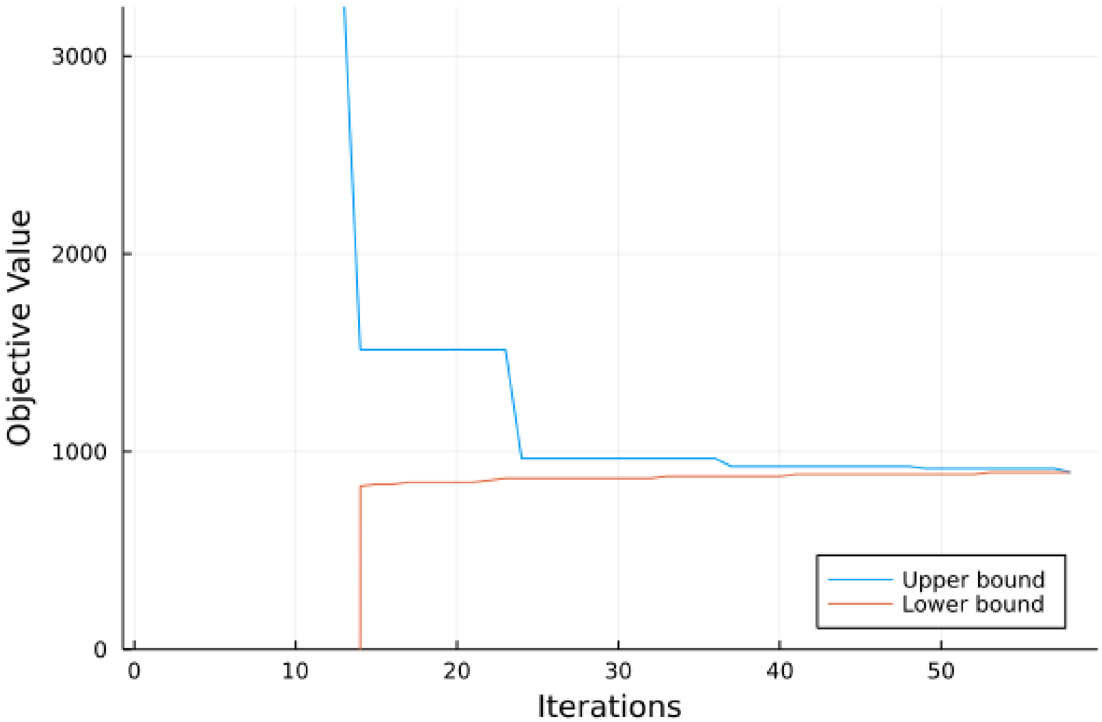


Figure 6.2

Using lazy constraints

In the above we have used a classic loop based approach where we resolve every time we add a new constraint to the master problem. A more modern approach adds the constraints dynamically to the master problem. We can add the constraints dynamically by use of callback functions implemented in the chosen solve, e.g. GUROBI. We have implemented this is the file "Q6_callback.jl". A short summary is presented in table 6.2 and in figure 6.3 we have plotted the upper and lower bound for the nodes explored in the callback optimization. A more extensive summary can by found in "Q6_callback.log".

Explored nodes	Found solution
324	895

Table 6.2 – Table showing summary statistics for the ray version of Benders decomposition using callback.

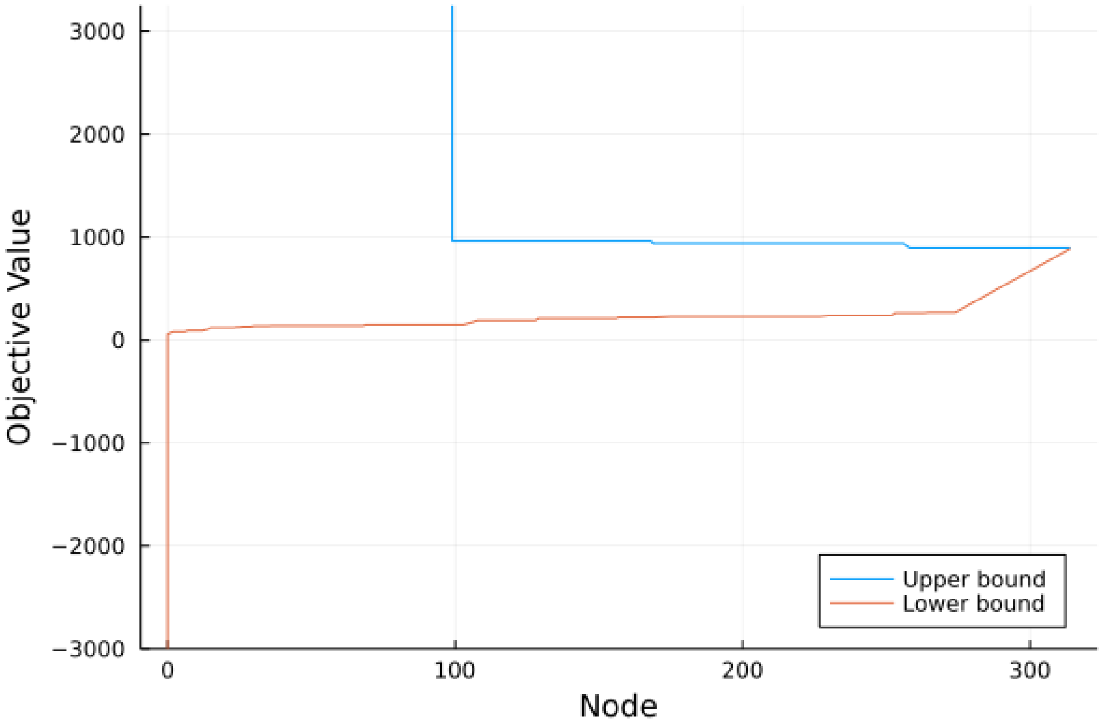


Figure 6.3

7 Question 7

We now introduce scenarios for the demand.

$$D_{\text{Scenarios}} = \begin{array}{|c|cccccccccc|}\hline & \text{A} & \text{B} & \text{C} & \text{D} & \text{E} & \text{F} & \text{G} & \text{H} & \text{I} & \text{J} \\ \hline 1 & 5 & 35 & 25 & 0 & 50 & 0 & 20 & 30 & 25 & 0 \\ 2 & 5 & 35 & 25 & 10 & 30 & 0 & 20 & 30 & 35 & 0 \\ 3 & 25 & 45 & 25 & 0 & 50 & 0 & 10 & 20 & 15 & 0 \\ 4 & 5 & 25 & 35 & 0 & 50 & 0 & 20 & 30 & 25 & 0 \\ 5 & 5 & 35 & 35 & 0 & 50 & 0 & 10 & 20 & 35 & 0 \\ 6 & 5 & 35 & 25 & 0 & 60 & 0 & 20 & 20 & 25 & 0 \\ 7 & 5 & 25 & 15 & 0 & 60 & 0 & 20 & 40 & 25 & 0 \\ 8 & 5 & 25 & 25 & 0 & 60 & 0 & 30 & 30 & 15 & 0 \\ 9 & 5 & 35 & 25 & 10 & 30 & 0 & 20 & 30 & 35 & 0 \\ 10 & 5 & 35 & 15 & 0 & 50 & 0 & 20 & 50 & 15 & 0 \\ \hline \end{array} \tag{7.0.1}$$

where all rows correspond to a scenario ω_k which all have probability 1/10. This can be formulated as a two stage program

$$\begin{aligned} \min_{y} \quad & f^T y + \mathbb{E}\left[Q(y,\omega)\right] \\ \text{s.t.} \quad & y \in \{0,1\}^{N^2} \\ \text{where} \quad Q(y,\omega) = \min_{x} \quad & c^T x(\omega) \\ \text{s.t.} \quad & Ax(\omega) + By \geq b_\omega \quad \forall \omega \in \Omega \\ & x(\omega) \in \mathbb{R}_+^{N^2} \end{aligned} \tag{7.0.2}$$

where Ω is the set of the 10 scenarios given in 7.0.1. We can rewrite 7.0.2 to the deterministic equivalent as

$$\begin{aligned} \min_{x_{DE},y} \quad & c_{DE}^T x_{DE} + f^T y \\ \text{s.t.} \quad & A_{DE} x_{DE} + B_{DE} y \geq b_{DE} \\ & x_{DE} \in \mathbb{R}_+^{SN^2}, \; y \in \{0,1\}^{N^2} \end{aligned} \tag{7.0.3}$$

where S denotes the number of scenarios and the subscript $_{DE}$ indicates a modification of the vector or matrix compared to the non stochastic version, 1.0.7. We will hence go through these modifications.

c_{DE} and x_{DE} are just 10 times the original stacked on top of each others.

$$c_{DE} = \frac{1}{10}\begin{bmatrix} c \\ \vdots \\ c \end{bmatrix}, \; c_{DE} \in \mathbb{R}^{SN^2} \tag{7.0.4}$$

$$x_{DE} = \begin{bmatrix} x \\ \vdots \\ x \end{bmatrix}, \; x_{DE} \in \mathbb{R}^{SN^2} \tag{7.0.5}$$

$$\tag{7.0.6}$$

The matrix B_{DE} is also a stacked version of the original given as

$$B_{DE} = \begin{bmatrix} MI^{(N^2 \times N^2)} \\ \vdots \\ MI^{(N^2 \times N^2)} \\ \mathbf{0}^{(N \times N)} \\ \vdots \\ \mathbf{0}^{(N \times N)} \end{bmatrix}, \ B_{DE} \in \mathbb{R}^{(S(N+N^2) \times N^2)} \tag{7.0.7}$$

where each element in the matrix is repeated S times. The matrix A_{DE} is a bit more complicated than B_{DE} because it needs to take care of each x in the 10 scenarios. We will in the following denote the bottom of the original matrix A, as A_{Bot}, i.e. for the 3 by 3 example A_{Bot} becomes

$$A_{Bot}^{(N=3)} = \begin{bmatrix} 0 & -1 & -1 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & -1 & -1 & 0 \end{bmatrix} \tag{7.0.8}$$

The matrix A_{DE} is then given as

$$A_{DE} = \begin{bmatrix} -I^{(SN^2 \times SN^2)} \\ I(A_{Bot}^{(N)})^{(S \times S)} \end{bmatrix}, \quad A_{DE} \in \mathbb{R}^{(S(N+N^2) \times N^2)} \tag{7.0.9}$$

where $I(A_{Bot}^{(N)})^{(S \times S)}$ denotes a block diagonal with S $A_{Bot}^{(N)}$ in the diagonal. Lastly we have b_{DE} which is given by

$$b_{DE} = \begin{bmatrix} \mathbf{0}^{(SN^2 \times 1)} \\ D_1 - P \\ \vdots \\ D_S - P \end{bmatrix}, \quad b_{DE} \in \mathbb{R}^{S(N+N^2)} \tag{7.0.10}$$

Solving the problem

We can now solve 7.0.3 in Julia where we have named the file "Q7.jl". We have assumed that OptiGas is building a network up from scratch and obtain a optimal cost of 940. The associated flow through the network is given in figure 7.1.

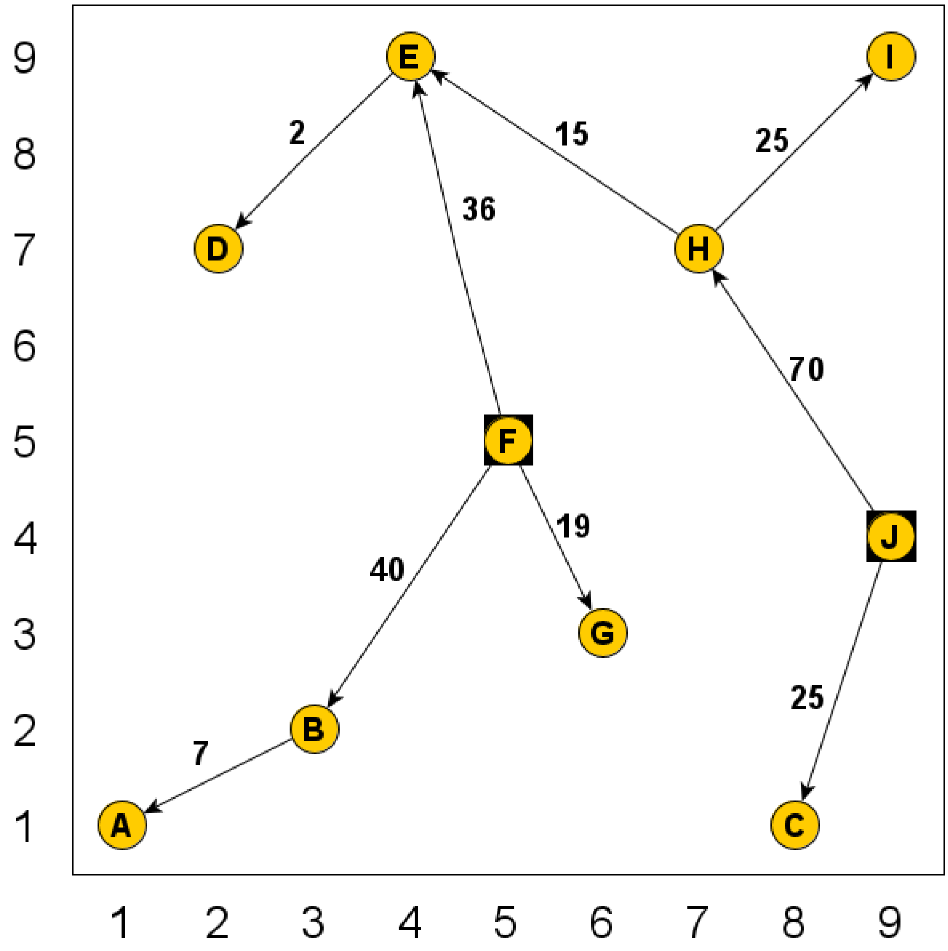


Figure 7.1

8 Question 8

We can now use exactly the same frame work as in Question 2 to setup the subproblem for the deterministic equivalent for 7.0.2. Hence the dual subproblem becomes

$$\begin{aligned} g(\bar{y}) = \max_{u_{DE}} \quad & (b_{DE} - B_{DE}\bar{y})^T u_{DE} \\ \text{s.t.} \quad & A_{DE}^T u_{DE} \leq c_{DE} \\ & u \in \mathbb{R}_+^{S*(N^2+N)} \end{aligned} \tag{8.0.1}$$

This is just solving everything in one huge subproblem but we can also use the uncoupled structure of the subproblems. By doing this we can solve the problems in parallel and hence obtain 10 of the following subproblems.

$$\begin{aligned} g(\bar{y})_\omega = \max_{u_\omega} \quad & (b_\omega - B\bar{y})^T u_\omega \\ \text{s.t.} \quad & A^T u_\omega \leq c \\ & u_\omega \in \mathbb{R}_+^{(N^2+N)} \end{aligned} \tag{8.0.2}$$

In the above program the matrices without subscript are the same matrices as in the rest of the assignment and the two elements with subscript $\_\omega$ are as follows.

$$b_{\omega=k} = \begin{bmatrix} \mathbf{0}^{(N^2\times 1)} \\ D_k - P \end{bmatrix}, \quad b_{DE} \in \mathbb{R}^{(N+N^2)} \tag{8.0.3}$$

$$u_{\omega=k} \in \mathbb{R}^{(N+N^2)} \tag{8.0.4}$$

Ray problems

If we do not utilize the uncoupled structure of the subproblems it is quite straight forward and we just copy the workflow and structure from the rest of the assignment. The ray problem then becomes

$$\begin{aligned} \max_{u} \quad & 1 \\ \text{s.t.} \quad & (b_{DE} - B_{DE}\bar{y})^T u_{DE} = 1 \\ & A_{DE}^T u_{DE} \leq 0 \\ & u_{DE} \in \mathbb{R}_+^{S(N^2+N)} \end{aligned} \tag{8.0.5}$$

If we though want to utilize the uncoupled structure we must solve for a feasibility cut every time any of the the 10 subproblems are unbounded. Then we set all 10 u_ω to zero and only change those of them which were unbounded. For example if only subproblem 4 was unbounded all the other u_ω are set to 0 and only $u_{\omega=4}$ is solved for in the ray problem. The ray problem for the uncoupled structure becomes

$$\begin{aligned} \max_{u_\omega} \quad & 1 \\ \text{s.t.} \quad & (b_\omega - B\bar{y})^T u_\omega = 1 \\ & A^T u_\omega \leq 0 \\ & u_\omega \in \mathbb{R}_+^{(N^2+N)} \end{aligned} \tag{8.0.6}$$

9 | Question 9

Similarly as for the previous question we can use the same framework as in Question 3 to setup the master problem for the deterministic equivalent for 7.0.2. Hence the master problem becomes

$$
\begin{aligned}
h(\{\bar{u_{DE}}^p\}, \{\bar{u_{DE}}^r\}) = \min_{y,q} \quad & f^T y + q \\
\text{s.t.} \quad & (b_{DE} - B_{DE} y)^T u_{DE}^p \leq q, \quad \forall p \in \mathcal{I}_P \\
& (b_{DE} - B_{DE} y)^T u_{DE}^r \leq 0, \quad \forall r \in \mathcal{I}_R \\
& y \in \{0,1\}^{N^2}
\end{aligned}
\tag{9.0.1}
$$

This is again not utilizing the uncoupled structure of the subproblems. If we were to utilize the uncoupled structure, then the master problem would become

$$
\begin{aligned}
\min_{y,q} \quad & f^T y + q \\
\text{s.t.} \quad & \sum_{\omega=1}^{S} (b_\omega - By)^T u_\omega^p \leq q, \quad \forall p \in \mathcal{I}_P \\
& \sum_{\omega=1}^{S} (b_\omega - By)^T u_\omega^r \leq 0, \quad \forall r \in \mathcal{I}_R \\
& y \in \{0,1\}^{N^2}
\end{aligned}
\tag{9.0.2}
$$

10 Question 10

We are now to solve 7.0.3 using Benders decomposition initializing the problem with a vector of zeros. We first solve the problem without using the uncoupled structure. The solution is found after 71 iterations and hence we will again only show a summary table here and refer to the log file "Q10.log" for information regarding every iteration.

Iterations	Found solution	Optimality cuts	Feasibility cuts
71	940	43	28

Table 10.1 – Table showing summary statistics for the deterministic equivalent not using the uncoupled structure.

We see that the cost found is the same as in Question 7 and the obtained network is shown in figure 10.1. The code for the solution can be found in "Q10.jl" and the upper and lower bound it plotted in figure 10.2.

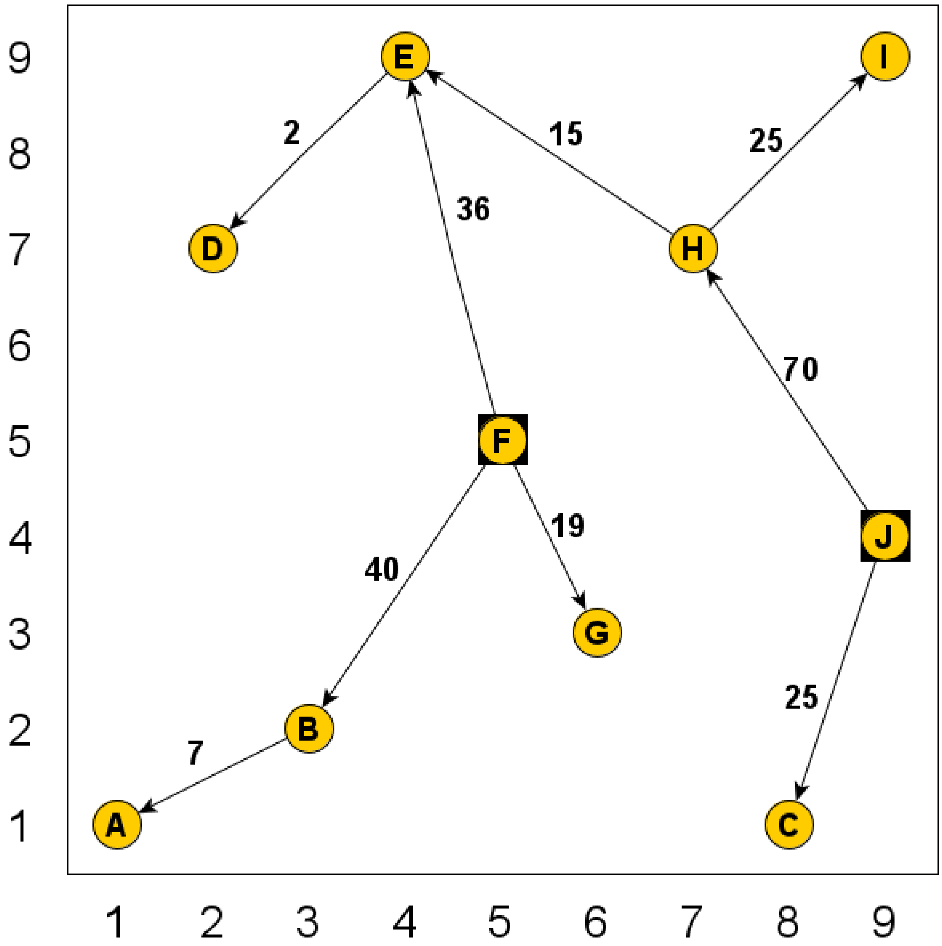


Figure 10.1

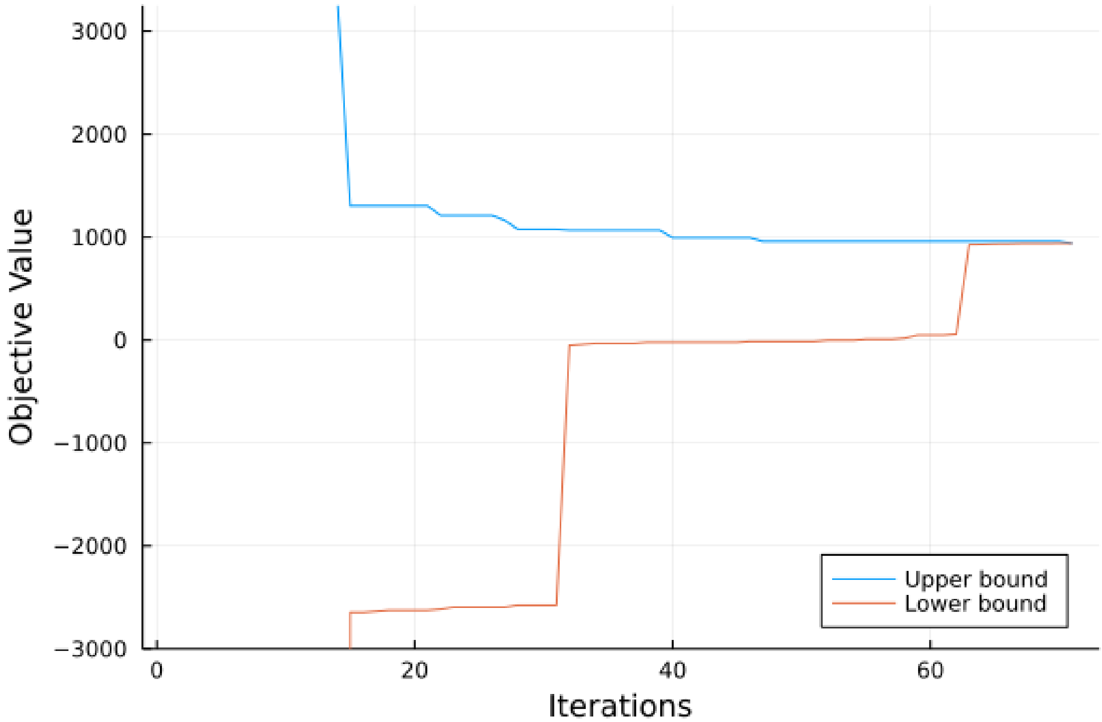


Figure 10.2

Using the uncoupled structure

If we make use of the uncoupled structure we use 98 iterations to obtain the solution but here every subproblem can be parallelized making each iteration much faster. The summary is given in table 10.2 and the full information regarding each iteration can be found in the log file "Q10_parallel.log".

Iterations	Found solution	Optimality cuts	Feasibility cuts
98	940	49	49

Table 10.2 – Table showing summary statistics for the deterministic equivalent using the uncoupled structure.

The code for the solution can be found in "Q10_parallel.jl" and the upper and lower bound is plotted in figure 10.3.

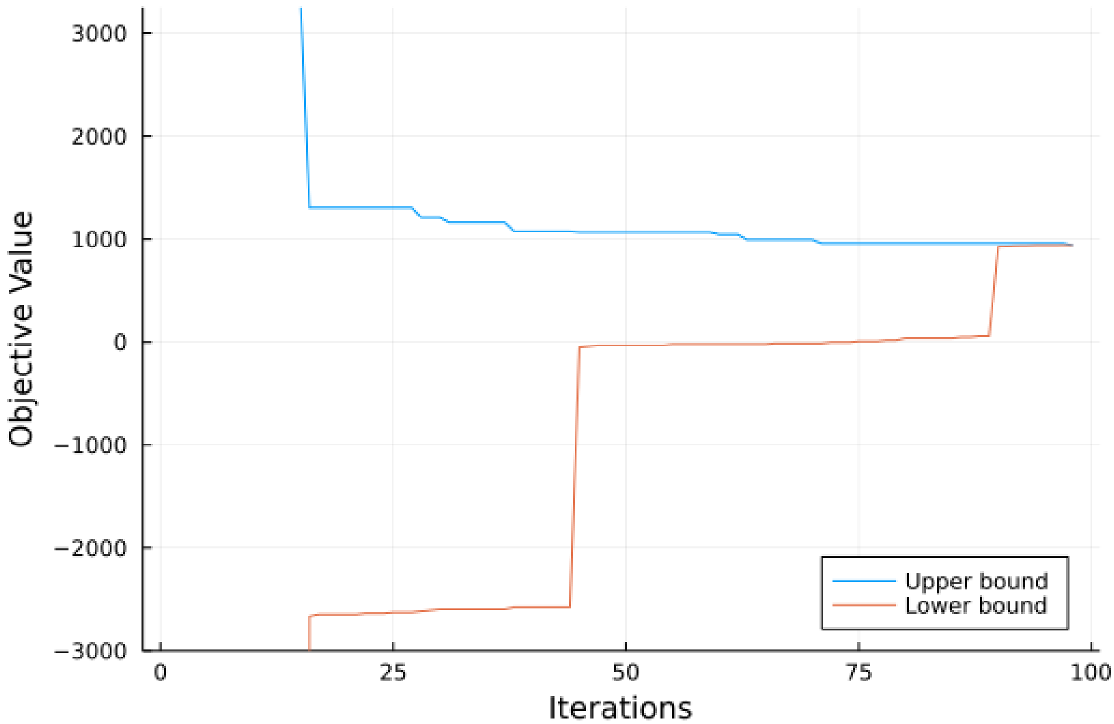


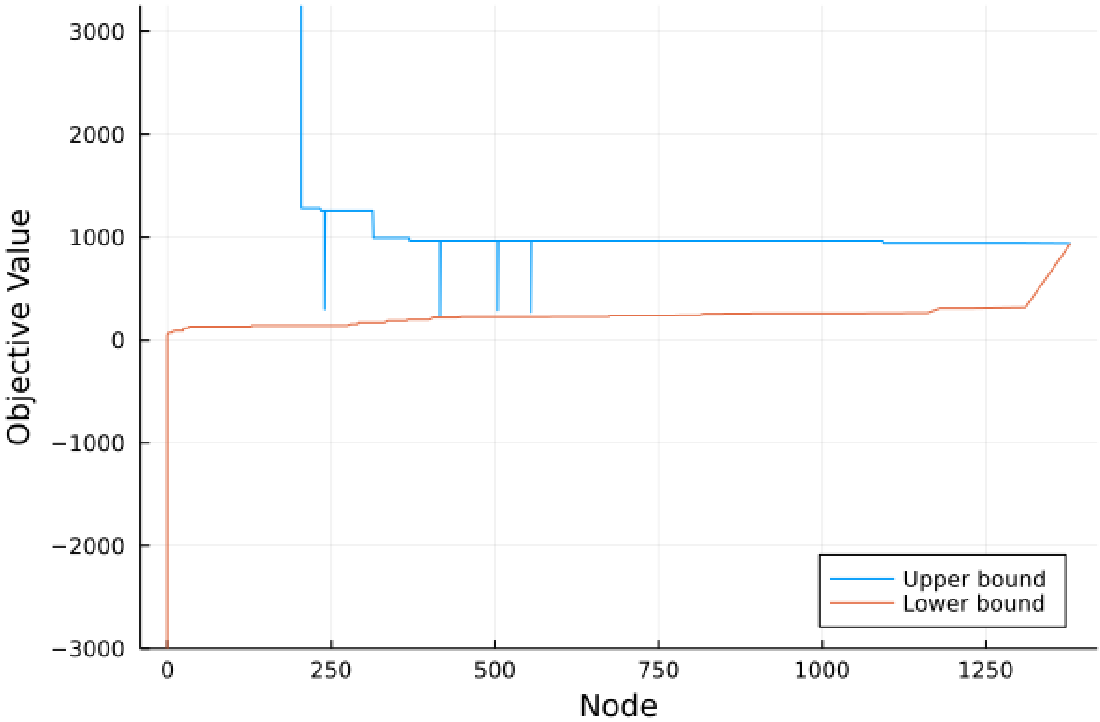
Figure 10.3

Using lazy constraints

As in Question 6 we have also implemented a callback version here. The implementation can be found in "Q10_callback.jl" and a short summary is here given in 10.3. Further we have in figure 10.4 plotted the upper and lower bound for the nodes explored in the callback optimization. A more extensive summary can by found in "Q10_callback.log".

Explored nodes	Found solution
1380	940

Table 10.3 – Table showing summary statistics for the deterministic equivalent using callback.



Objective Value
3000
2000
1000
0
−1000
−2000
−3000
0
250
500
750
1000
1250
Node
Upper bound
Lower bound

Figure 10.4

11 Question 11

In this exercise we are to use Danzig Wolfe on the original problem without scenarios. We recall Danzig Wolfe decomposition to work on problems of the form

$$\begin{aligned} \max_{x} \quad & cx \\ \text{s.t.} \quad & A^0 x \leq b^0 \\ & A^1 x \leq b^1 \\ & x_1, \dots x_p \in \mathbb{Z}^p, \quad x_{p+1}, \dots x_n \mathbb{R}^{n-p} \end{aligned} \tag{11.0.1}$$

We now chose to convexify the constraints $A^1 x \leq b^1$ by use of Minkowski-Weyl and obtain the following new form

$$\begin{aligned} \max_{x} \quad & cx \\ \text{s.t.} \quad & A^0 x \leq b^0 \\ & x = \bar{X}^1 \lambda \\ & \mathbf{1}\lambda = 1 \\ & \lambda \geq 0 \\ & x_1, \dots x_p \in \mathbb{Z}^p, \quad x_{p+1}, \dots x_n \mathbb{R}^{n-p} \end{aligned} \tag{11.0.2}$$

where $\bar{X}^1$ is the set of extreme points defining the set $A^1 x \leq b^1$. The set $\bar{X}^1$ can be exponentially large and hence infeasible to work with the complete set. We hence recall the procedure column generation which chose only relevant variables from $\bar{X}^1$ similarly to the delayed constraint generation we have used in Benders.
To identify how we should split up the constraints in our problem, 1.0.7, to fit the structure in Danzig Wolfe we restate it here.

$$\begin{aligned} \min_{x,y} \quad & \sum_i \sum_j c_{ij} x_{ij} + \sum_i \sum_j f_{ij} y_{ij} \\ \text{s.t.} \quad & M y_{ij} - x_{ij} \geq 0, \quad \forall i, j, \\ & \sum_j x_{ji} - \sum_j x_{ij} \geq D_i - P_i \quad \forall i, j. \\ & x \in \mathbb{R}_+^{N \times N}, \ y \in \{0,1\}^{N \times N} \end{aligned} \tag{11.0.3}$$

We see that the first constraint is the only one including integer variables. Hence we will use this constraint as the one we will convexify but before we do so we will convert the problem into a matrix form.

$$\begin{aligned}
\max_{x,y} \quad & -\begin{bmatrix} c \\ f \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} \\
\text{s.t.} \quad & \begin{bmatrix} A_1 & B_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \geq 0 \\
& \begin{bmatrix} A_2 & B_2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \geq D - P \\
& x \in \mathbb{R}_+^{N^2},\ y \in \{0,1\}^{N^2}
\end{aligned} \tag{11.0.4}$$

where $\begin{bmatrix} A_1 \\ B_1 \end{bmatrix}$ is the top part of the A and B matrix in Question 1 which only consisted of a diagonal. The matrix $\begin{bmatrix} A_2 \\ B_2 \end{bmatrix}$ is the bottom part of the same two matrices. This gives the following master problem in the column generation

$$\begin{aligned}
\max_{x,y} \quad & -\begin{bmatrix} c \\ f \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} \\
\text{s.t.} \quad & (\begin{bmatrix} A_2 & B_2 \end{bmatrix} \bar{X}_k^1)\lambda \geq D - P \\
& \mathbf{1}\lambda \geq 1 \\
& \lambda \geq 0
\end{aligned} \tag{11.0.5}$$

and the subproblem becomes

$$\begin{aligned}
\max_{x,y} \quad & \begin{bmatrix} c \\ f \end{bmatrix}^T \begin{bmatrix} x \\ y \end{bmatrix} - \pi \begin{bmatrix} A_2 & B_2 \end{bmatrix} - \kappa \\
\text{s.t.} \quad & \begin{bmatrix} A_1 & B_1 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \geq 0 \\
& x \in \mathbb{R}_+^{N^2},\ y \in \{0,1\}^{N^2}
\end{aligned} \tag{11.0.6}$$

We can now use branch-and-price to obtain the optimal integer solution. We have implemented the branch-and-price in the file "Q11_DW.jl". We initialize the program with the given network and should hence obtain the same solution as in Question 4. We obtain a objective value of 805.0 which is the same as in Question 4.

Bibliography

[1] İlker Birbil. Benders decomposition and delayed constraint generation. [Online]. Available: https://personal.eur.nl/birbil/ie606/05_Benders_Dec/05_Benders_Dec_Octave.html

[2] *Benders Algorithm Cook Book, With and Without Rays.* DTU.