COMILLAS UNIVERSIDAD PONTIFICIA
ICAI ICADE CIHS

comillas.edu

comillas.edu

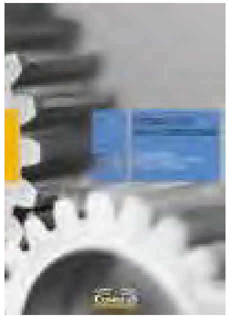
Stochastic Optimization

Prof. Andres Ramos
https://www.iit.comillas.edu/aramos/
Andres.Ramos@comillas.edu
arght@mit.edu

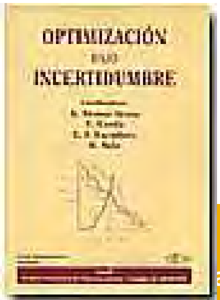
January 2023

References

- Birge, J.R. and Louveaux, F. (2011) *Introduction to Stochastic Programming.* Springer-Verlag.
- Ben-Tal, A., El Ghaoui, L. and Nemirovski, A. *Robust Optimization.* Princeton University Press, 2009
- G.B. Dantzig, M.N. Thapa *Linear Programming. 2: Theory and Extensions*. Springer 2003
- Ermoliev, Y. and Wets, R.J-B. (eds.) *Numerical Techniques for Stochastic Optimization*. Springer-Verlag. Berlin, Germany. 1988.
- Kall, P. and Wallace, S.W. (1995) *Stochastic Programming*. John Wiley and Sons.
- King, A.J. and Wallace, S.W. *Modeling with Stochastic Programming*, Springer Series in Operations Research and Financial Engineering, Springer, New York, 2012
- Infanger, G. *Planning Under Uncertainty: Solving Large-Scale Stochastic Linear Programs* Boyd & Fraser. 1994.
- Mayer, J. (1998) *Stochastic Linear Programming Algorithms. A Comparison Based on a Model Management System.* Gordon and Breach Science Publishers.
- Prekopa, A. (1995) *Stochastic Programming* Kluwer Academic Publishers.
- Ruszczynski, A. and Shapiro, A. (eds.) (2003) *Stochastic Programming* Elsevier.



Ramos, A., A. Alonso-Ayuso, G. Pérez (eds.) (2008) *Optimización bajo Incertidumbre* Universidad Pontificia Comillas. Madrid, España



Alonso-Ayuso, A., E. Cerdá, L. F. Escudero, R. Sala (eds.) (2004) *Optimización bajo incertidumbre* Tirant lo Blanch. Valencia, España.

Other resources

- Stochastic Programming Society (https://www.stoprog.org/)
- Stochastic Programming Resources (www, papers, tutorials, lecture notes, books, software) (https://www.stoprog.org/resources)
- Stochastic Programming Bibliography (http://www.eco.rug.nl/mally/spbib.html)
- Stochastic Programming E-Print Series (http://www.speps.org/)
- Optimization Online. Stochastic Programming submissions (http://www.optimization-online.org/ARCHIVE_CAT/STOCH/index.html)

- Red Temática de Optimización bajo Incertidumbre (ReTOBI)

- International Conference in Stochastic Programming (ICSP)

Contents

1

General overview

Stochasticity or uncertainty

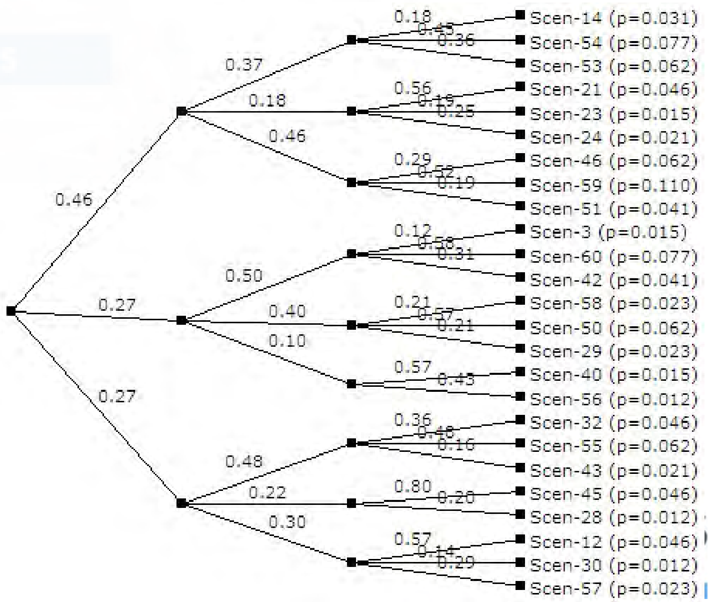
- Origin
  - Future information (fuel prices or future demand)
  - Lack of reliable data
  - Measurement errors
- In electric energy systems planning
  - Demand (seasonal/daily variation, yearly, increment over time)
  - Wind power
  - Hydro inflows
  - Availability of generation and network elements
  - Electricity or fuel prices
- Uncertainty relevant for each time scale

Decision under uncertainty

- DETERMINISTIC optimization
  - Best decision when future is known

- Simulation. Scenario analysis
  - What could happen if …?

- STOCHASTIC optimization
  - Best decision when future is uncertain but with a known probability



Deterministic vs. Stochastic Optimization

- Deterministic
  - Parameters known with certainty (it can be the mean value)
- Stochastic
  - Parameters modeled as stochastic variables with known distributions
    - Discrete
    - Historical
    - Continuous $\Rightarrow$ simulation

Example: Newsvendor

- It is a mathematical model in operations management and applied economics used to determine optimal inventory levels. It is (typically) characterized by fixed prices and uncertain demand. If the inventory level is q, each unit of demand above q is lost. This model is also known as the *Newsvendor Problem* or *Newsboy Problem* by analogy with the situation faced by a newspaper vendor who must decide how many copies of the day's paper to stock in the face of uncertain demand and knowing that unsold copies will be worthless at the end of the day.

Source: Wikipedia

http://www.siemens.com/

Example: Generation expansion planning (GEP)

- The following two-stage problem consists of determining the optimal capacity investment in various types of power plants to meet next period demands for electricity. Four power plants are considered, and they can operate in three different modes. The next period demand for each of the three modes are to be met. There is a budget constraint and a constraint on the minimum total capacity.

- Stages
  - First stage: investment decisions
  - Second stage: operation decisions

Example: Generation expansion planning (GEP)

- Time scope divided into three periods
- 3 Stochastic demand scenarios with change only in the first period
- Several type of generators available
- Expansion decisions must be unique for all the scenarios
- Variables
  - Generation expansion
  - Unit output
- Constraints
  - Maximum budget available
  - Minimum capacity to be installed
  - Load-generation balance for each scenario
  - Power output lower than the installed capacity

Generation expansion planning model (i)

```
$Title Generation Expansion Planning Model

sets
   I generators         /  gen-1 *   gen-4 /
   J periods            /  per-1 *   per-3 /
   S demand scenarios  / scen-1 * scen-3 /
```

Generation expansion planning model (ii)

```
parameters
    F(i)     fixed investment cost [€ per MW]
                                                    / gen-1  10
                                                      gen-2   7
                                                      gen-3  16
                                                      gen-4   6 /
    PROB(s) scenario probability [p.u.]
                                                    / scen-1 0.2
                                                      scen-2 0.5
                                                      scen-3 0.3 /
    DEM(j)   scenario load              [MW]
    CAPMIN   minimum capacity to install [MW] /  12 /
    BUDGLM   budget limit               [€]  / 120 /

table V(i,j) variable operation cost [€ per MW]
        per-1  per-2  per-3
gen-1    40     24     4
gen-2    45     27     4.5
gen-3    32     19.2   3.2
gen-4    55     33     5.5

table DEMS(s,j) stochastic demand [MW]
        per-1 per-2 per-3
scen-1    3     3     2
scen-2    5     3     2
scen-3    7     3     2
```

Generation expansion planning model (iii)

```
variables
   X(     i)             installed capacity [MW]
   Y(   j,i)             operation output   [MW]
   YS(s,j,i) stochastic operation output   [MW]
   TCOST     total cost                    [€]
positive variables X, Y, YS
```

Generation expansion planning model (iv)

```
equations
   COST                 total cost                         [€]
   COSTS    stochastic total cost                         [€]
   BUDGET   budget limit                                   [€]
   MININST  minimum capacity to install                    [MW]
   OUTPIN   power output < installed                       [MW]
   OUTPINS  power output < installed stochastic            [MW]
   LOADBAL              load balance                       [MW]
   LOADBALS stochastic load balance                        [MW] ;

COST   .. TCOST =e= sum[     i ,               F(i  ) * X(     i)]
                  + sum[(  j,i),               V(i,j) * Y(   j,i)] ;
COSTS .. TCOST =e= sum[      i ,               F(i  ) * X(     i)]
                  + sum[(s,j,i), PROB(s) * V(i,j) * YS(s,j,i)] ;

BUDGET  .. sum[i, F(i) * X(i)] =l= BUDGLM ;
MININST .. sum[i,        X(i)] =g= CAPMIN ;

OUTPIN (  j,i) .. Y(   j,i) =l= X(i) ;
OUTPINS(s,j,i) .. YS(s,j,i) =l= X(i) ;

LOADBAL (  j) .. sum[i, Y (  j,i)] =g= DEM (  j) ;
LOADBALS(s,j) .. sum[i, YS(s,j,i)] =g= DEMS(s,j) ;
```

Generation expansion planning model (v)

```
model DETERMINISTIC / COST , MININST, BUDGET, OUTPIN , LOADBAL  /
model STOCHASTIC    / COSTS, MININST, BUDGET, OUTPINS, LOADBALS /

* Each deterministic (low, intermediate, high) scenario
loop (s,
   DEM(j) = DEMS(s,j)
   solve DETERMINISTIC minimizing TCOST using LP
) ;

* Mean-load scenario
DEM(j) = sum[s, PROB(s) * DEMS(s,j)]
solve DETERMINISTIC minimizing TCOST using LP

* Stochastic problem
solve STOCHASTIC minimizing TCOST using LP
```

GEP model: analysis of the results

- Deterministic (perfect information) decisions not necessarily appear in the optimal stochastic solution
- Optimal stochastic solution doesn't appear necessarily in any deterministic scenario

	Det 1	Det 2	Det 3	Mean	Stochast
Gen 1 [MW]	.	0.33	3.67	0.67	0.67
Gen 2 [MW]	.	.	.	.	2
Gen 3 [MW]	3	4.67	3.33	4.53	4.33
Gen 4 [MW]	9	7	5	6.8	5
Total Cost [€]	262	346.67	437.33	355.73	362.47

Alternatives for modeling the uncertainty

- Wait and see o scenario analysis o what-if analysis o sensitivity analysis
  - Decisions are taken once solved the uncertainty
  - The problem is solved independently for each scenario
  - The scenario with mean value of the parameters is just a special case
  - A priori, decisions will be different for each deterministic scenario (anticipative, clairvoyant, not implementable)
  - Solution of a scenario can be infeasible in the others

- Here and now (stochastic) decisions
  - Decisions must be taken before solving uncertainty
  - Non anticipative decisions (only the available information so far can be used, no future information)
  - The only relevant decisions are those of the first stage, given that are the only to be taken immediately
  - Stochastic solution considers the stochasticity distribution
  - It allows to include risk averse attitudes, penalizing worst cases

Definitions in stochastic problems. Stochastic measures

- *Expected value with perfect information* ($\boldsymbol{EVWPI}$) o *Wait and See* ($\boldsymbol{WS}$)
  - Weighted mean of the objective function of each scenario knowing that is going to happen (356.93 for the example) (for minimization problems always ≤ than the objective function for the stochastic problem, 280, 349.33 and 439.33 respectively, that give a weighted value of 362.47)
- *Value of the stochastic solution* ($\boldsymbol{VSS}$) o *Expected Value of Including Uncertainty* ($\boldsymbol{EVIU}$)
  - Difference between the expected objective function for the mean-value solution of the stochastic parameters *Expectation of EV Problem* ($\boldsymbol{EEV}$) (280x0.2+347.73x0.5+454.73x0.3=366.28) and that of the stochastic problem $\boldsymbol{RP}$ (366.28–362.47=3.81)

$$\boldsymbol{VSS = EEV - RP}$$

$$\boldsymbol{VSS \geq 0}$$

- *Expected value of perfect information* ($\boldsymbol{EVPI}$) o mean regret
  - Weighted average of the difference between the stochastic solution for each scenario and the perfect information solution in this scenario (always positive for minimization) (280–262=18, 349.33–346.67=2.66, 439.33–437.33=2) (18x0.2+2.66x0.5+2x0.3=5.54)

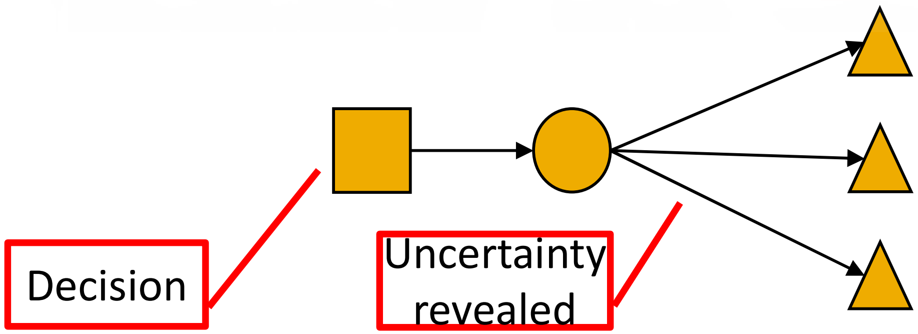
$$\boldsymbol{EVPI = RP - WS}$$

$$\boldsymbol{EVPI \geq 0}$$

$$\boldsymbol{WS \leq RP \leq EEV}$$

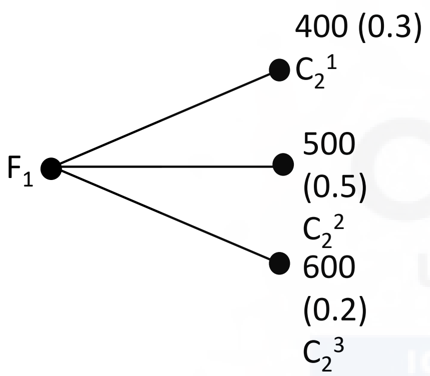
Decision tree

- Represents a sequence of decisions and random events that constitute the decision process.
- Elements of the tree:
  - Chance node: random points, are represented by a circle.
  - Decision node : decision points, are represented by a square.
  - Initial node or root: root of the tree with the initial decisions. Always the first thing to do is to take a decision.
  - Final node or leaf: finals points, are represented by triangles.



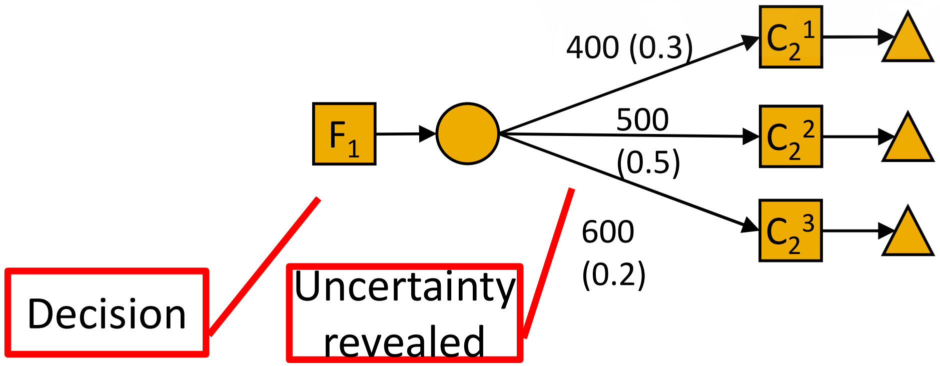
Example: Manufacturing. Two-stage stochastic problem

- Manufacturing decision of an amount of a product with a cost of 2 €/unit to satisfy a random demand. If no enough product is manufactured it can be bought from an external provider at a price of 4 €/unit.



$$\min 2F_1 + 4(0.3C_2^1 + 0.5C_2^2 + 0.2C_2^3)$$
$$F_1 + C_2^1 \geq 400$$
$$F_1 + C_2^2 \geq 500$$
$$F_1 + C_2^3 \geq 600$$
$$F_1, C_2^1, C_2^2, C_2^3 \geq 0$$

F_1	C_2^1	C_2^2	C_2^3



$$F_1^* = 500$$
$$C_2^{1*} = 0; C_2^{2*} = 0; C_2^{3*} = 100$$

Two-stage stochastic problem

1. Today, a set of first-stage decisions are taken

2. At night, some (exogenous) random events occur

3. Tomorrow a set of corrective actions in the second stage are taken to mitigate (fix) the effects of the random events over the today decisions. Second stage decisions are the recourses.

Recourse. Recourse function

- Capability of taking a corrective action after occurring a random event
- Recourse function: objective function associated with the corrective actions.
- Depends on the previous decisions and on the random events.
- If stages are in the time domain, the recourse function is the future cost function (cost-to-go function)
- Two-stage stochastic linear optimization (Generation expansion planning):
  - First stage decisions are deterministic, unique (investment)
  - Second stage (recourse) decisions are stochastic (operation)

Type of recourse

- Complete
  - All the first-stage decisions are feasible for any second-stage scenario
- Relatively complete
  - All the first-stage feasible decisions are feasible for any second-stage scenario
- Partial

Penalizing the constraint violation

- Slack (deficit and surplus) variables are introduced in the stochastic constraints and are penalized in the objective function

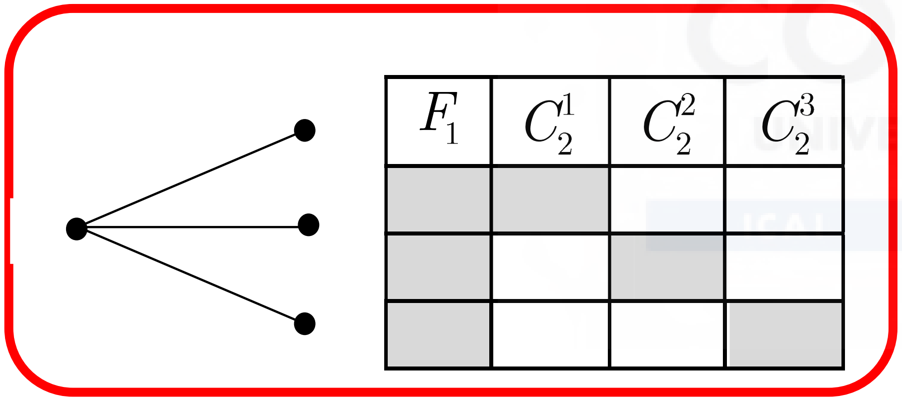
Example: Manufacturing. Two-stage stochastic problem

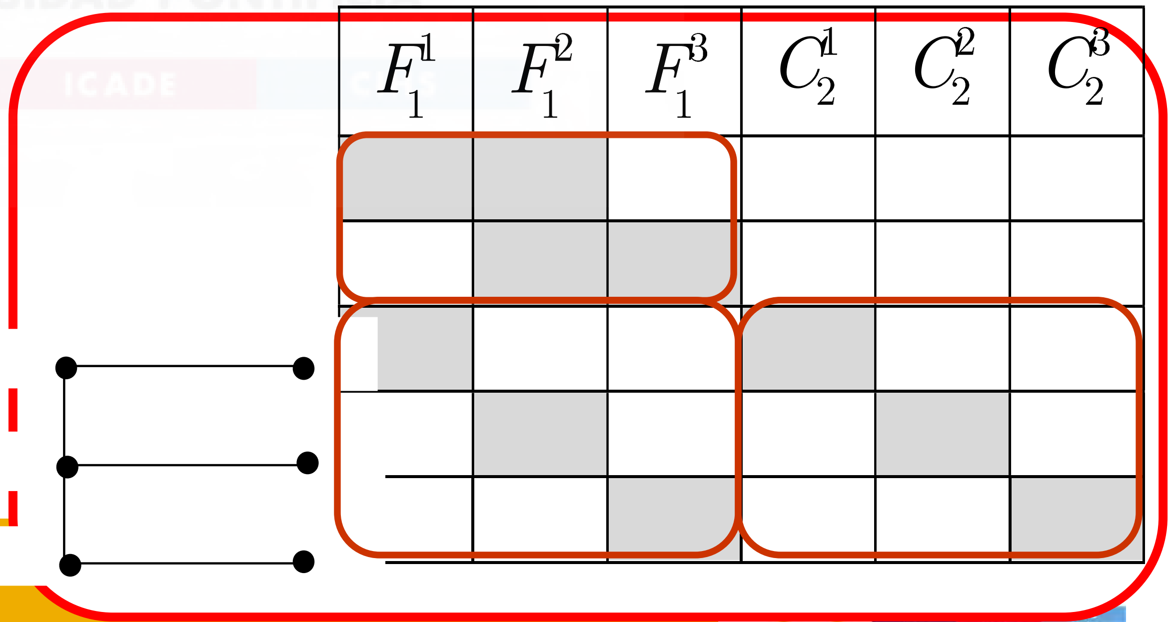
$$\min 2F_1 + 4(0.3C_2^1 + 0.5C_2^2 + 0.2C_2^3)$$
$$F_1 + C_2^1 \geq 400$$
$$F_1 + C_2^2 \geq 500$$
$$F_1 + C_2^3 \geq 600$$
$$F_1, C_2^1, C_2^2, C_2^3 \geq 0$$

$$\min 2(0.3F_1^1 + 0.5F_1^2 + 0.2F_1^3) + 4(0.3C_2^1 + 0.5C_2^2 + 0.2C_2^3)$$
$$F_1^1 = F_1^2$$
$$F_1^2 = F_1^3$$

Non anticipativity constraints

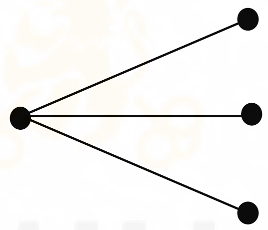
$$F_1^1 + C_2^1 \geq 400$$
$$F_1^2 + C_2^2 \geq 500$$
$$F_1^3 + C_2^3 \geq 600$$
$$F_1^1, F_1^2, F_1^3, C_2^1, C_2^2, C_2^3 \geq 0$$



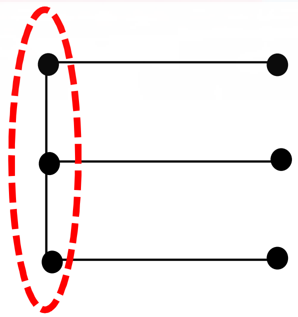


Tree representation

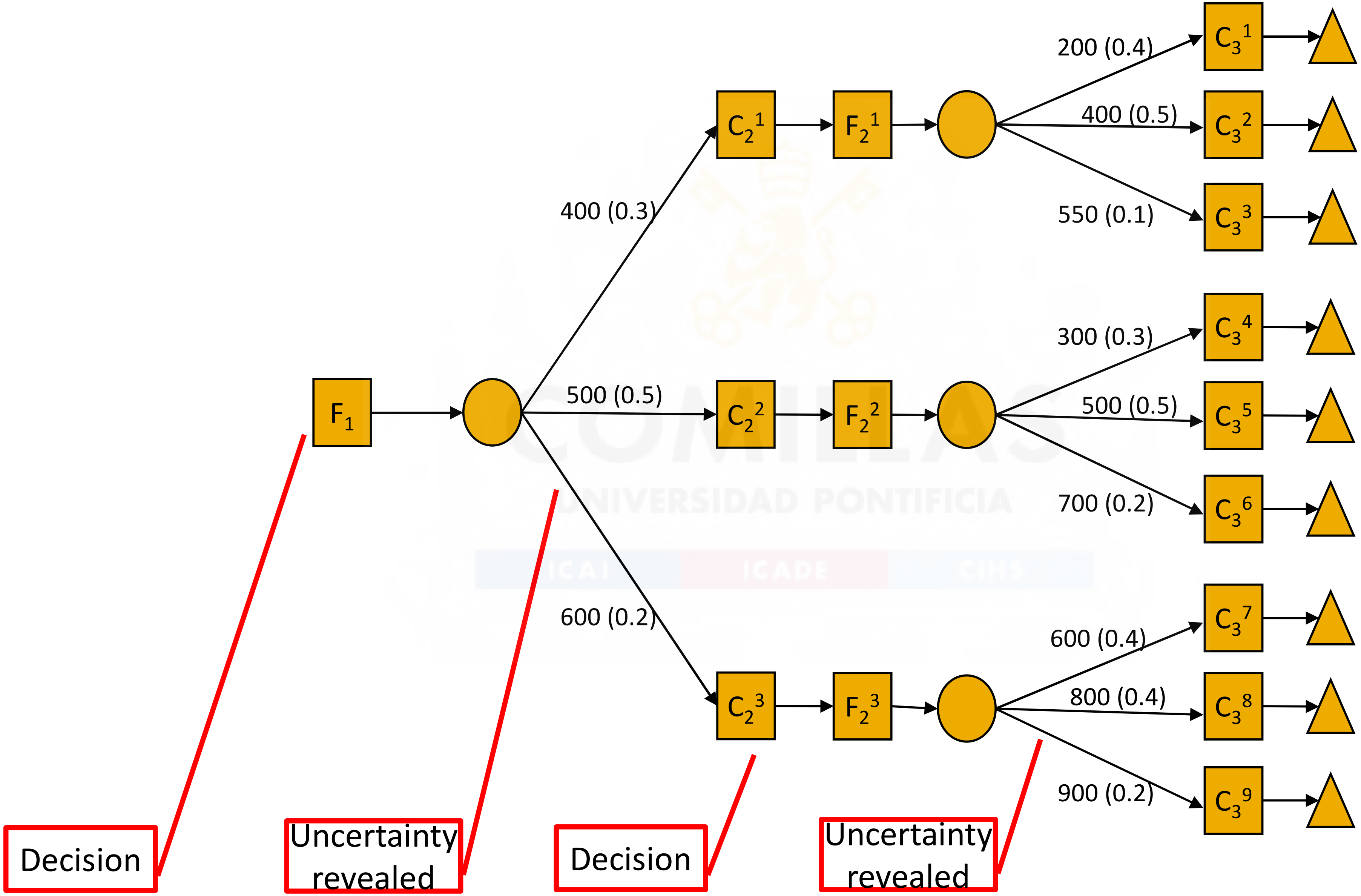
- Implicit formulation in the definition of parameters and variables
  - Decomposition technique (Benders, etc.)



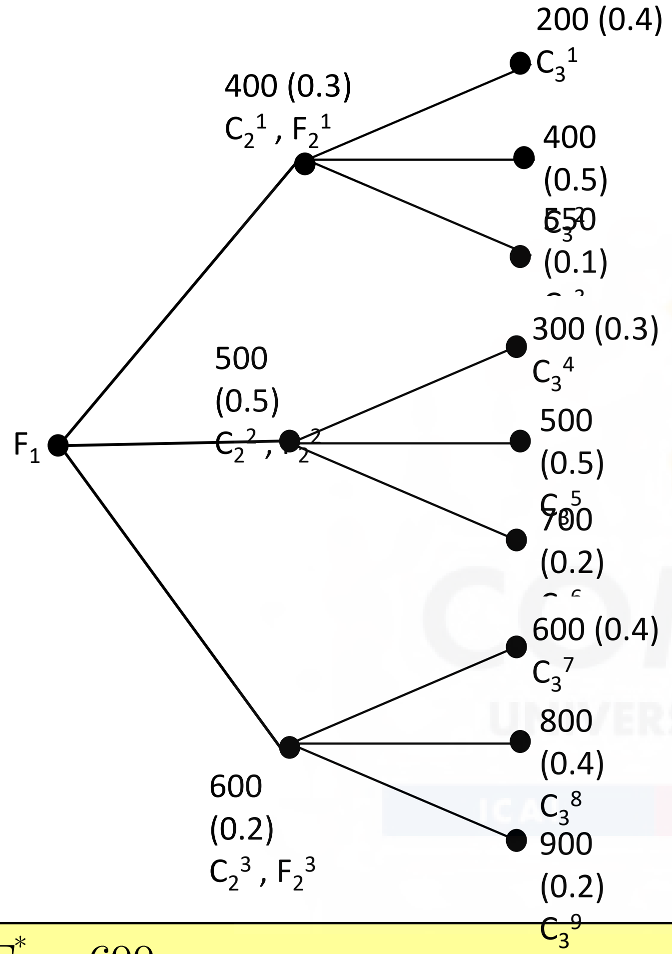
- Explicit formulation by constraints
  - *Scenario decomposition* (*splitting variables, non anticipativity constraints*) y Lagrangian relaxation



Example: Manufacturing. Three-stage stochastic problem. Decision tree



Example: Manufacturing. Three-stage stochastic problem



$$\min 2F_1 + 4(0.3C_2^1 + 0.5C_2^2 + 0.2C_2^3) + 2(0.3F_2^1 + 0.5F_2^2 + 0.2F_2^3) + 4(0.12C_3^1 + 0.15C_3^2 + 0.03C_3^3 + 0.15C_3^4 + 0.25C_3^5 + 0.1C_3^6 + 0.08C_3^7 + 0.08C_3^8 + 0.04C_3^9)$$

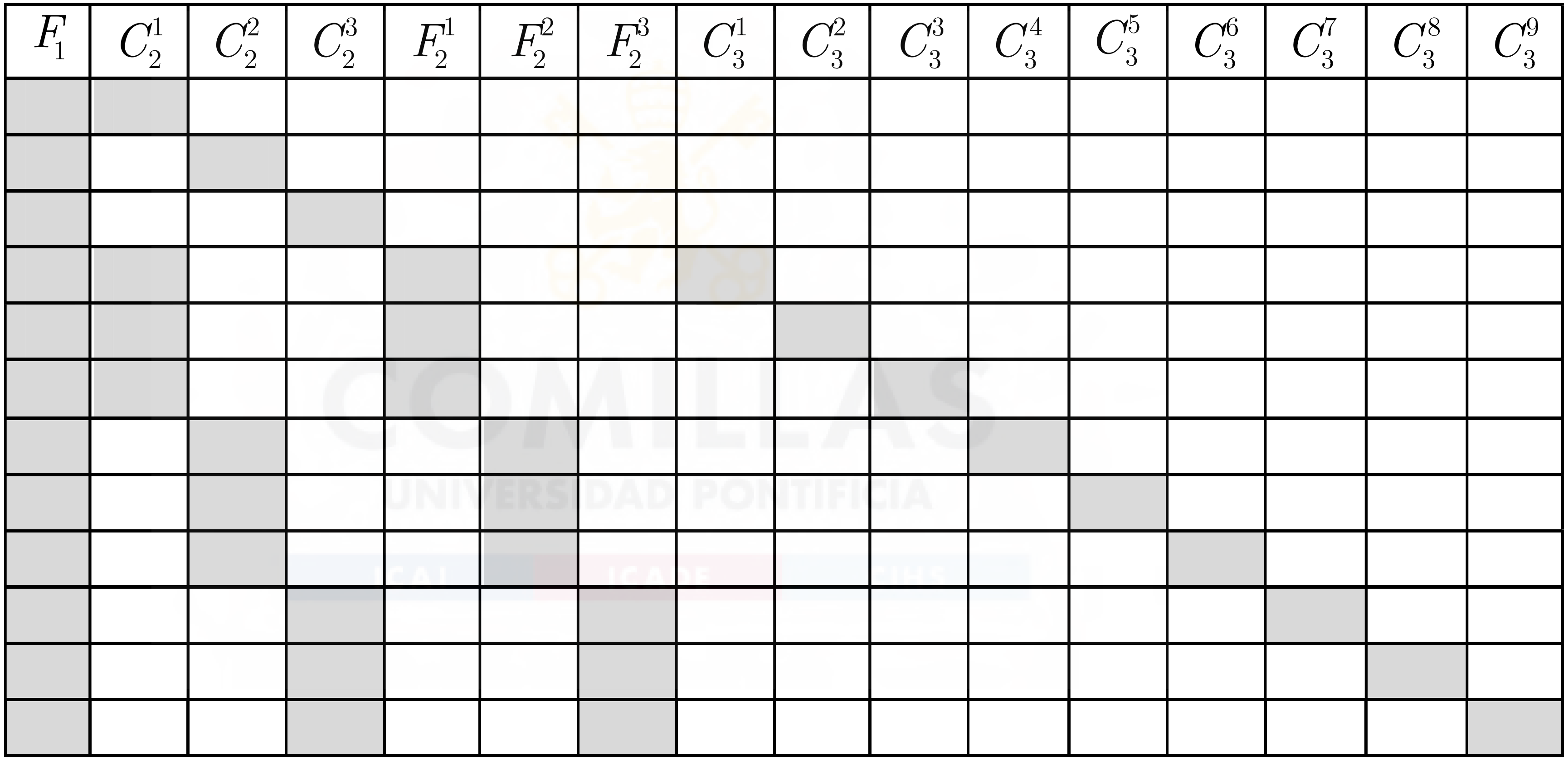
$$F_1 + C_2^1 \geq 400$$
$$F_1 + C_2^2 \geq 500$$
$$F_1 + C_2^3 \geq 600$$
$$F_1 + C_2^1 - 400 + F_2^1 + C_3^1 \geq 200$$
$$F_1 + C_2^1 - 400 + F_2^1 + C_3^2 \geq 400$$
$$F_1 + C_2^1 - 400 + F_2^1 + C_3^3 \geq 550$$
$$F_1 + C_2^2 - 500 + F_2^2 + C_3^4 \geq 300$$
$$F_1 + C_2^2 - 500 + F_2^2 + C_3^5 \geq 500$$
$$F_1 + C_2^2 - 500 + F_2^2 + C_3^6 \geq 700$$
$$F_1 + C_2^3 - 600 + F_2^3 + C_3^7 \geq 600$$
$$F_1 + C_2^3 - 600 + F_2^3 + C_3^8 \geq 800$$
$$F_1 + C_2^3 - 600 + F_2^3 + C_3^9 \geq 900$$
$$F_1, C_2^1, C_2^2, C_2^3, F_2^1, F_2^2, F_2^3, C_3^1, C_3^2, C_3^3, C_3^4, C_3^5, C_3^6, C_3^7, C_3^8, C_3^9 \geq 0$$

$$F_1^* = 600$$
$$C_2^{1*} = C_2^{2*} = C_2^{3*} = 0$$
$$F_2^{1*} = 200; F_2^{2*} = 400; F_2^{3*} = 800$$
$$C_3^{1*} = C_3^{2*} = C_3^{4*} = C_3^{5*} = C_3^{7*} = C_3^{8*} = 0$$
$$C_3^{3*} = 150; C_3^{6*} = 200; C_3^{9*} = 100$$

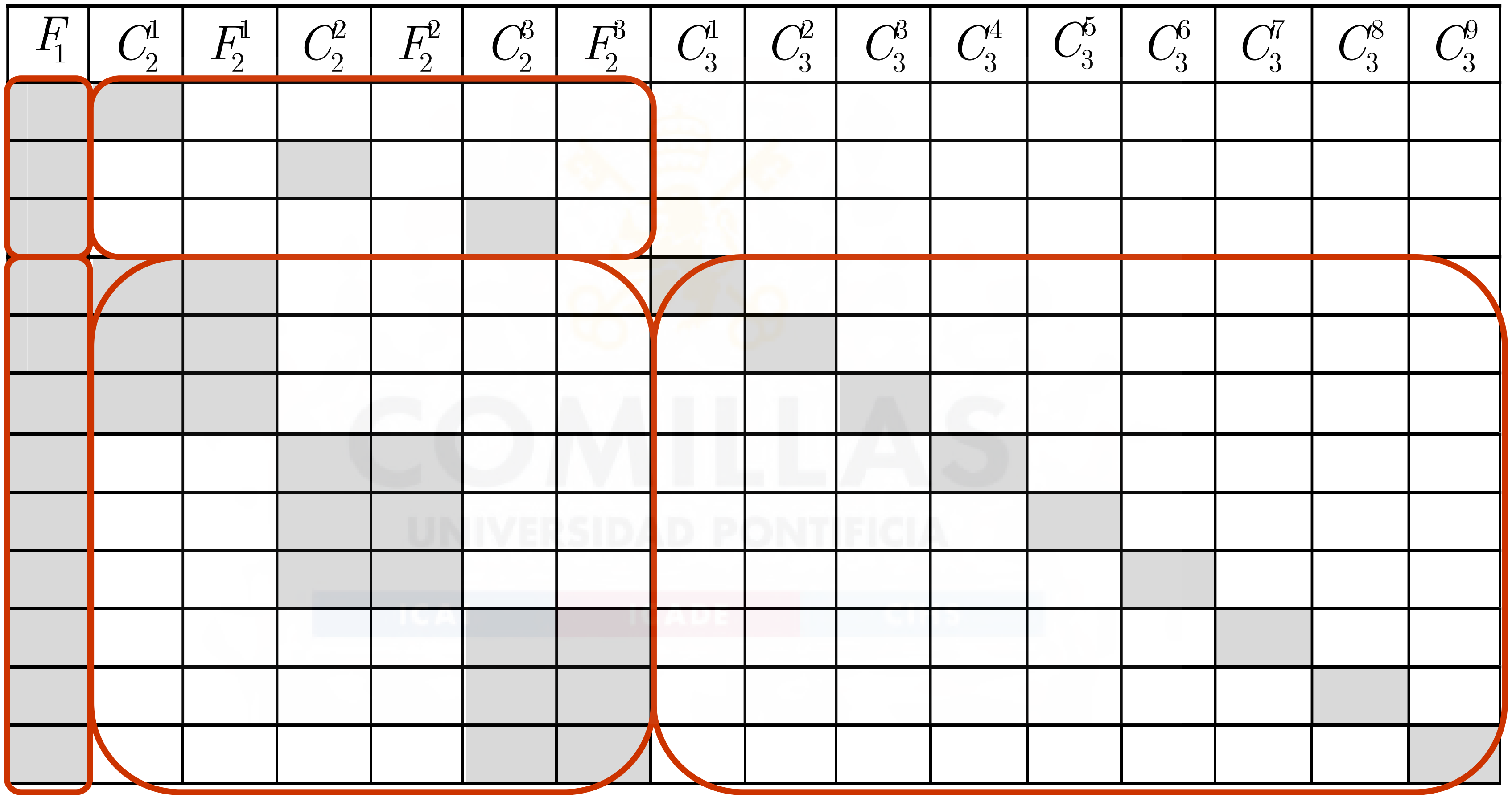
Probability or scenario tree

- Represents the evolution in realization of uncertainty along the time, different values of the random parameters along the time.
- Scenario: any path from the root to the leafs
- The scenarios that share information up to a certain time period share the same decisions in the tree (implementable decisions)
- The probability tree represents the dynamics of the random parameters and the non anticipativity of the decisions and, therefore, is implicit in the constraint matrix

Example: Manufacturing. Three-stage stochastic problem. Constraint matrix



Example: Manufacturing. Three-stage stochastic problem. Reordered constraint matrix

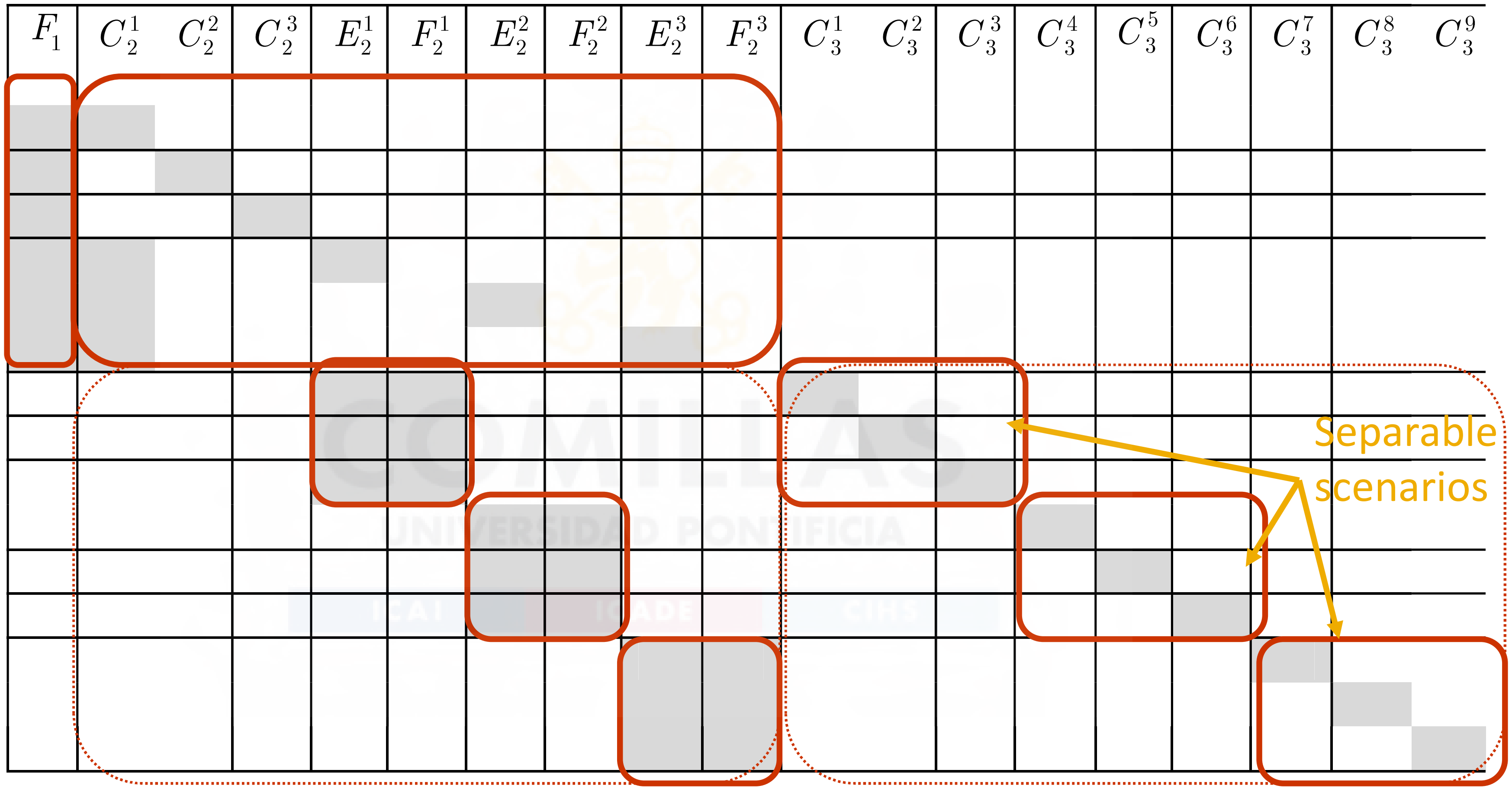


Example: Manufacturing. Three-stage stochastic problem. Problem reformulation

$$\min 2F_1 + 4(0.3C_2^1 + 0.5C_2^2 + 0.2C_2^3) + 2(0.3F_2^1 + 0.5F_2^2 + 0.2F_2^3) +$$
$$+4(0.12C_3^1 + 0.15C_3^2 + 0.03C_3^3 + 0.15C_3^4 + 0.25C_3^5 +$$
$$+0.1C_3^6 + 0.08C_3^7 + 0.08C_3^8 + 0.04C_3^9)$$
$$F_1 + C_2^1 \geq 400$$
$$F_1 + C_2^2 \geq 500$$
$$F_1 + C_2^3 \geq 600$$
$$F_1 + C_2^1 - 400 = E_2^1$$
$$F_1 + C_2^2 - 500 = E_2^2$$
$$F_1 + C_2^3 - 600 = E_2^3$$
$$E_2^1 + F_2^1 + C_3^1 \geq 200$$
$$E_2^1 + F_2^1 + C_3^2 \geq 400$$
$$E_2^1 + F_2^1 + C_3^3 \geq 550$$
$$E_2^2 + F_2^2 + C_3^4 \geq 300$$
$$E_2^2 + F_2^2 + C_3^5 \geq 500$$
$$E_2^2 + F_2^2 + C_3^6 \geq 700$$
$$E_2^3 + F_2^3 + C_3^7 \geq 600$$
$$E_2^3 + F_2^3 + C_3^8 \geq 800$$
$$E_2^3 + F_2^3 + C_3^9 \geq 900$$
$$F_1, C_2^1, C_2^2, C_2^3, F_2^1, F_2^2, F_2^3, E_2^1, E_2^2, E_2^3, C_3^1, C_3^2, C_3^3, C_3^4, C_3^5, C_3^6, C_3^7, C_3^8, C_3^9 \geq 0$$

An inventory variable is introduced at the start of period 2

Example: Manufacturing. Three-stage stochastic problem. Constraint matrix



Fixed Cost Transportation Problem (FCTP)

- It is a transportation problem where the arc connecting two nodes (i and j) has a fixed cost f_{ij} associated with its installation and a variable cost c_{ij} by the use. We want to minimize the total fixed (investment) and variable (transportation) costs subject to the constraints of demand supply b_j at the destinations and maximum capacity at the origins a_i.

Deterministic and stochastic FCTP

Flows
(second stage)

Investment decisions
(first stage)

Capacity of each origin

Demand of each destination

Flow can go only for installed connections

Deterministic

$$\min_{x_{ij}, y_{ij}} \sum_{ij} (f_{ij} y_{ij} + c_{ij} x_{ij})$$

$$\sum_j x_{ij} \leq a_i \quad \forall i$$

$$\sum_i x_{ij} \geq \boldsymbol{b_j} \quad \forall j$$

$$x_{ij} \leq M_{ij} y_{ij} \quad \forall ij$$

$$x_{ij} \geq 0, y_{ij} \in \{0,1\}$$

Stochastic

$$\min_{x_{ij}^\omega, y_{ij}} \sum_{ij} \left(f_{ij} y_{ij} + \sum_\omega p_\omega c_{ij} x_{ij}^\omega \right)$$

$$\sum_j x_{ij}^\omega \leq a_i \quad \forall i\omega$$

$$\sum_i x_{ij}^\omega \geq \boldsymbol{b_j^\omega} \quad \forall j\omega$$

$$x_{ij}^\omega \leq M_{ij}^\omega y_{ij} \quad \forall ij\omega$$

$$x_{ij}^\omega \geq 0, y_{ij} \in \{0,1\}$$

Deterministic & Stochastic FCTP

```
$title Deterministic fixed-charge transportation problem (DFCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins           / i1 * i4 /
   J         destinations      / j1 * j3 /


parameters
   A(i)      product offer  / i1 20, i2 30, i3 40, i4 20 /
   B(j)      product demand / j1 20, j2 50, j3 30 /








table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

abort $(sum[i, A(i)] < sum[j, B(j)]) 'Infeasible problem'

positive variable
   X(i,j)        arc flow
binary   variable
   Y(i,j)        arc investment decision
variables
   Z1            objective function

equations
   EQ_OBJ          complete problem objective function
   Offer    (i  )  offer  at origin
   Demand   (  j)  demand at destination
   FlowLimit(i,j)  arc flow limit ;

EQ_OBJ         .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(i,j), C(i,j)*X(i,j)] ;
Offer    (i  ) .. sum[j, X(i,j)] =l= A(i) ;
Demand   (  j) .. sum[i, X(i,j)] =g= B(j) ;
FlowLimit(i,j) .. X(i,j) =l= min[A(i),B(j)] * Y(i,j) ;

model Complete      / EQ_OBJ Offer Demand FlowLimit / ;

X.up(i,j) = min[A(i),B(j)]

solve Complete using MIP minimizing Z1
```

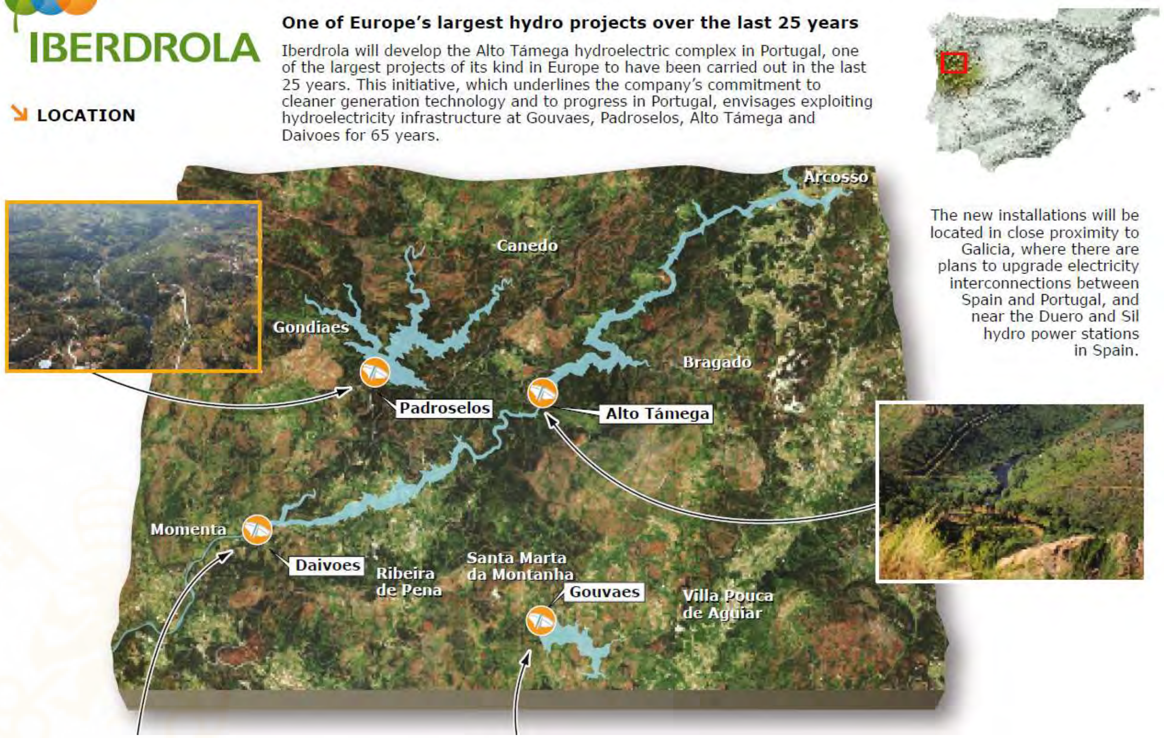
```
$title Stochastic fixed-charge transportation problem (SFCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins           / i1 * i4 /
   J         destinations      / j1 * j3 /
   S         scenarios         / s1 * s3 /

parameters
   A(i)      product offer        / i1 20, i2 30, i3 40, i4 20 /
   P(s)      scenario probability / s1 0.5, s2 0.3, s3 0.2 /

table B(s,j) product demand
       j1  j2  j3
   s1  21  51  31
   s2  32  22  52
   s3  53  33  23

table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

loop (s, abort $(sum[i, A(i)] < sum[j, B(s,j)]) 'Infeasible problem' )

positive variable
   X(s,i,j)      arc flow
binary   variable
   Y(i,j  )      arc investment decision
variables
   Z1            objective function

equations
   EQ_OBJ            complete problem objective function
   Offer    (s,i  )  offer  at origin
   Demand   (s,  j)  demand at destination
   FlowLimit(s,i,j)  arc flow limit ;

EQ_OBJ           .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(s,i,j), P(s)*C(i,j)*X(s,i,j)] ;
Offer    (s,i  ) .. sum[j, X(s,i,j)]  =l= A(  i) ;
Demand   (s,  j) .. sum[i, X(s,i,j)]  =g= B(s,j) ;
FlowLimit(s,i,j) .. X(s,i,j) =l= min[A(i),B(s,j)] * Y(i,j) ;

model Complete      / EQ_OBJ Offer Demand FlowLimit / ;

X.up(s,i,j) = min[A(i),B(s,j)]

solve Complete using MIP minimizing Z1
```

Stochastic FCTP with EMP

```
$title Deterministic fixed-charge transportation problem (FCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins          / i1 * i4 /
   J         destinations     / j1 * j3 /

parameters
   A(i)      product offer
             / i1 20, i2 30, i3 40, i4 20 /
   B(j)      product demand
             / j1 11, j2 44, j3 66 /

table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

positive variable
   X(i,j)        arc flow

binary   variable
   Y(i,j)        arc investment decision

variables
   Z1            objective function

equations
   EQ_OBJ          complete problem objective function
   Offer     (i  ) offer  at origin
   Demand    (  j) demand at destination
   FlowLimit(i,j) arc flow limit ;

EQ_OBJ          .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(i,j), C(i,j)*X(i,j)] ;

Offer    (i  ) .. sum[j, X(i,j)] =l= A(i) ;

Demand   (  j) .. sum[i, X(i,j)] =g= B(j) ;

FlowLimit(i,j) .. X(i,j) =l= 100 * Y(i,j) ;

model Complete       / all / ;

X.up(i,j) = 100
```

```
set          S       scenarios            / s1 * s3 /
parameter P(s)       scenario probability / s1 0.5, s2 0.3, s3 0.2 /
          YS(s,i,j) arc investment decision
          XS(s,i,j) arc flow

table  BS(s,j) product demand
       j1  j2  j3
   s1  21  51  31
   s2  32  22  52
   s3  53  33  23

file     emp / '%emp.info%' / ; emp.pc=2
put      emp
put '* problem %gams.i%' / "jrandvar "
loop (j,
   put B.tn(j) " "
)
loop (s,
   put P(s)
   loop (j,
      put BS(s,j)
   )
)
put / "stage 2 B X Offer Demand FlowLimit"
putclose emp

set dict / s . scenario . ''
           B . randvar  . BS
           X . level    . XS
           Y . level    . YS /

loop (s, abort $(sum[i, A(i)] < sum[j, BS(s,j)]) 'Infeasible problem'
) ;

solve Complete minimizing Z1 using emp scenario dict

display XS, YS
```

Hydrothermal Scheduling



- The objective of the problem is to determine a generation schedule for a hydrothermal generating system such that expected operating costs are minimized. These costs are composed of fuel costs for the thermal units and penalties for failure to meet power demand. The hydrothermal system consists of four hydroelectric reservoirs and one thermal generation unit. Constraints to the problem are the mass balance equations for the water supply across the reservoirs and across the stages; the demand constraints and the capacity of the operating units. The schedule must be developed for three time periods.

Example: Hydrothermal Scheduling Problem

- Scenario analysis (deterministic)
  - Run the model supposing that the natural inflows will be the same as any of the previous historical inflows (i.e., year 1989 or 2004, etc.) for the time scope
  - Run the model supposing that the natural inflows for each period will be exactly the mean of the historical values (i.e., average year) for the time scope

- Stochastic optimization
  - Run the model considering that the distribution of future natural inflows will be the same as it has been in the past

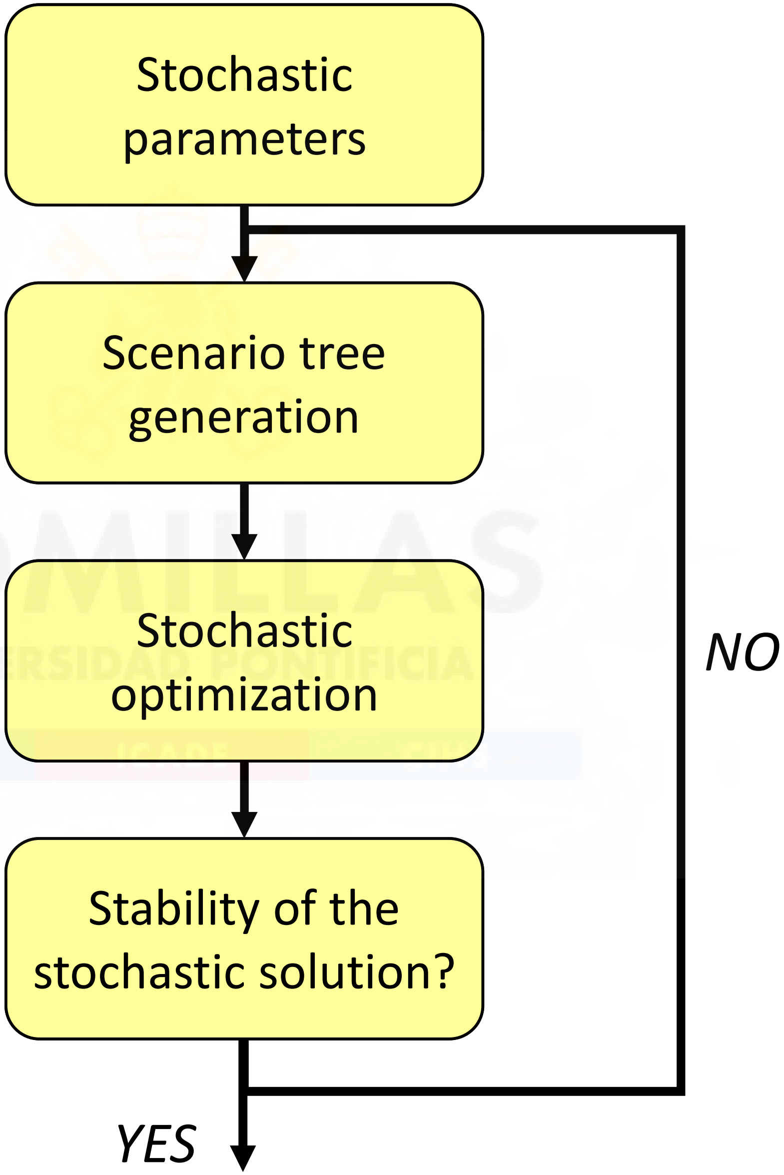
Multistage stochastic optimization

- Taking optimal decisions in different stages in presence of random parameters with known distributions to minimize the expected value
- General formulation of the stochastic optimization problem:

$$\min_{x \in X} E_P\{f(\omega, x)\} = \min_{x \in X} \int_{\Omega} f(\omega, x) \cdot dP(\omega)$$

- Uncertainty is represented by a scenario tree

Solving a stochastic model

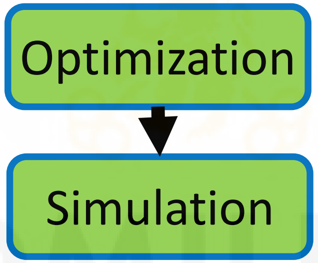


Stability of the stochastic solution

- The main stochastic solutions (i.e., the first-stage ones) must be “the same” against the uncertainty modeling (structure and number of scenarios of the tree)
- A scenario tree must be generated such as the solution of the stochastic model ought to be independent of it
- Analyze the stochastic solutions for different increasing scenario trees and stop when there is no change in the optimal solution (or vice versa)

Optimization-simulation combination

- Use the model in an open-loop control mechanism with rolling horizon
  1. First: planning by stochastic optimization
  2. Second: simulation of the random parameters

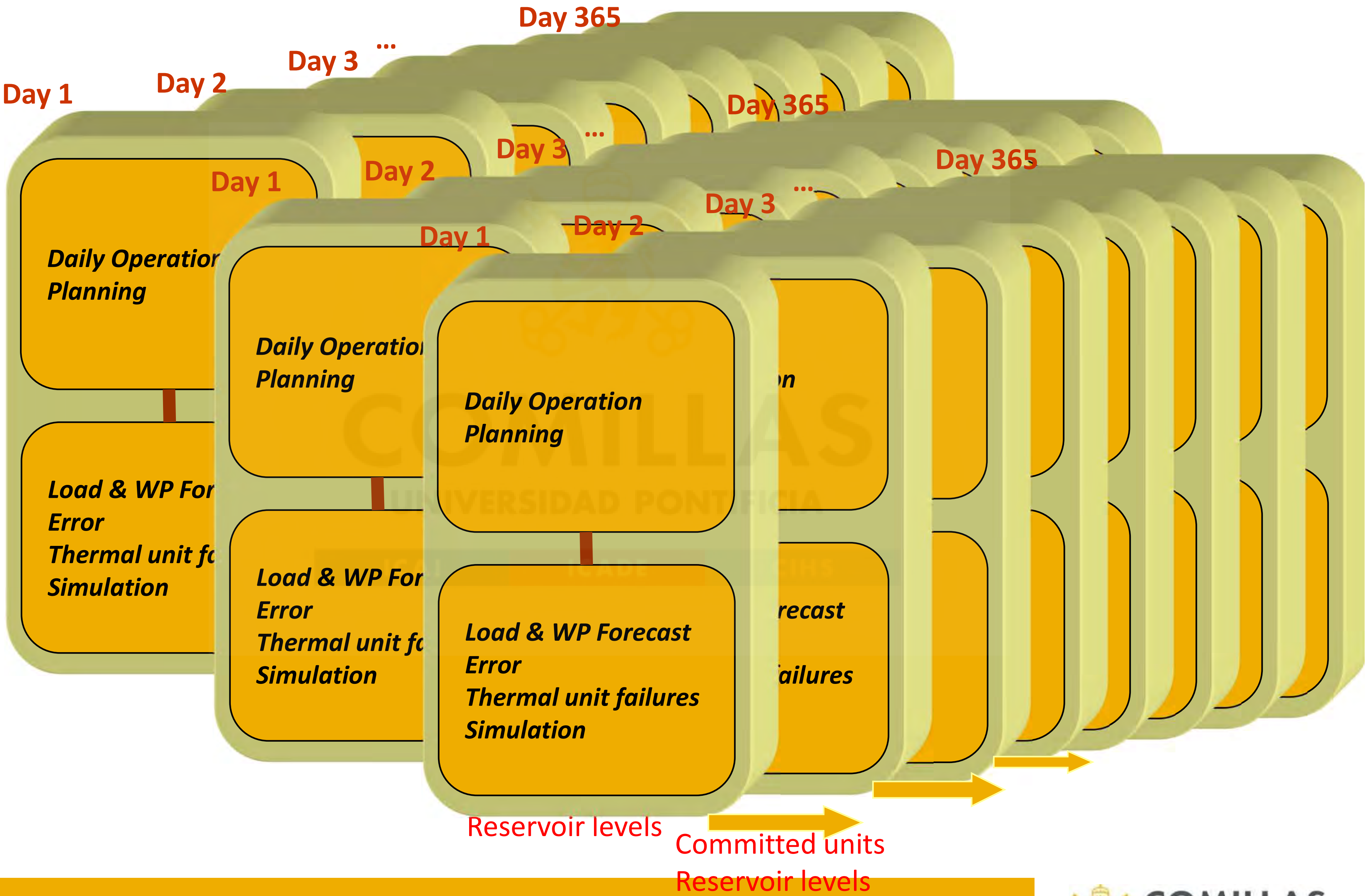


- Stochastic optimization
  - Determines optimal policies considering the uncertainty
- Simulation
  - Evaluates possible future outcomes of random parameters given the optimal policies obtained previously

Rolling horizon ⏭

- It is mechanism to reproduce in an accelerated way the decision process
- Rolling horizon for making decisions under uncertainty:
  1. Representation of the uncertainty (e.g., scenario tree)
     - It can be a single scenario
  2. Stochastic optimization but only first-stage decisions are considered implementable
     - Deterministic vs. stochastic
     - Scope of the model (perfect foresight, myopic)
  3. Draw a scenario for the first stage (Monte Carlo sampling, time series forecasting) and determine system operation
  4. Advance horizon and update the statistical representation of stochastic parameters and system state
  5. Go to 1

ROM General overview (https://pascua.iit.comillas.edu/aramos/ROM.htm)



Rolling horizon application in ROM

1. Two-stage scenario tree with RES uncertainty
2. Stochastic daily UC (myopic)
3. Sample forecast errors (demand, RES) and generation contingencies and determine system operation
4. Advance one day and no update of the statistical representation of stochastic parameters (i.e., forecast errors do not consider the previous samples)
5. Go to 1

Rolling horizon application in GADES

1. Multistage scenario tree with hydro inflows uncertainty
2. Medium term stochastic hydrothermal model (24-month scope)
3. Sample hydro inflows and determine monthly system operation with future hydro storage value represented by future-cost function
4. Advance one month and no update of the statistical representation of stochastic parameters (i.e., same scenario tree) nor of the optimal policy (i.e., future-cost function)
5. Go to 3

Cross-validation

- Stochastic data split in two sets:
  - Training set to determine the scenario tree (in-sample)
  - Test/validation set to evaluate/validate the optimal policies (out-of-sample)

- Performance measures of optimal policies based on out-of-sample values

Probabilistic or chance constraints

- The probability of satisfying a constraint with stochastic parameters should be greater than a certain probability
- Or, alternatively, the probability of satisfying a set of constraints with stochastic parameters should be greater than a certain probability

$$P\left(\sum_j a_j^\omega x_j \le b^\omega \quad \forall \omega\right) \ge \alpha$$

- In case of discrete distributions a binary variable is needed for each event (a realization of the discrete parameters)

Robust optimization

- Risk-averse criterion (minimax or Savage's criterion)
- There may exist catastrophic scenarios or high nonlinearity in the objective function
- There are no probabilities associated with the scenarios or they are not used
- Objective function:
  - Minimize the maximum regret
  - Minimize the maximum value of the objective function
- Robust optimization
  - Robust solution if it is like the optimal one in all the scenarios
  - Robust model if it is almost feasible in all the scenarios
  - Robust optimization balances both objectives: like optimal solution and feasible

2

1. General overview
2. **Applications in electric systems**
3. Two-stage and multistage programming
4. Decomposition techniques
5. Benders decomposition
6. Nested Benders decomposition
7. Dantzig-Wolfe decomposition
8. Lagrangian relaxation
9. Scenario tree
10. Decomposition in two-stage and multistage stochastic programming
11. Improvements in decomposition techniques
12. Simulation in stochastic optimization
13. Stochastic dual dynamic programming

Applications in electric systems

Applications: optimal resource assignment

- Tutorial talks prior to the XIV Int. Conf. on Stochastic Programming (Búzios, Brazil, 2016):
- Finance (http://www.univie.ac.at/spxi/tutorial/TutPres_Vladimirou.ppt)
- Pension fund management (http://www.univie.ac.at/spxi/tutorial/TutPres_Hochreiter.pdf)
- Electric systems (http://www.univie.ac.at/spxi/tutorial/TutPres_Philpott.pps)
- Gas markets (http://www.univie.ac.at/spxi/tutorial/TutPres_Tomasgard.pdf)
- Logistics (http://www.univie.ac.at/spxi/tutorial/TutPres_Wallace.ppt)

Mathematical formulation of a stochastic UC

- Objective function
  - Minimize the total expected variable costs plus penalties for energy not served
- Variables
  - BINARY: commitment, startup and shutdown of thermal units
  - Hydro and thermal output
- Operation constraints
  - Load balance and operating reserve
  - Hydro and thermal operation constraints
  - Energy inventory in water reservoirs
- Mixed integer linear programming (MIP)

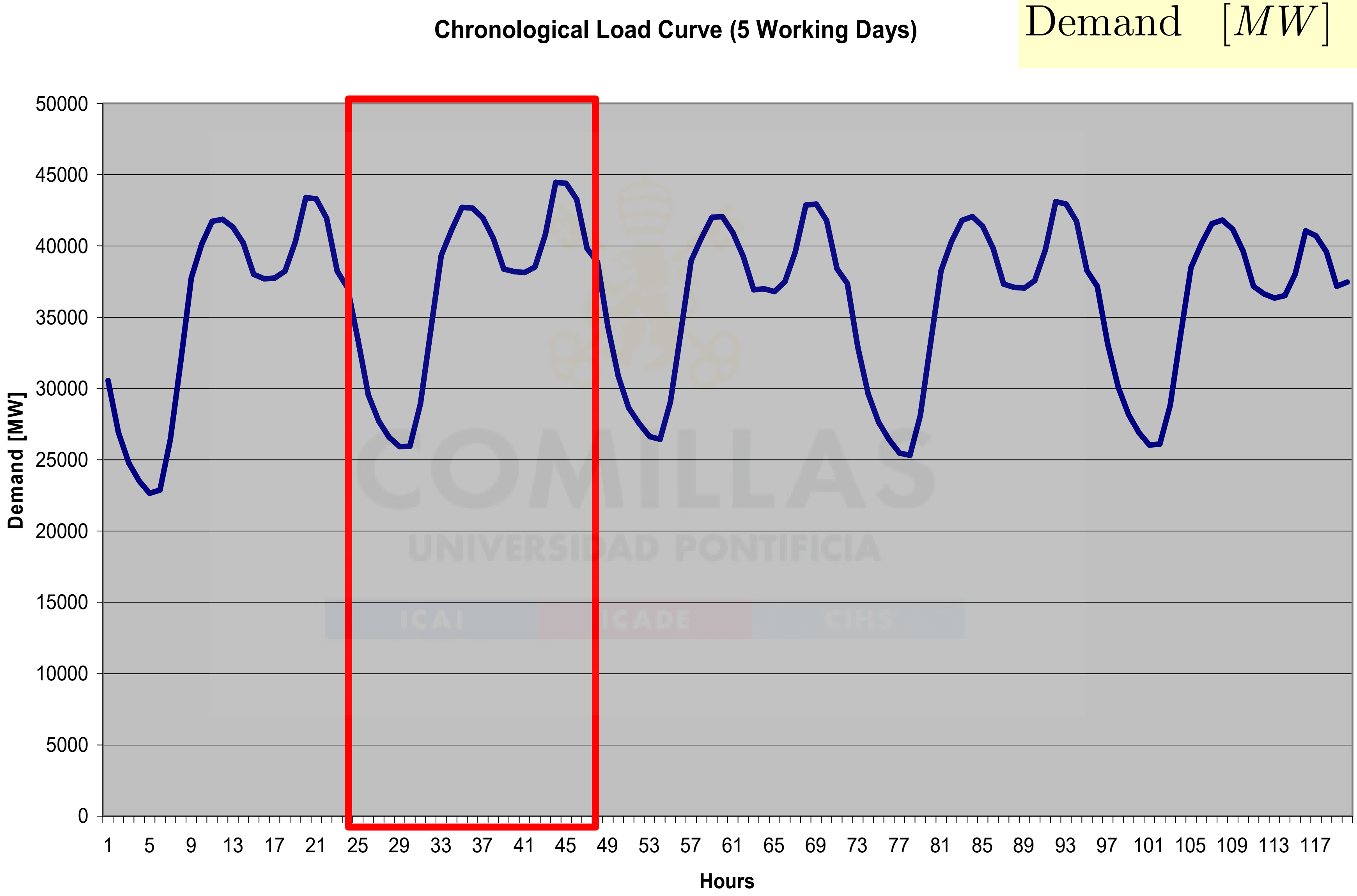
Indices

- Time scope
  - 1 day
- Period
  - 1 hour
- Scenario

Hour	n
Scenario	ω

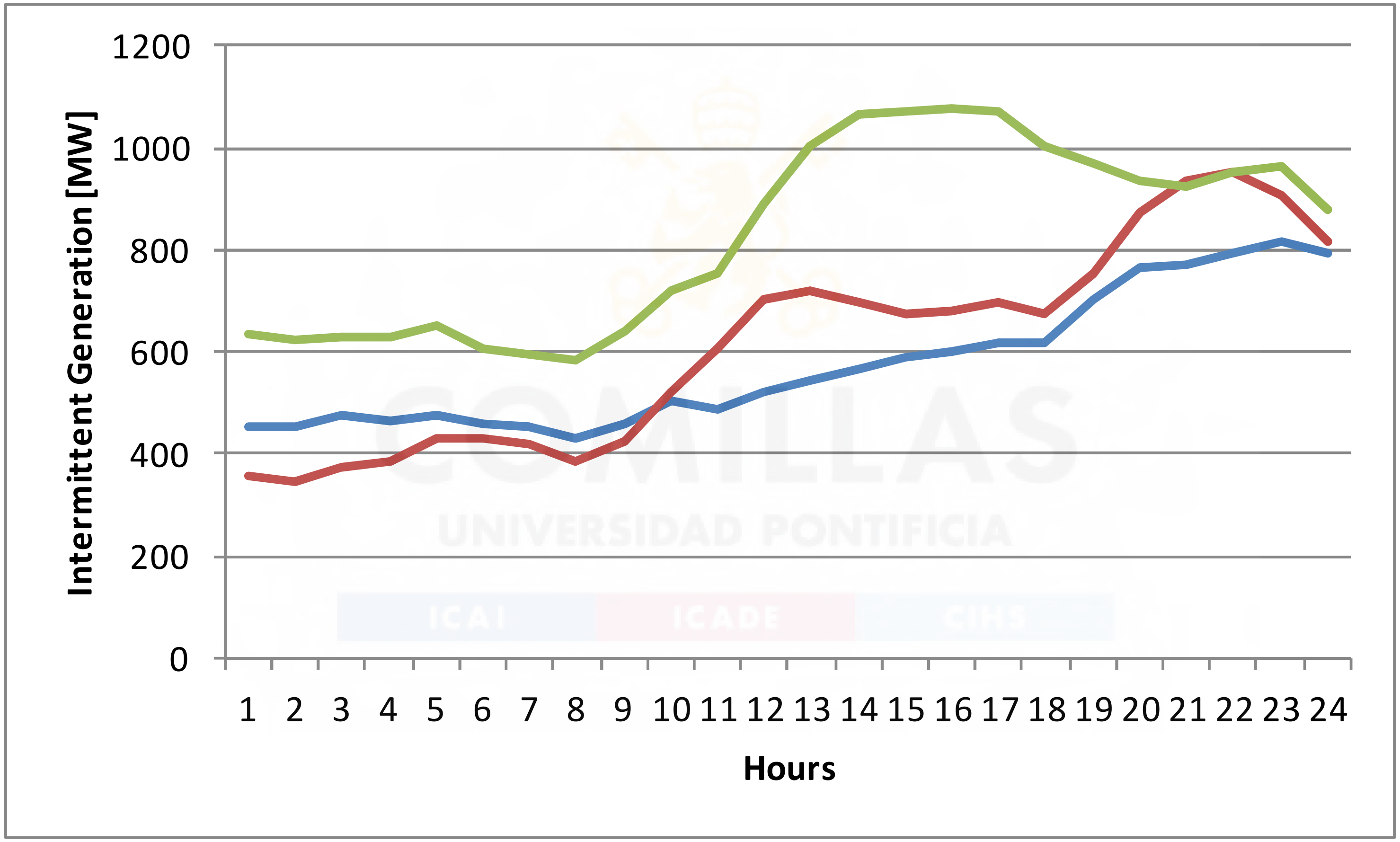
Demand (5 weekdays)

$\text{Demand} \quad [MW] \quad D_n$



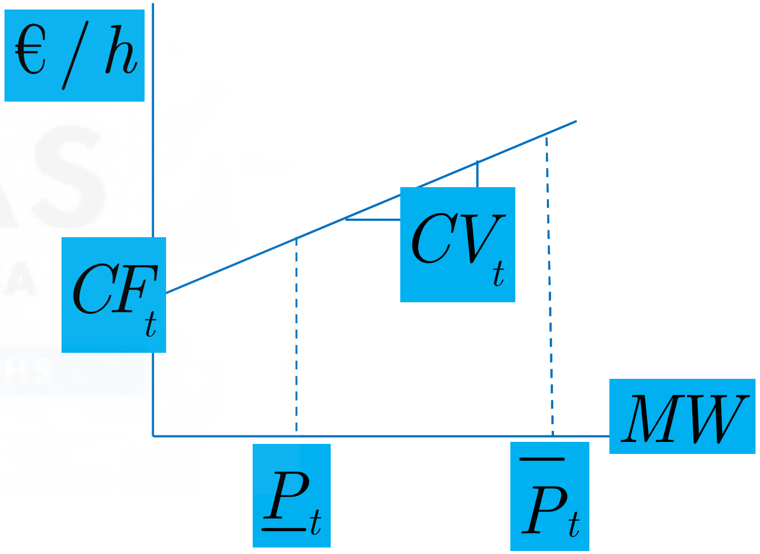
Intermittent generation (IG)

Intermittent generation $[MW]$ IG_n^{ω}



Technical characteristics of thermal units (t)

- Maximum and minimum output
- Fuel cost
- Slope and intercept of the heat rate straight line
- Operation and maintenance (O&M) variable cost
  - No load cost = fuel cost x heat rate intercept
  - Variable cost = fuel cost x heat rate slope + O&M cost
- Cold startup and shutdown cost
- Up and down ramps



Max and min output	$[MW]$	$\overline{P}_t, \underline{P}_t$
No load cost	$[€ / h]$	CF_t
Variable cost	$[€ / MWh]$	CV_t
Startup cost	$[€]$	CSU_t
Shutdown cost	$[€]$	CSD_t

Ramp up	$[MW / h]$	RU_t
Ramp down	$[MW / h]$	RD_t

Technical characteristics of hydro plants (*h*)

- Maximum and minimum output
- Production function (efficiency for conversion of water release in m^3/s to electric power MW)
- Efficiency of pumped storage hydro plants
  - Only this ratio of the energy consumed to pump the water is recovered by turbining this water

Max and min output	$[MW]$	$\overline{P}_h, \underline{P}_h$
Production function	$[kWh / m^3]$	C_h
Efficiency	$[p.u.]$	η_h

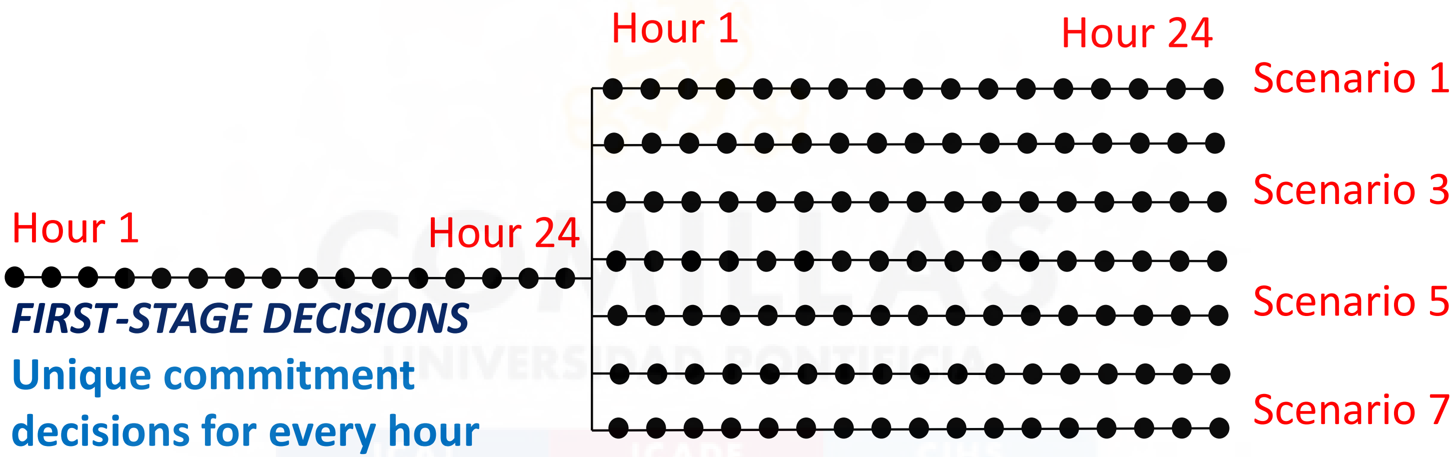
Technical characteristics of hydro reservoirs (*h*)

- Maximum and minimum reserve
- Initial reserve
  - Final reserve = initial reserve
- Inflows

Max and min reserve	$[hm^3]$	$\overline{R}_h, \underline{R}_h$
Initial and final reserve	$[hm^3]$	R'_h
Inflows	$[m^3 / s]$	I_{nh}

Scenario tree for the SDUC

- Commitment decisions of thermal units (the set of committed units) are unique under different stochastic scenarios (intermittent generation IG, demand, etc.)



FIRST-STAGE DECISIONS

Unique commitment decisions for every hour

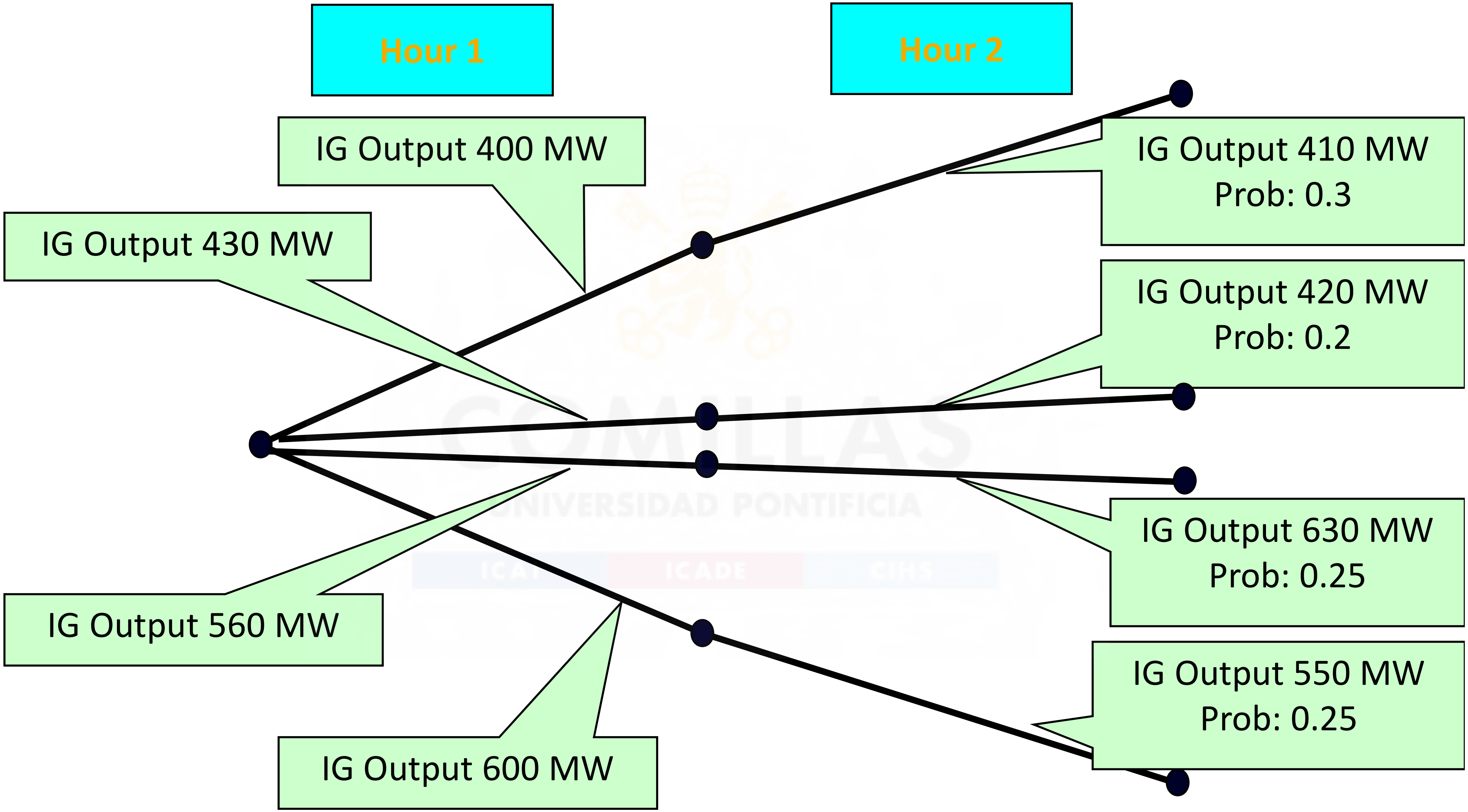
SECOND-STAGE DECISIONS

Generator output for every hour depend on the scenario

TWO-STAGE DECISION PROBLEM

Probability of scenario P^{ω}

Scenario tree example with IG uncertainty



Variables

- Commitment, startup and shutdown of thermal units (BINARY)

$$\text{Commitment, startup and shutdown} \quad \{0,1\} \quad uc_{nt}, su_{nt}, sd_{nt}$$

- Production of hydro and thermal units

$$\text{Production of a thermal and hydro unit} \quad [MW] \quad p_{nt}^{\omega}, p_{nh}^{\omega}$$

- Intermittent generation

$$\text{Intermittent generation} \quad [MW] \quad ig_{n}^{\omega}$$

- Reservoir volume

$$\text{Reservoir volumen} \quad [GWh] \quad r_{nh}^{\omega}$$

- Energy not served

$$\text{Energy not served} \quad [MW] \quad ens_{n}^{\omega}$$

Constraints: Operating power reserve

Committed output of thermal units
\+ *Maximum output of hydro plants*
≥ *Demand*
\+ *Operating reserve* *for each load level and scenario*

$$\sum_t \overline{P}_t uc_{nt} + \sum_h \overline{P}_h \geq D_n + O_n \quad \forall n$$

Constraints: Generation and load balance

Generation of hydro and thermal units
+ Energy not served
= Demand for each load level and scenario

$$\sum_t p_{nt}^{\omega} + \sum_h p_{nh}^{\omega} + ig_n^{\omega} + ens_n^{\omega} = D_n \quad \forall \omega n$$

Constraints: Production in consecutive load levels

Unit output in any hour - Unit output in previous one ≤ ramp up

Unit output in any hour - Unit output in previous one ≥ – ramp down

$$p_{nt}^{\omega} - p_{n-1t}^{\omega} \leq RU_t \quad \forall \omega nt$$

$$p_{nt}^{\omega} - p_{n-1t}^{\omega} \geq -RD_t \quad \forall \omega nt$$

Constraints: Commitment, startup and shutdown

Commitment of a thermal unit in an hour
– *Commitment of a thermal unit in the previous hour*
= *Startup of a thermal unit in this hour*
– *Shutdown of a thermal unit in this hour*

$$uc_{nt} - uc_{n-1t} = su_{nt} - sd_{nt} \quad \forall nt$$

Constraints: Commitment and production

Production of a thermal unit on every scenario
≤ Commitment of a thermal unit x *the maximum output*

Production of a thermal unit on every scenario
≥ Commitment of a thermal unit x *the minimum output*

$$uc_{nt}\underline{P}_t \leq p_{nt}^{\omega} \leq uc_{nt}\overline{P}_t \quad \forall \omega nt$$

- If the thermal unit is committed ($uc_{nt} = 1$) it can produce between its minimum and maximum output
- If the thermal unit is not committed ($uc_{nt} = 0$) it can't produce

Constraints: Energy balance for each reservoir

Reservoir energy in hour $n-1$
– *Reservoir energy in hour* n
+ *Natural inflows*
– *Spillage from this reservoir*
– *Turbined energy from this reservoir* = 0 *for each reservoir, hour and scenario*

$$r^{\omega}_{n-1h} - r^{\omega}_{nh} + I_{nh} - s^{\omega}_{nh} - p^{\omega}_{nh} = 0 \qquad \forall \omega nh$$

Constraints: Operation limits

Power output between limits for each unit

$$0 \leq p_{nt}^{\omega} \leq \overline{P}_t \quad \forall \omega nt \qquad 0 \leq p_{nh}^{\omega} \leq \overline{P}_h \quad \forall \omega nh$$

Commitment, startup and shutdown for each unit

$$uc_{nt}, su_{nt}, sd_{nt} \in \{0,1\} \quad \forall nt$$

Intermittent generation limit

$$0 \leq ig_n^{\omega} \leq IG_n^{\omega} \quad \forall \omega n$$

Multiobjective function

- Minimize
  - Thermal unit expected variable costs (first stage + second stage)

$$\sum_{nt} CSU_t su_{nt} + \sum_{nt} CSD_t sd_{nt} + \sum_{nt} CF_t uc_{nt} +$$

$$\sum_{\omega nt} P^{\omega} CV_t p_{nt}^{\omega}$$

  - Expected penalty introduced in the objective function for energy not served

$$\sum_{\omega n} P^{\omega} CV' ens_n^{\omega}$$

Short Run Marginal Cost (SRMC)

- Short Run Marginal Cost = Dual variable of generation and load balance when binary variables (commitment, startup and shutdown) are fixed [€/MWh]
  - Change in the objective function due to a marginal increment in the demand

$$\sum_t p_{nt}^{\omega} + \sum_h p_{nh}^{\omega} + ig_n^{\omega} + ens_n^{\omega} = D_n \quad : \sigma_n^{\omega} \quad \forall \omega n$$

$$SRMC_n^{\omega} = \sigma_n^{\omega} \; / \; P^{\omega} \quad \forall \omega n$$

Dual variable = change in the objective function with respect to a marginal increase in the RHS of a constraint

StarGenLite_SDUC Stochastic Daily Unit Commitment Model (https://pascua.iit.comillas.edu/aramos/StarGenLite_SDUC.zip)

- Files
  - Microsoft Excel interface for input and output data StarGenLite_SDUC.xlsm
  - GAMS file StarGenLite_SDUC.gms
- How to use it
  - Save the Excel workbook if data have changed
  - Run the model [Run]
  - The model creates
    - tmp_StarGenLite_SDUC.xlsx with the output data and
    - StarGenLite_SDUC.lst as the listing file of the GAMS execution
  - Load the results into the Excel interface [Load results]

Prototype. Mathematical formulation

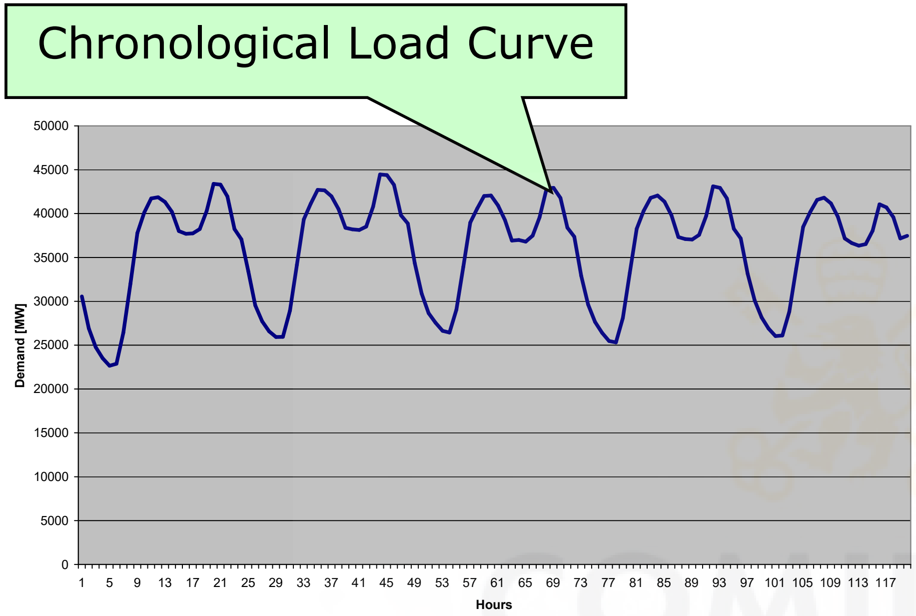
- Objective function
  - Minimize the total expected variable costs plus penalties for energy and power not served
- Variables
  - BINARY: Commitment, startup and shutdown of thermal units
  - Thermal, storage hydro and pumped storage hydro output
  - Reservoir levels
- Operation constraints
  - Inter-period
    - Storage hydro and pumped storage hydro scheduling
      Water balance with stochastic inflows
  - Intra-period
    - Load balance and operating reserve
    - Detailed hydro basin modeling
    - Thermal, storage hydro and pumped-storage hydro operation constraints
- Mixed integer linear programming (MIP)

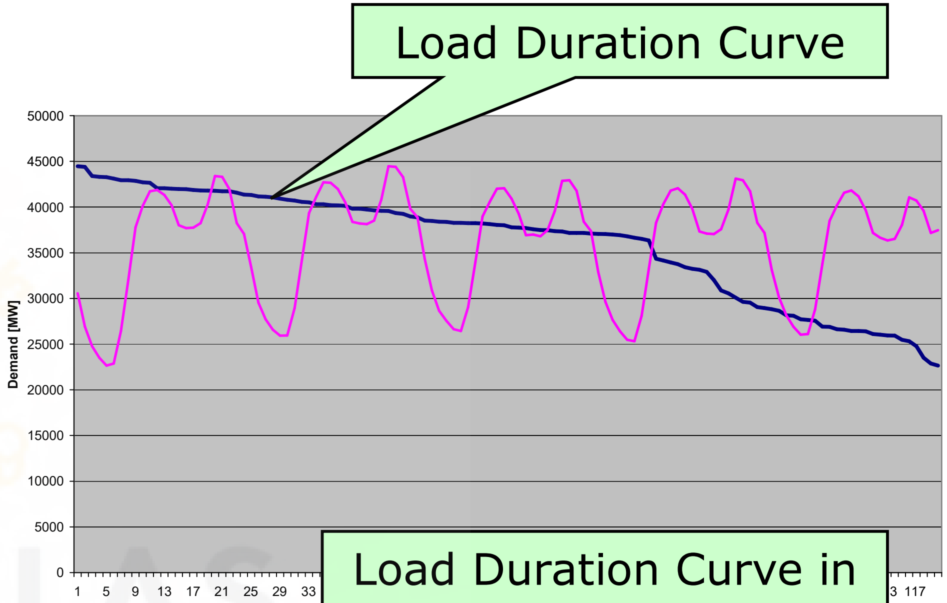
Indices

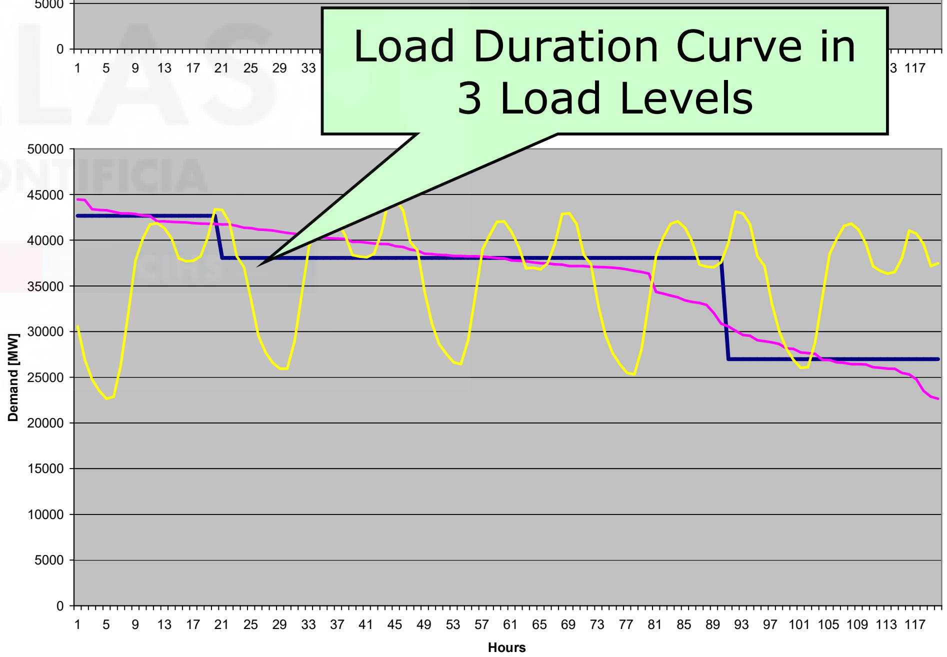
- Time scope
  - 1 year
- Period
  - 1 month
- Subperiod
  - weekdays and weekends
- Load level
  - peak, shoulder and off-peak

Period	p
Subperiod	s
Load level	n

Demand (5 weekdays)



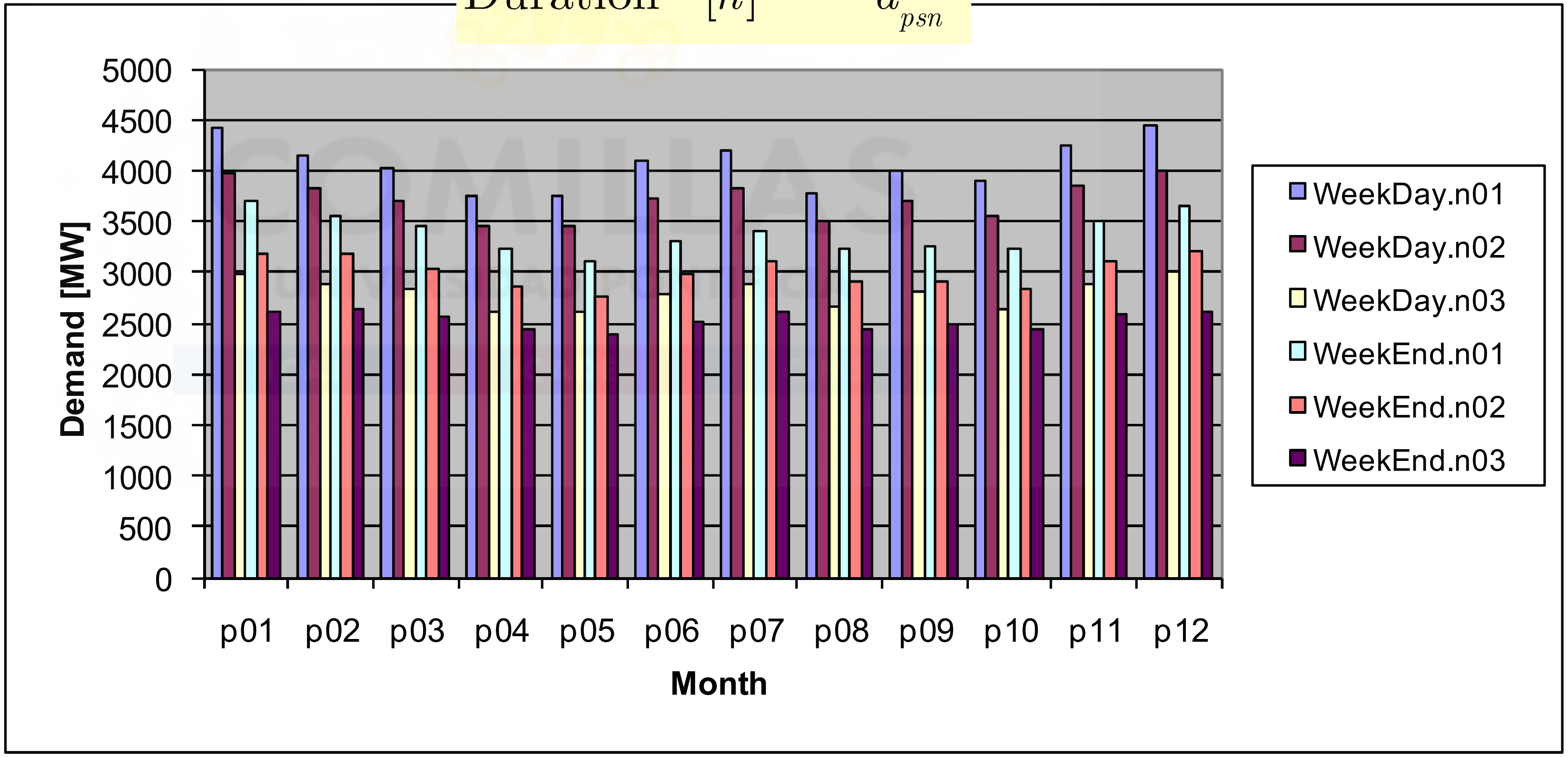




Demand

- Monthly demand with several load levels
  - Peak, shoulder and off-peak for weekdays and weekends
- All the weekdays of the same month are similar (same for weekends)

Demand	$[MW]$	D_{psn}
Duration	$[h]$	d_{psn}



Technical characteristics of thermal units (t)

- Maximum and minimum output
- Fuel cost
- Slope and intercept of the heat rate straight line
- Operation and maintenance (O&M) variable cost
  - No load cost = fuel cost x heat rate intercept
  - Variable cost = fuel cost x heat rate slope + O&M cost
- Cold startup and shutdown cost
- Equivalent forced outage rate (EFOR)

Max and min output	$[MW]$	$\overline{p}_t, \underline{p}_t$
No load cost	$[€ / h]$	f_t
Variable cost	$[€ / MWh]$	v_t
Startup, shutdown cost	$[€]$	su_t, sd_t
$EFOR$	$[p.u.]$	q_t

Technical characteristics of hydro plants (h)

- Maximum and minimum output
- Production function (efficiency for conversion of water release in m^3/s to electric power MW)
- Efficiency of pumped storage hydro plants
  - Only this ratio of the energy consumed to pump the water is recovered by turbining this water

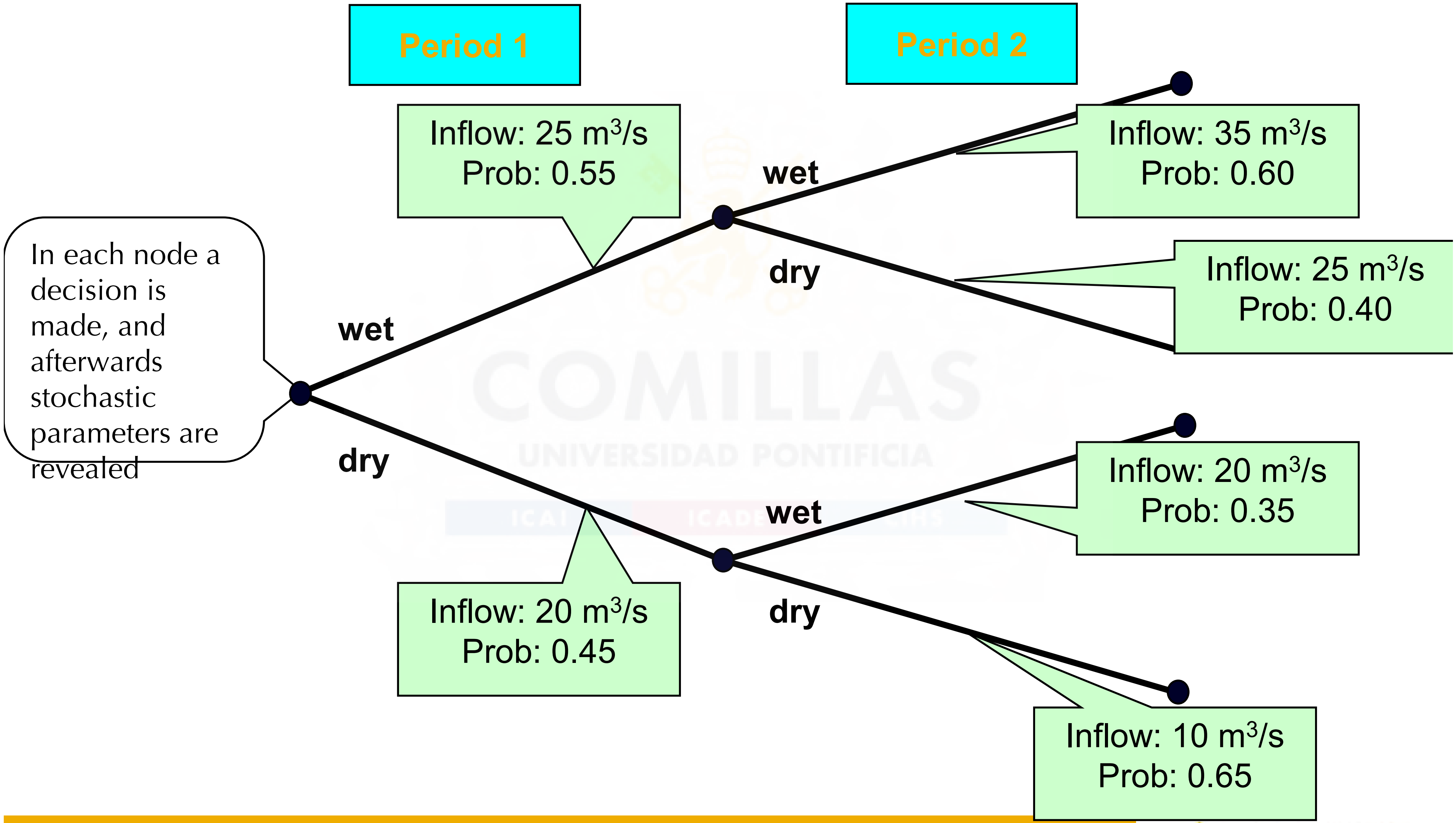
Max and min output	$[MW]$	$\overline{p}_h, \underline{p}_h$
Production function	$[kWh / m^3]$	c_h
Efficiency	$[p.u.]$	η_h

Technical characteristics of hydro reservoirs (*r*)

- Maximum and minimum water reserve
- Initial water reserve
  - Final water reserve = initial water reserve
- Stochastic inflows

Max and min reserve	$[hm^3]$	$\overline{r}_r, \underline{r}_r$
Initial and final reserve	$[hm^3]$	r'_r
Stochastic inflows	$[m^3/s]$	i^{ω}_{pr}

Scenario tree example



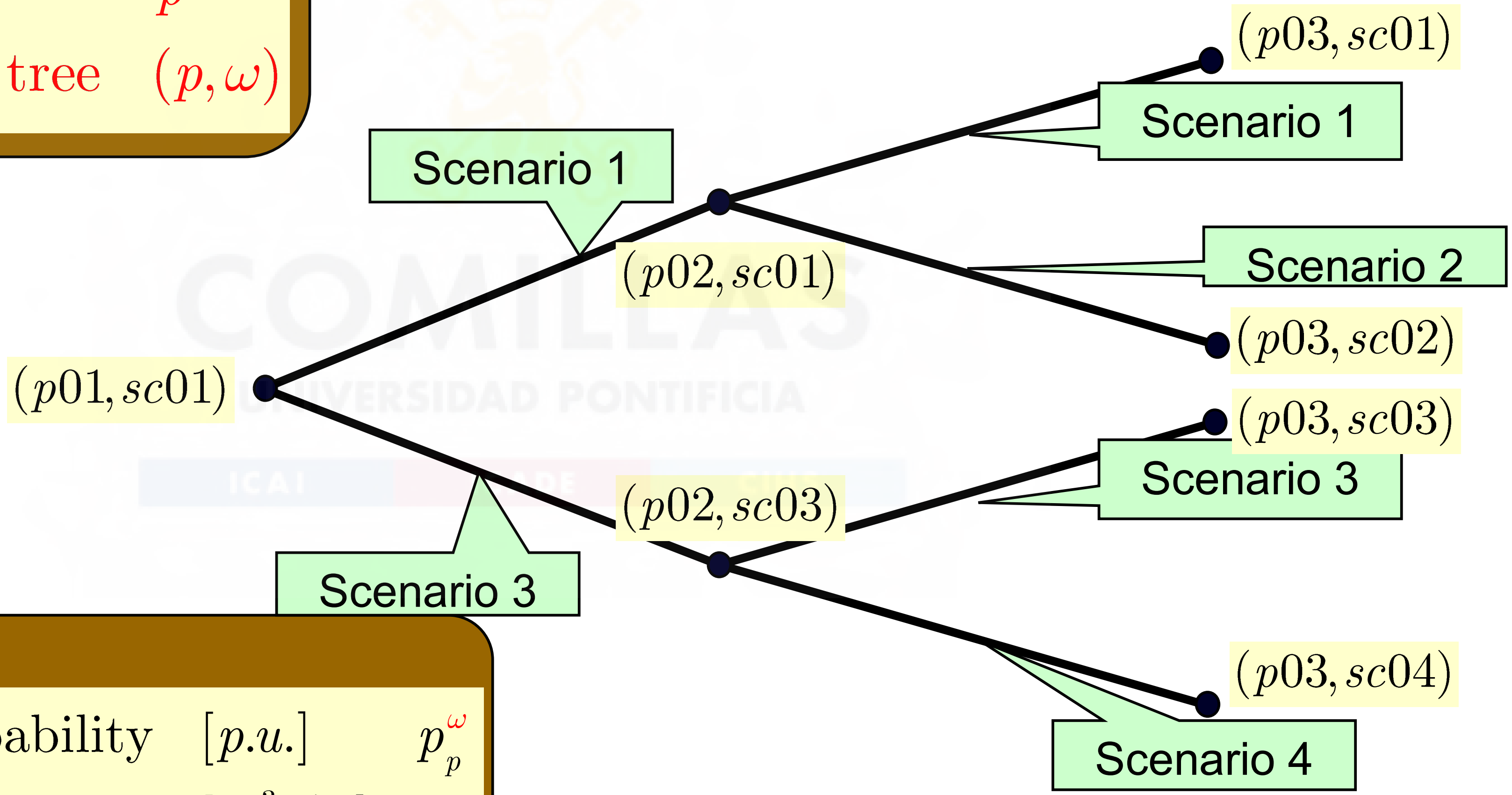
Scenario tree. Ancestor and descendant

Tree structure

Scenario	ω
Period	p
Scenario tree	(p,ω)

Tree relations

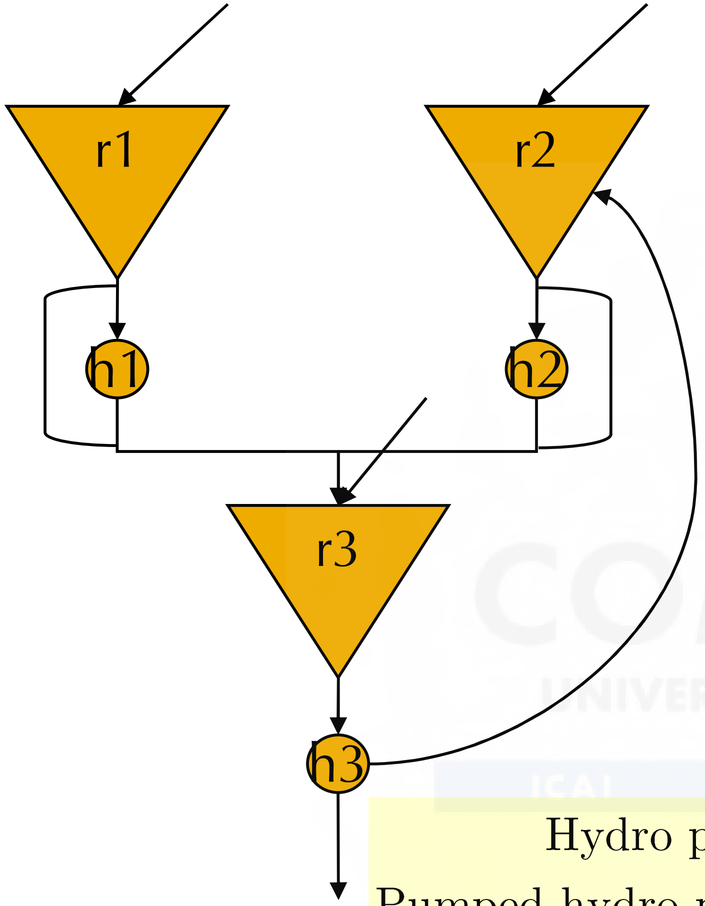
$$\omega' \in a(\omega) \quad (p02,sc03) \in a\left[(p03,sc03)\right]$$



Tree data

Scenario probability	$[p.u.]$	p_p^{ω}
Stochastic inflows	$[m^3/s]$	i_{pr}^{ω}

Hydro topology



Only one spillage per reservoir can be considered

	Written in Math	Written in GAMS	Example
Hydro plant upstream of reservoir	$h \in up(r)$	$hur(h,r)$	$(h1, r3)$
Pumped hydro plant upstream of reservoir	$h \in up(r)$	$hpr(h,r)$	$(h3, r2)$
Reservoir upstream of hydro plant	$h \in dw(r)$	$ruh(r,h)$	$(r2, h2)$
Reservoir upstream of pumped hydro plant	$h \in dw(r)$	$rph(r,h)$	$(r3, h3)$
Reservoir upstream of reservoir	$r' \in up(r)$	$rur(r,r)$	$(r1, r3)$

Other system parameters

- Energy not served cost
- Operating power reserve not served cost
- Operating power reserve

Energy not served cost	[€ / MWh]	v'
Operating power reserve not served cost	[€ / MW]	v''
Operating reserve	[MW]	O_{ps1}

Variables

- Commitment, startup and shutdown of thermal units (BINARY)

$$\text{Commitment, startup and shutdown} \quad \{0,1\} \quad UC_{pst}^{\omega}, SU_{pst}^{\omega}, SD_{pst}^{\omega}$$

- Production of thermal units and hydro plants

$$\text{Production of a thermal or hydro unit} \quad [MW] \quad P_{psnt}^{\omega}, P_{psnh}^{\omega}$$

- Consumption of pumped storage hydro plants

$$\text{Consumption of a hydro plant} \quad [MW] \quad C_{psnh}^{\omega}$$

- Reservoir levels (at the beginning of the period)

$$\text{Reservoir level} \quad [hm^3] \quad R_{pr}^{\omega}$$

- Energy and power not served

$$\text{Energy and power not served} \quad [MW] \quad ENS_{psn}^{\omega}, PNS_{ps}^{\omega}$$

Constraints: Operating power reserve

  Committed output of thermal units
+ Maximum output of hydro plants
+ Power not served
≥ Demand
+ Operating reserve *for peak load level, subperiod, period and scenario*

$$\sum_t \overline{p}_t UC^{\omega}_{pst} + \sum_h \overline{p}_h + PNS^{\omega}_{ps} \geq D_{ps1} + O_{ps1} \quad \forall \omega ps$$

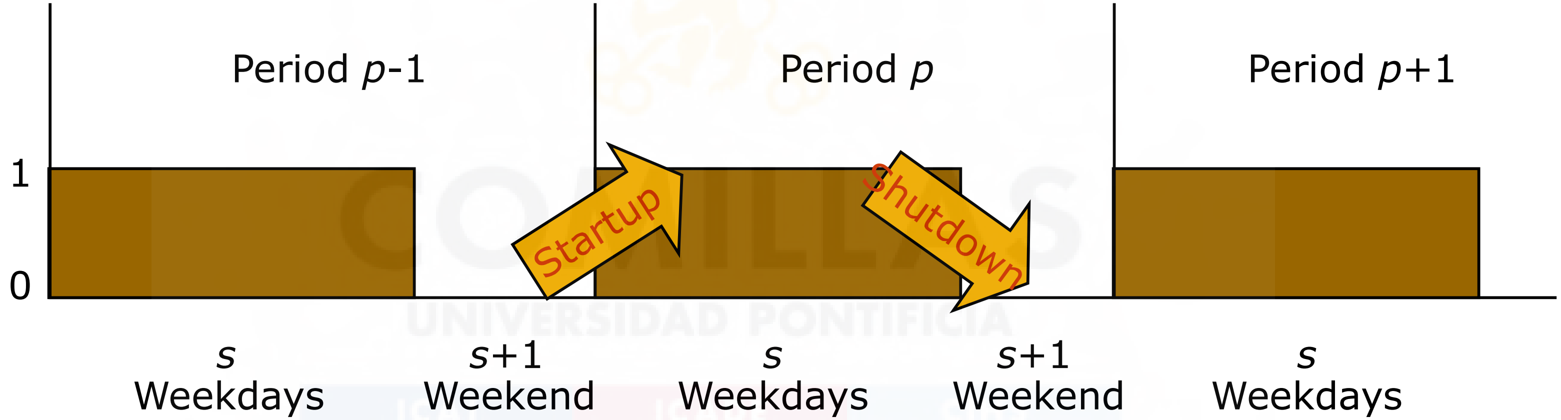
Constraints: Generation and load balance

Generation of thermal units
+ Generation of storage hydro plants
– Consumption of pumped storage hydro plants
+ Energy not served
= Demand for each load level, subperiod, period and scenario

$$\sum_t P^{\omega}_{psnt} + \sum_h P^{\omega}_{psnh} - \sum_h C^{\omega}_{psnh} + ENS^{\omega}_{psn} = D_{psn} \qquad \forall \omega psn$$

Constraints: Commitment, startup and shutdown

- All the weekdays of the same month are similar (same for weekends)
- Commitment decision of a thermal unit



Constraints: Commitment, startup and shutdown

- Startup of thermal units can only be made in the transition between consecutive weekend and weekdays

Commitment of a thermal unit in a weekday
– Commitment of a thermal unit in the weekend of previous period
= Startup of a thermal unit in this weekday
– Startup of a thermal unit in this weekday

$$UC^{\omega}_{pst} - UC^{\omega'}_{p-1s+1t} = SU^{\omega}_{pst} - SD^{\omega}_{pst} \quad \forall \omega pst \quad \omega' \in a(\omega)$$

- Shutdown only in the opposite transition

Commitment of a thermal unit in a weekend
– Commitment of a thermal unit in the previous weekday
= Startup of a thermal unit in this weekend
– Shutdown of a thermal unit in this weekend

$$UC^{\omega}_{ps+1t} - UC^{\omega}_{pst} = SU^{\omega}_{ps+1t} - SD^{\omega}_{ps+1t} \quad \forall \omega pst$$

Constraints: Commitment and production

Production of a thermal unit
≤ Commitment of a thermal unit x *the maximum output reduced by availability rate*

Production of a thermal unit
≥ Commitment of a thermal unit x *the minimum output reduced by availability rate*

$$UC^{\omega}_{pst}\underline{p}_t(1-q_t) \le P^{\omega}_{psnt} \le UC^{\omega}_{pst}\overline{p}_t(1-q_t) \qquad \forall \omega psnt$$

- If the thermal unit is committed ($UC^{\omega}_{pst} = 1$) it can produce between its minimum and maximum output
- If the thermal unit is not committed ($UC^{\omega}_{pst} = 0$) it can't produce

Constraints: Water balance for each reservoir

Reservoir volume at the beginning of the period
– Reservoir volume at the end of the period
+ Natural hydro inflows
– Spills from this reservoir
+ Spills from upstream reservoirs
+ Turbined water from upstream storage hydro plants
– Turbined and pumped water from this reservoir
+ Pumped water from upstream pumped hydro plants = 0 *for each reservoir, period and scenario*

$$R^{\omega'}_{p-1r} - R^{\omega}_{pr} + i^{\omega}_{pr} - S^{\omega}_{pr} + \sum_{r' \in up(r)} S^{\omega}_{pr'}$$
$$+ \sum_{\substack{sn \\ h \in up(r)}} d_{psn} P^{\omega}_{psnh} / c_h - \sum_{\substack{sn \\ h \in dw(r)}} d_{psn} P^{\omega}_{psnh} / c_h$$
$$+ \sum_{\substack{sn \\ h \in up(r)}} d_{psn} C^{\omega}_{psnh} \eta_h / c_h - \sum_{\substack{sn \\ h \in dw(r)}} d_{psn} C^{\omega}_{psnh} \eta_h / c_h = 0 \qquad \forall \omega pr \quad \omega' \in a(\omega)$$

Constraints: Operation limits

Reservoir volumes between limits for each hydro reservoir

$$\underline{r}_r \leq R^{\omega}_{pr} \leq \overline{r}_r \qquad \forall \omega pr$$
$$R_{0r} = R^{\omega}_{Pr} = r'_r \quad \forall \omega r$$

Power output between limits for each unit

$$0 \leq P^{\omega}_{psnt} \leq \overline{p}_t(1 - q_t) \quad \forall \omega psnt$$
$$0 \leq P^{\omega}_{psnh}, C^{\omega}_{psnh} \leq \overline{p}_h \quad \forall \omega psnh$$

Commitment, startup and shutdown for each unit

$$UC^{\omega}_{pst}, SU^{\omega}_{pst}, SD^{\omega}_{pst} \in \{0,1\} \quad \forall \omega pst$$

Multiobjective function

- Minimize
  - Expected thermal variable costs

$$\sum_{\omega pst} p_p^\omega su_t SU_{pst}^\omega + \sum_{\omega pst} p_p^\omega sd_t SD_{pst}^\omega + \sum_{\omega psnt} p_p^\omega d_{psn} f_t UC_{pst}^\omega + \sum_{\omega psnt} p_p^\omega d_{psn} v_t P_{psnt}^\omega$$

  - Expected penalties introduced in the objective function for energy and power not served

$$\sum_{\omega psn} p_p^\omega d_{psn} v' ENS_{psn}^\omega + \sum_{\omega ps} p_p^\omega v'' PNS_{ps}^\omega$$

Short Run Marginal Cost (*SRMC*)

- Dual variable of generation and load balance [€/MW]
  - Change in the objective function due to a marginal increment in the demand when binary variables (commitment, startup and shutdown) are fixed

$$\sum_t P^{\omega}_{psnt} + \sum_h P^{\omega}_{psnh} - \sum_h C^{\omega}_{psnh} + ENS^{\omega}_{psn} = D_{psn} \quad : \sigma^{\omega}_{psn} \quad \forall \omega psn$$

- Short Run Marginal Cost = dual variable / load level duration / scenario probability. Expressed in [€/MWh]

$$SRMC^{\omega}_{psn} = \sigma^{\omega}_{psn} \,/\, d_{psn} \,/\, p^{\omega}_{p} \quad \forall \omega psn$$

Water value

- Dual variable of water balance for each reservoir [€/hm^3]
  - Change in the objective function due to a marginal increment in the reservoir inflow

$$R^{\omega'}_{p-1r} - R^{\omega}_{pr} + i^{\omega}_{pr} - S^{\omega}_{pr} + \sum_{r' \in up(r)} S^{\omega}_{pr'}$$
$$+ \sum_{\substack{sn \\ h \in up(r)}} d_{psn} P^{\omega}_{psnh} / c_h - \sum_{\substack{sn \\ h \in dw(r)}} d_{psn} P^{\omega}_{psnh} / c_h$$
$$+ \sum_{\substack{sn \\ h \in up(r)}} d_{psn} C^{\omega}_{psnh} \eta_h / c_h - \sum_{\substack{sn \\ h \in dw(r)}} d_{psn} C^{\omega}_{psnh} \eta_h / c_h = 0 \quad : \pi^{\omega}_{pr} \qquad \forall \omega pr \qquad \omega' \in a(\omega)$$

- Turbining water has no variable cost. However, an additional hm^3 turbined allows to substitute energy produced by thermal units with the corresponding variable cost (this is called water value)

- Files
  - Microsoft Excel interface for input and output data StarGenLite_SHTCM.xlsm
  - GAMS file StarGenLite_SHTCM.gms
- How to use it
  - Save the Excel workbook if data have changed
  - Run the model [Run]
  - The model creates
    - tmp_StarGenLite_SHTCM.xlsx with the output data and
    - StarGenLite_SHTCM.lst as the listing file of the GAMS execution
  - Load the results into the Excel interface

[Load results]

3

Two-stage and multistage programming

Two-stage (PL-2) and multistage (PL-P) linear programming

- Two-stage PL-2: decisions in two stages
- Multistage PL-P: decisions in multiple stages

- Stairway structure of the constraint matrix (block diagonal)
  - Each stage is only related to the previous one

- Problems of each stage are similar (they have the same structure)
- The matrix structure can be detected by visual inspection

Two-stage linear programming PL-2

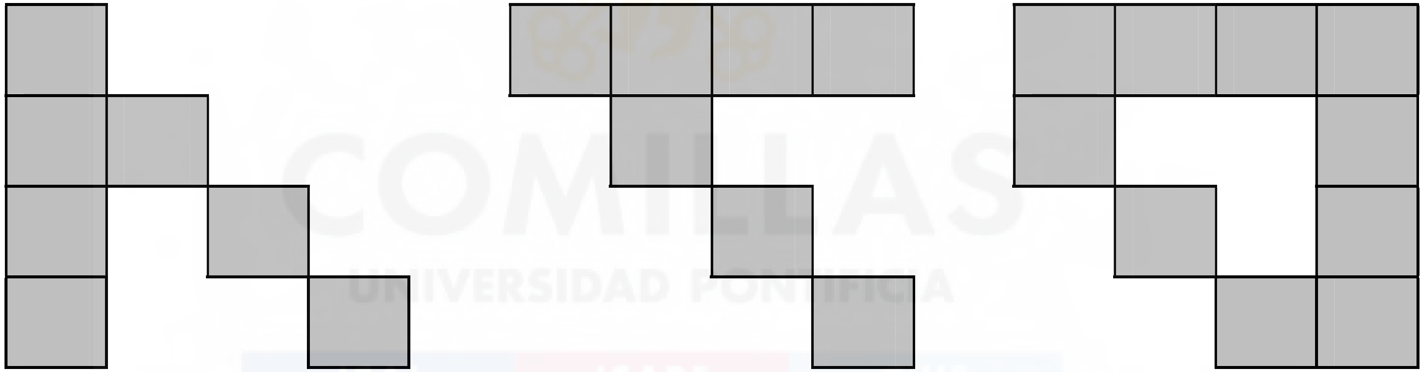
$$
\begin{array}{ll}
c_1 \in \mathbb{R}^{n_1} & c_2 \in \mathbb{R}^{n_2} \\
A_1 \in \mathbb{R}^{m_1 \times n_1} & A_2 \in \mathbb{R}^{m_2 \times n_2} \\
b_1 \in \mathbb{R}^{m_1} & b_2 \in \mathbb{R}^{m_2} \\
x_1 \in \mathbb{R}^{n_1} & x_2 \in \mathbb{R}^{n_2}
\end{array}
$$

$$
\begin{array}{|c|c|}
\hline
A_1 & \\
\hline
B_1 & A_2 \\
\hline
\end{array}
$$

$$
\begin{array}{llll}
\min\limits_{x_1, x_2} (c_1^T x_1 + c_2^T x_2) & & & \\
A_1 x_1 & & = & b_1 \\
B_1 x_1 & + A_2 x_2 & = & b_2 \\
x_1, & x_2 & \geq & 0
\end{array}
$$

Structures of the constraint matrix

- Block diagonal with variables that complicate, constraints or both



Two-stage stochastic linear programming PLE-2

- O.F. minimizes first-stage costs and expected value of second-stage costs

$$
\begin{aligned}
\min_{x_1, x_2^\omega} \; & c_1^T x_1 + \sum_{\omega \in \Omega} p^\omega c_2^{\omega T} x_2^\omega \\
& A_1 x_1 && = b_1 \\
& B_1^\omega x_1 + A_2^\omega x_2^\omega && = b_2^\omega \\
& x_1, \quad x_2^\omega && \geq 0
\end{aligned}
$$

- If A_2^ω doesn't depend on ω it is called fixed resource
- Structure of the constraint matrix

A_1			
B_1^1	A_2^1		
B_1^2		A_2^2	
B_1^3			A_2^3

Deterministic equivalent problem (DEP)

- State space is finite (and small)
- Formulation of the deterministic equivalent problem

$$
\begin{array}{lllll}
\min\limits_{x_1, x_2^{\omega_1}, x_2^{\omega_2}, x_2^{\omega_3}} & c_1^T x_1 + p^{\omega_1} c_2^{\omega_1 T} x_2^{\omega_1} & + p^{\omega_2} c_2^{\omega_2 T} x_2^{\omega_2} & + p^{\omega_3} c_2^{\omega_3 T} x_2^{\omega_3} & \\
A_1 x_1 & & & & = b_1 \\
B_1^{\omega_1} x_1 & + A_2^{\omega_1} x_2^{\omega_1} & & & = b_2^{\omega_1} \\
B_1^{\omega_2} x_1 & & + A_2^{\omega_2} x_2^{\omega_2} & & = b_2^{\omega_2} \\
B_1^{\omega_3} x_1 & & & + A_2^{\omega_3} x_2^{\omega_3} & = b_2^{\omega_3} \\
x_1, & x_2^{\omega_1}, & x_2^{\omega_2}, & x_2^{\omega_3}, & \geq 0
\end{array}
$$

https://www.gams.com/latest/docs/S_DE.html

Two-stage stochastic linear programming PLE-2 (ii)

- Minimization of the maximum regret

$$
\begin{aligned}
&\min_{\alpha, x_1, x_2^\omega} \alpha \\
&\alpha \quad -c_1^T x_1 \quad -c_2^{\omega T} x_2^\omega \quad \geq -f^\omega \quad \forall \omega \in \Omega \\
&\qquad A_1 x_1 \qquad\qquad\qquad = b_1 \\
&\qquad B_1^\omega x_1 \quad +A_2^\omega x_2^\omega \quad = b_2^\omega \\
&\qquad x_1, \qquad\quad x_2^\omega \qquad \geq 0
\end{aligned}
$$

f^ω o.f. with perfect information for scenario ω

- Minimization of the maximum cost

$$
\begin{aligned}
&\min_{\alpha, x_1, x_2^\omega} \alpha \\
&\alpha \quad -c_1^T x_1 \quad -c_2^{\omega T} x_2^\omega \quad \geq 0 \quad \forall \omega \in \Omega \\
&\qquad A_1 x_1 \qquad\qquad\qquad = b_1 \\
&\qquad B_1^\omega x_1 \quad +A_2^\omega x_2^\omega \quad = b_2^\omega \\
&\qquad x_1, \qquad\quad x_2^\omega \qquad \geq 0
\end{aligned}
$$

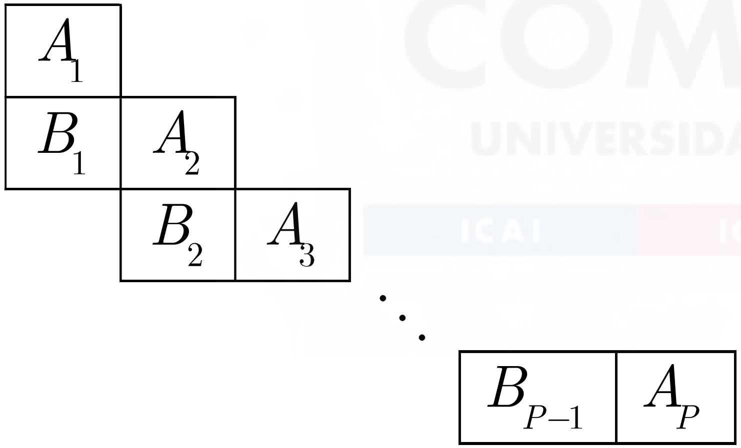
Multistage linear programming PL-P

$c_p \in \mathbb{R}^{n_p}$

$A_p \in \mathbb{R}^{m_p \times n_p}$

$b_p \in \mathbb{R}^{m_p}$

$x_p \in \mathbb{R}^{n_p}$



$$\min_{x_p} \sum_{p=1}^{P} c_p^T x_p$$

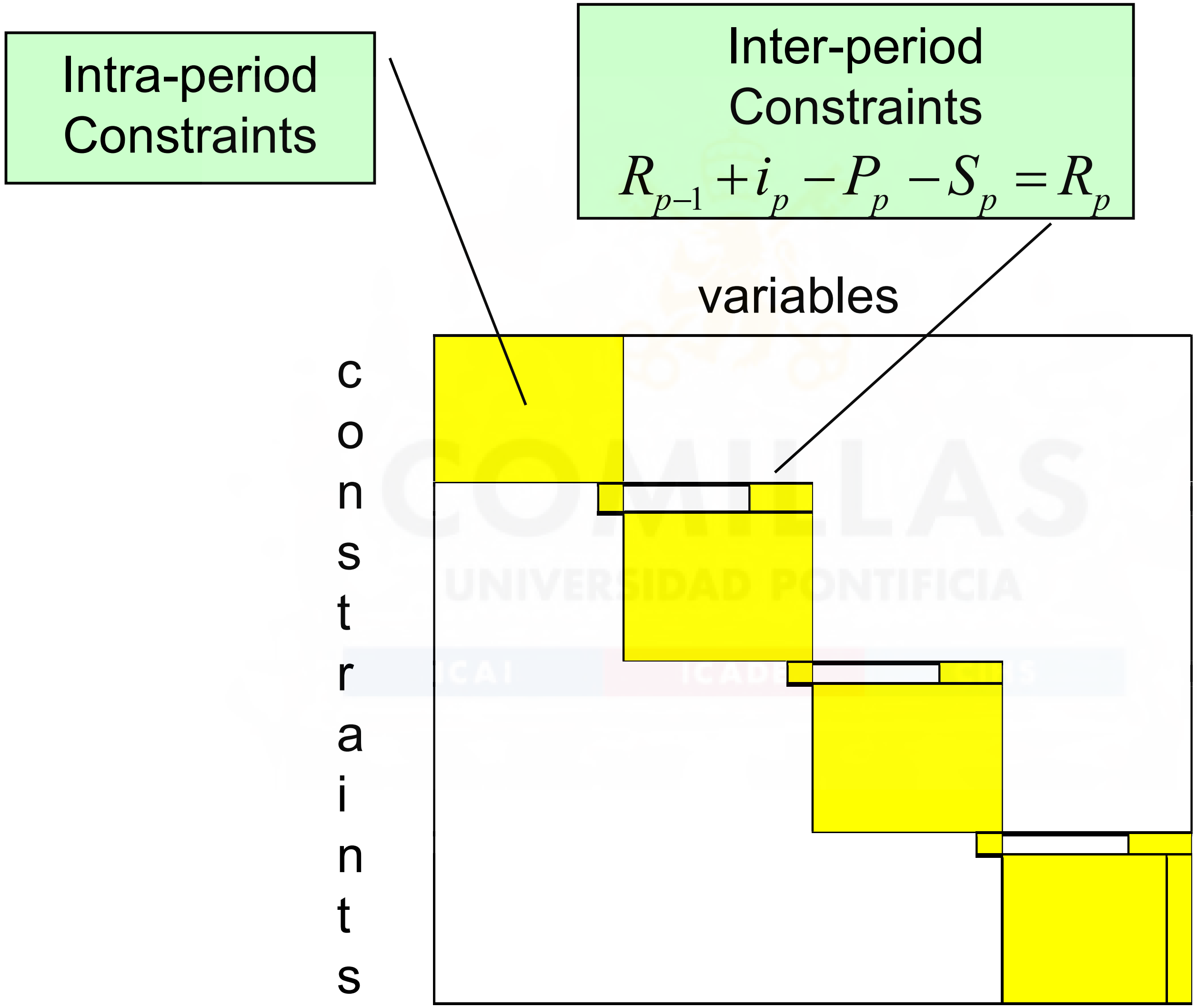
$$B_{p-1} x_{p-1} + A_p x_p = b_p \quad p = 1, \ldots, P$$

$$x_p \geq 0$$

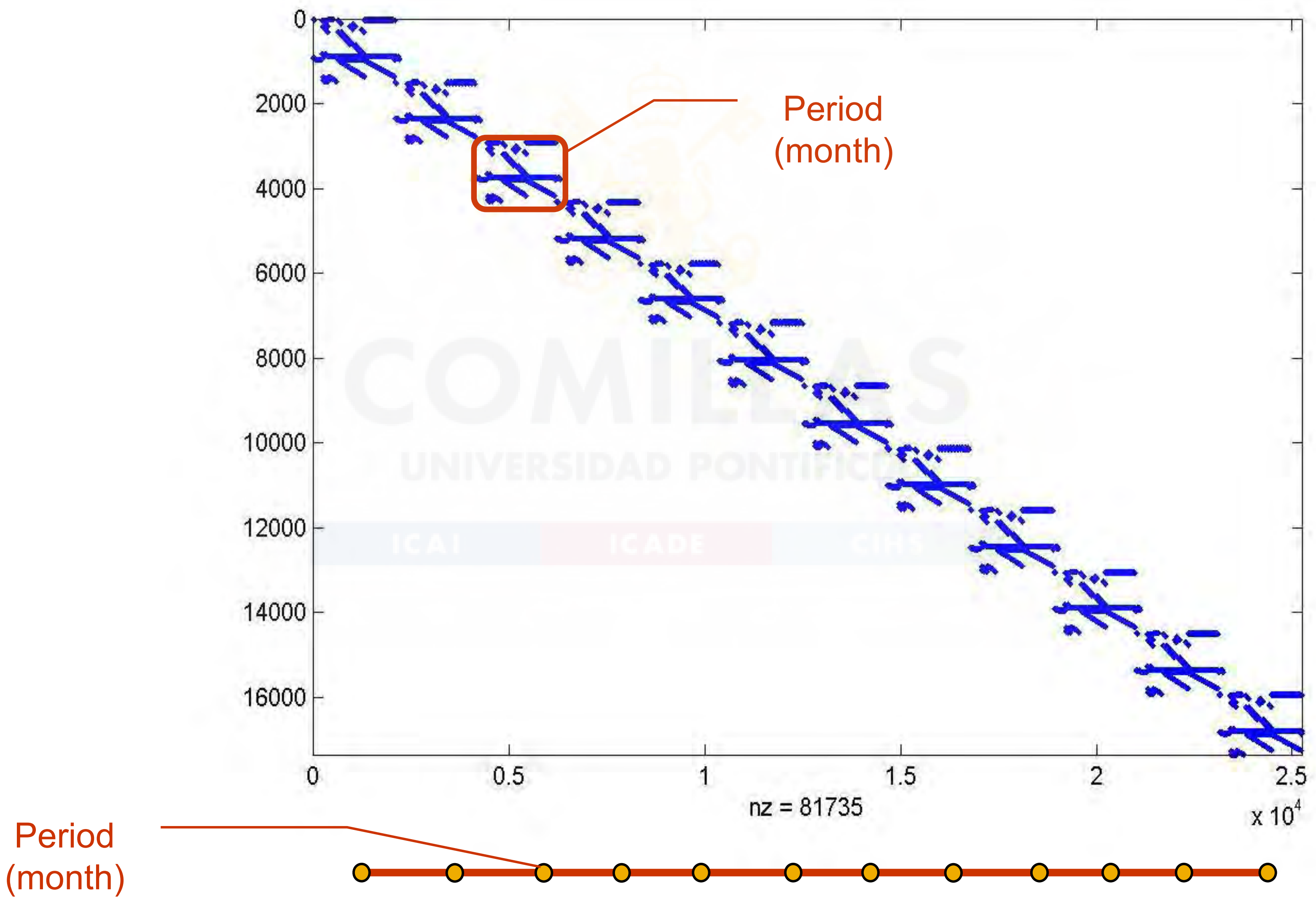
$$B_0 \equiv 0$$

Medium term hydrothermal scheduling problem: constraint matrix

R_{p-1}	reservoir level
i_p	inflow
P_p	hydro output
S_p	reservoir spillage



Medium term hydrothermal scheduling problem: constraint matrix



Multistage stochastic linear programming PLE-P

- O. F. minimizes expected costs of all the stages

$$
\begin{aligned}
&\min_{x_p^{\omega_p}} \sum_{p=1}^{P} \sum_{\omega_p \in \Omega_p} p_p^{\omega_p} c_p^{\omega_p T} x_p^{\omega_p} \\
&B_{p-1}^{\omega_p} x_{p-1}^{\omega_{p-1}} + A_p^{\omega_p} x_p^{\omega_p} = b_p^{\omega_p} \qquad p = 1,\dots,P \\
&x_p^{\omega_p} \ge 0 \\
&B_0^{\omega_1} \equiv 0
\end{aligned}
$$

- Size grows exponentially with the number of scenarios
- Probabilities $p_p^{\omega_p}$ are conditional
- Constraint matrix

$$
\begin{array}{|c|c|c|c|c|c|c|}
\hline
A_1 & & & & & & \\
B_1^1 & A_2^1 & & & & & \\
B_1^2 & & A_2^2 & & & & \\
 & B_2^1 & & A_3^1 & & & \\
 & B_2^2 & & & A_3^2 & & \\
 & & B_2^3 & & & A_3^3 & \\
 & & B_2^4 & & & & A_3^4 \\
\hline
\end{array}
$$

4

1. General overview
2. Applications in electric systems
3. Two-stage and multistage programming
4. **Decomposition techniques**
5. Benders decomposition
6. Nested Benders decomposition
7. Dantzig-Wolfe decomposition
8. Lagrangian relaxation
9. Scenario tree
10. Decomposition in two-stage and multistage stochastic programming
11. Improvements in decomposition techniques
12. Simulation in stochastic optimization
13. Stochastic dual dynamic programming

Decomposition techniques

Decomposition techniques

- Divide and conquer strategy
  - Time division by periods
  - Spatial division by (thermal) units
  - Division by scenarios
- Allow the solution of very large-scale problems (not directly solvable) with a specific structure by solving iteratively small-size problems
- Objective function and feasible region must be convex whenever obtaining dual variables is needed
- Algorithms for solving any optimization problem
- Dantzig-Wolfe 1960, Benders 1962, Geoffrion 1972 (generalized Benders)

Decomposition techniques: classification

- According to the difficulties
  - Variables (Benders)
  - Constraints (Dantzig-Wolfe or Lagrangian relaxation)

- According to the exchanged information from the master to the subproblem
  - Primal (Benders)
  - Dual (Dantzig-Wolfe or Lagrangian relaxation)

Coordinating mechanisms. Hydrothermal model

- Primal (quantities)
  - Master assigns an amount of outflow to release in each period or the reservoir levels at the end of the period
  - Each subproblem returns the marginal price (water value) associated with the use of the previous amount of water

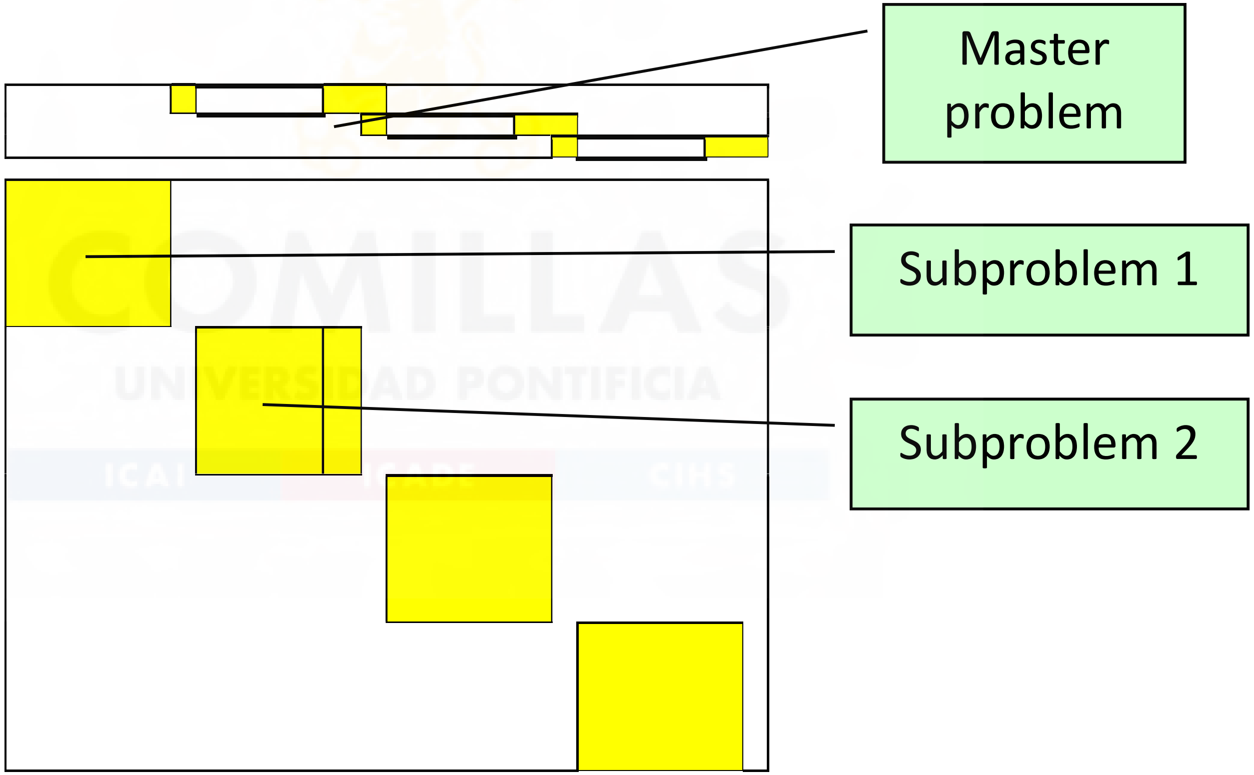
- Dual (prices)
  - Master gives a value to the water
  - Each subproblem returns the future cost function considering this value

Medium term hydrothermal scheduling problem

- Solvable by Bd, DW-LR, or nested Benders decomposition
  - *Variables* of hydro outflow *complicate* the solution ⇒ Benders
  - *Constraints* of hydro outflow *complicate* the solution ⇒ Dantzig-Wolfe, Lagrangian relaxation
- Criterion:
  - Engineering: context dependent
  - Mathematical:
    - What complicates? (foreseeable number of iterations)
    - Respective size of master and subproblems

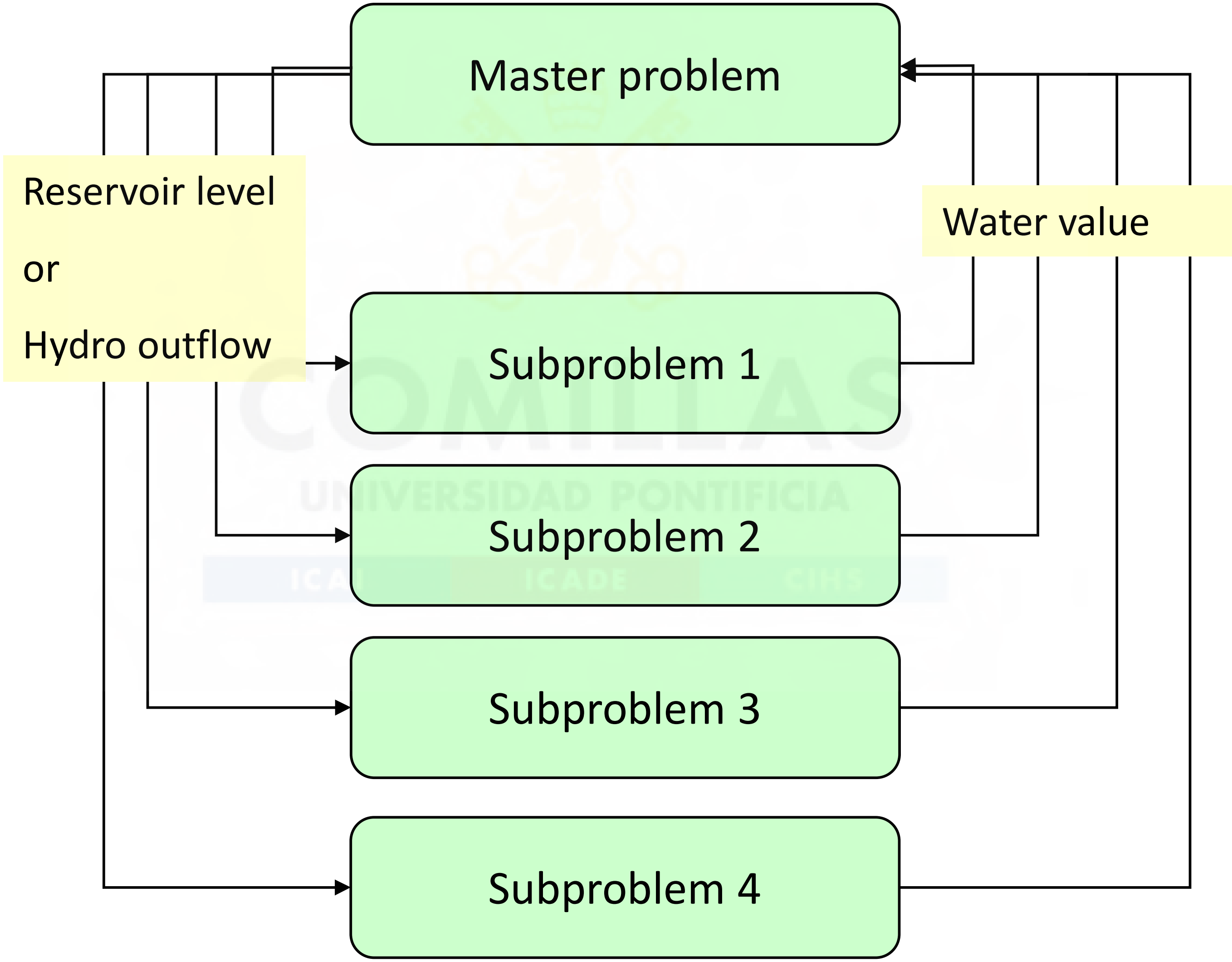
Algorithm: Benders

- Master problem: inter-period constraints
- Subproblem: intra-period constraints



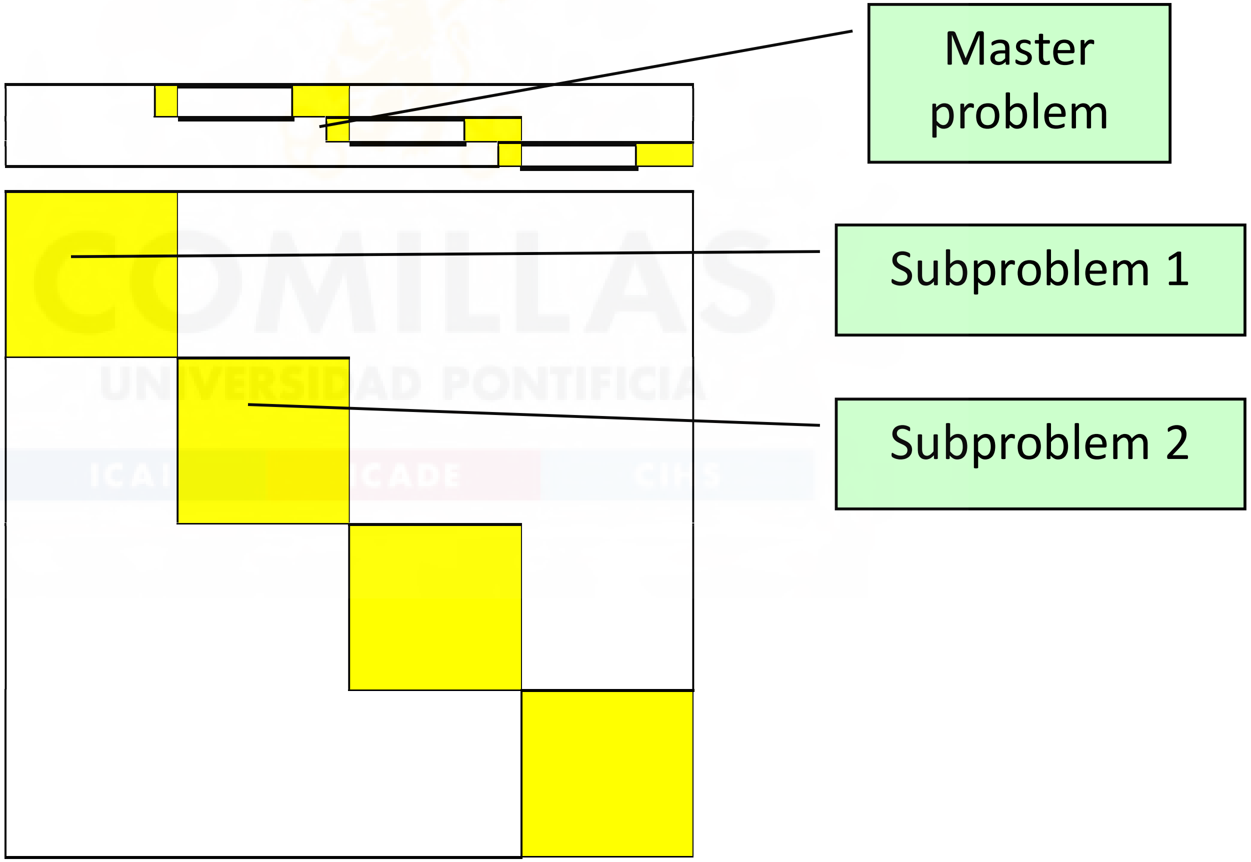
Benders decomposition

- Reservoir level or hydro outflow is a given for the subproblem



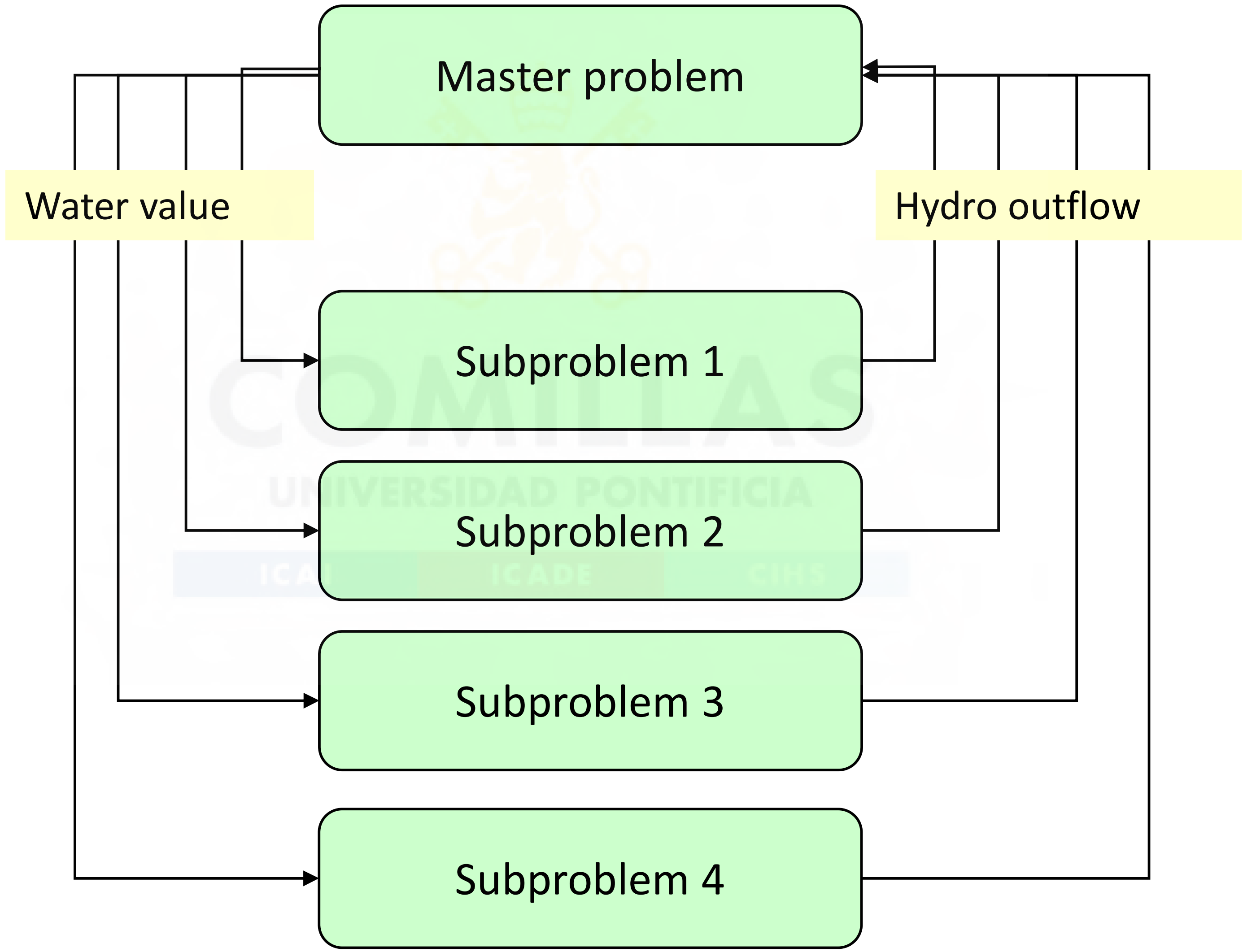
Algorithm: Dantzig-Wolfe or Lagrangian relaxation

- Master problem: inter-period constraints
- Subproblem: intra-period constraints

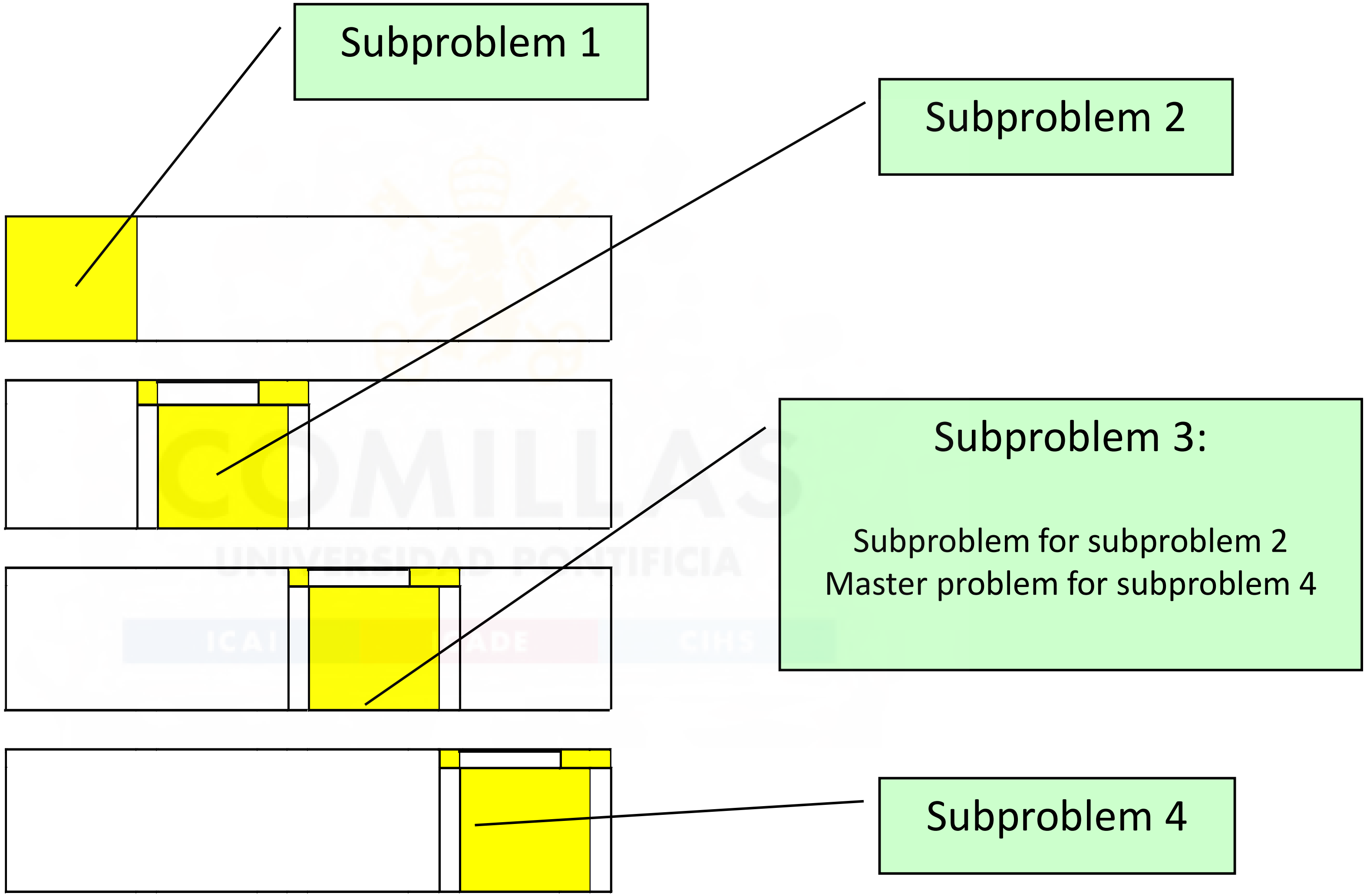


DW or LR decomposition

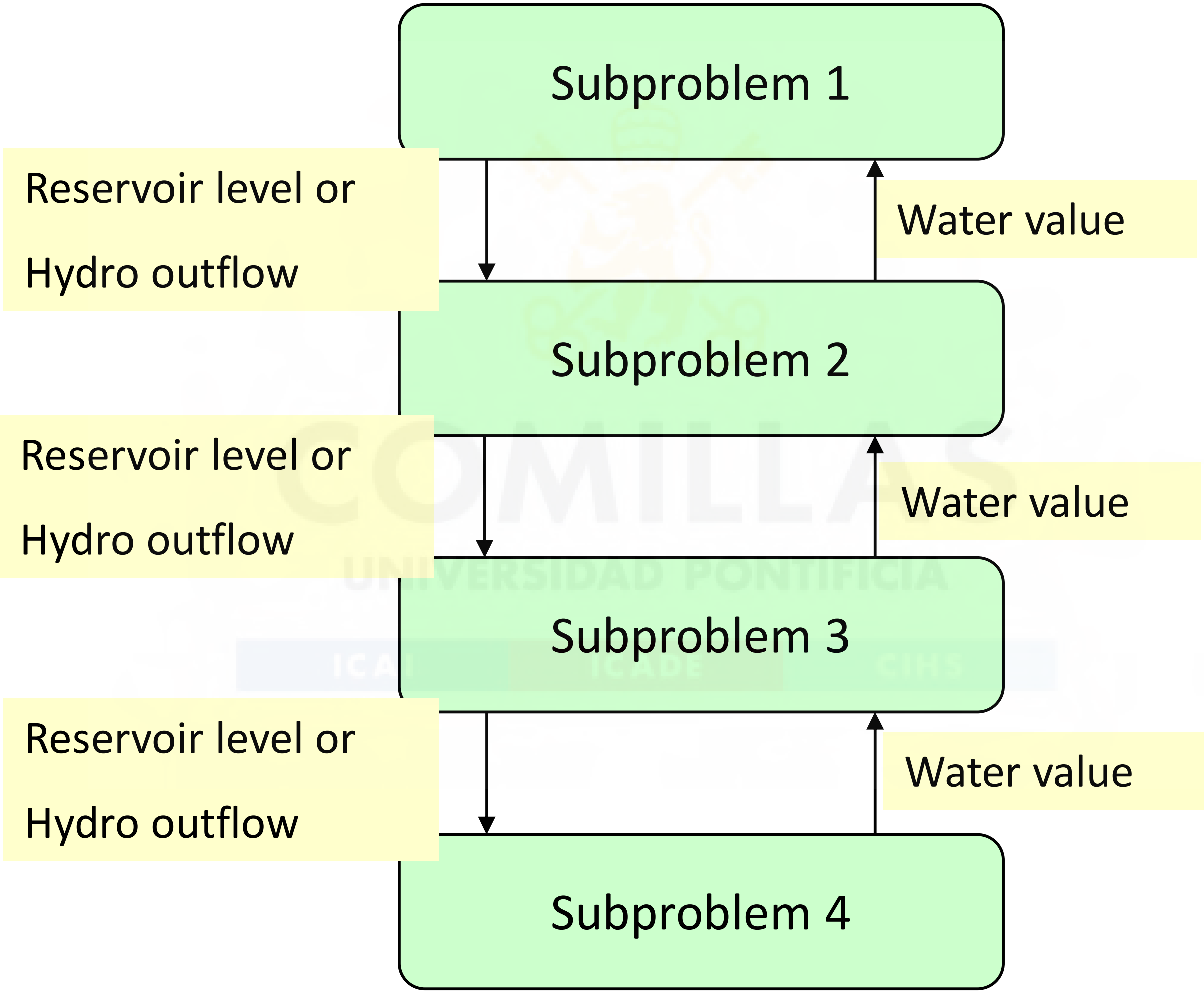
- Reservoir level is a variable for the subproblem



Algorithm: nested Benders decomposition



Nested Benders decomposition



Computer applications

- Currently there are no standard solvers, powerful and stable for stochastic problems. In many cases

  Stochastic optimization = "do-it-yourself"

- Solver with decomposition methods (https://www.stoprog.org/resources)
  - MSLiP (Horand I. Gassman)
  - SLP-IOR (Janos Mayer)
  - FortSP (OptiRisk Systems)
  - SPiNE (Gautam Mitra)
  - DDSIP (Claus C. Carøe)
  - Python-based Stochastic Programming PySP (Jean-Paul Watson)
  - Bouncing Nested Benders Solvers BNBS (Fredrik Altenstedt)
  - Stochastic Modeling Interface COIN-SMI (open-source interface for modeling stochastic linear programming problems)
- ILOG Concert + C++
- GAMS o AMPL + decomposition methods
- GAMS EMP (https://www.gams.com/latest/docs/UG_EMP.html)
- GAMS/DECIS for two-stage stochastic programming

5

1. General overview
2. Applications in electric systems
3. Two-stage and multistage programming
4. Decomposition techniques
5. **Benders decomposition**
6. Nested Benders decomposition
7. Dantzig-Wolfe decomposition
8. Lagrangian relaxation
9. Scenario tree
10. Decomposition in two-stage and multistage stochastic programming
11. Improvements in decomposition techniques
12. Simulation in stochastic optimization
13. Stochastic dual dynamic programming

Benders decomposition

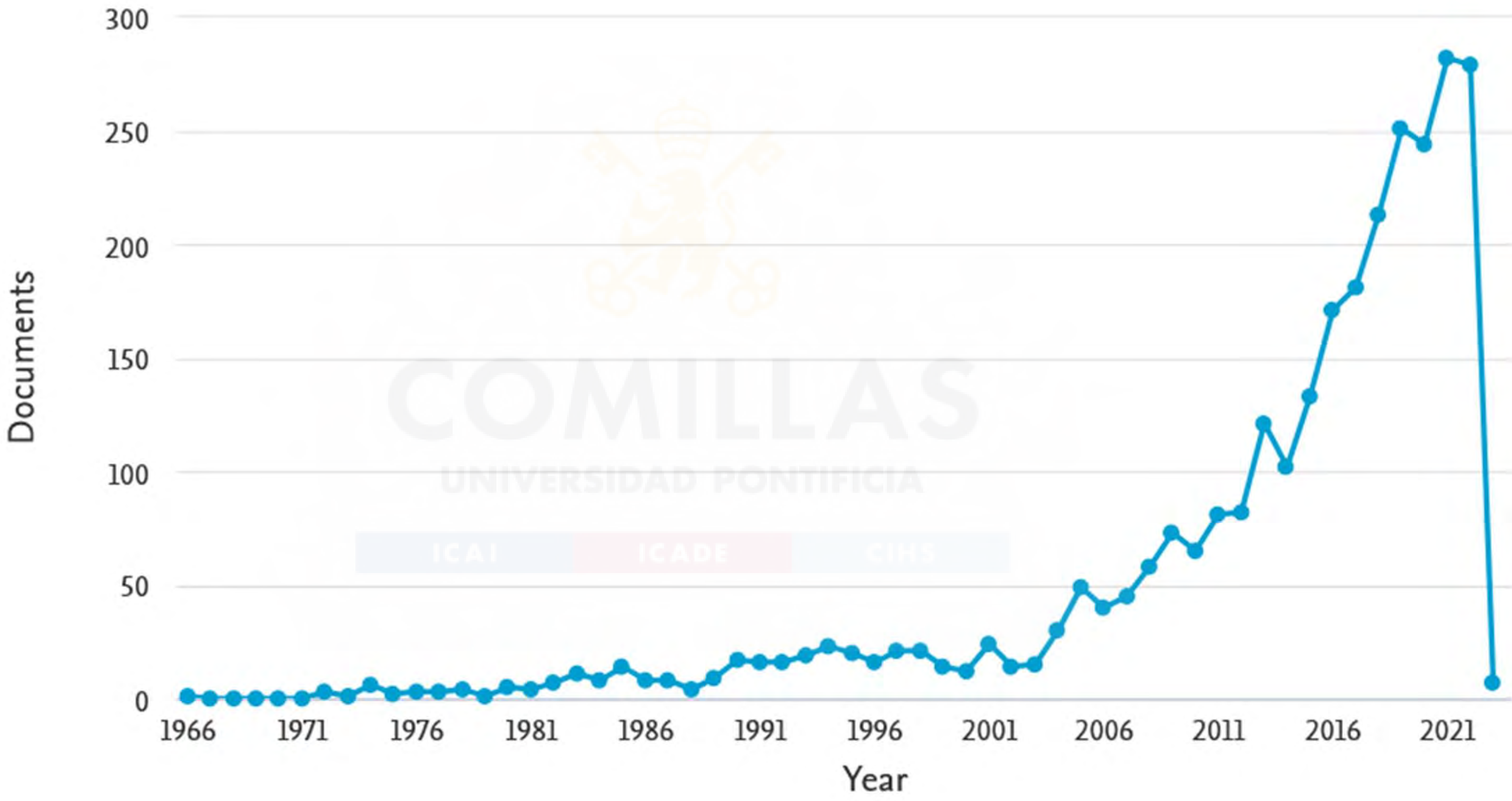
Benders decomposition

- Jacques F. Benders (https://en.wikipedia.org/wiki/Jacques_F._Benders)
- Benders decomposition or
  - Primal (because primal values are fixed),
  - L-shaped (because of the structure of the constraint matrix),
  - Resource (master problem assigns resources),
  - Outer approximation (recourse function is outer approximated)
- Splits the two-stage linear programming problem PL-2 into master and subproblem

R. Rahmaniania, T.G. Crainic, M. Gendreau, W. Reia, *The Benders decomposition algorithm: A literature review* European Journal of Operational Research 259 (3), 801–817, June 2017 10.1016/j.ejor.2016.12.005

Decomposing in Bender's Tavern in Denver, CO, USA (INFORMS Nov 2004)

“Benders decomposition” references in Scopus



Bd Algorithm deduction (i)

$c_1 \in \mathbb{R}^{n_1}$	$c_2 \in \mathbb{R}^{n_2}$
$A_1 \in \mathbb{R}^{m_1 \times n_1}$	$A_2 \in \mathbb{R}^{m_2 \times n_2}$
$b_1 \in \mathbb{R}^{m_1}$	$b_2 \in \mathbb{R}^{m_2}$
$x_1 \in \mathbb{R}^{n_1}$	$x_2 \in \mathbb{R}^{n_2}$

- Two-stage linear programming PL-2

$$\begin{aligned} \min_{x_1, x_2} (c_1^T x_1 &+ c_2^T x_2) \\ A_1 x_1 & &= b_1 \\ B_1 x_1 &+ A_2 x_2 &= b_2 \\ x_1, & \quad x_2 &\geq 0 \end{aligned}$$

- It can also be expressed as

$$\begin{aligned} \min_{x_1, \theta_2(x_1)} \; & c_1^T x_1 + \theta_2(x_1) \\ & A_1 x_1 = b_1 \\ & x_1 \geq 0 \end{aligned}$$

COMPLETE MASTER PROBLEM

- Being $\theta_2(x_1) \in \mathbb{R}$ the recourse function (piecewise linear)

$$\begin{aligned} \theta_2(x_1) = \quad & \min_{x_2} c_2^T x_2 \\ & A_2 x_2 = b_2 - B_1 x_1 \qquad : \pi_2 \\ & x_2 \geq 0 \end{aligned}$$

SUBPROBLEM

- And π_2 the dual variables of the second-stage constraints

Bd Algorithm deduction (ii)

- We express the subproblem is its dual form

$$\theta_2(x_1) = \max_{\pi_2} (b_2 - B_1 x_1)^T \pi_2$$
$$A_2^T \pi_2 \le c_2$$

- Feasible region is independent of x_1
- Vertices of the polyhedron $\Pi = \{\pi_2^1, \pi_2^2, \dots, \pi_2^\nu\}$
- Maximum will be at one vertex

$$\theta_2(x_1) = \max\{(b_2 - B_1 x_1)^T \pi_2^l\} \quad l = 1, \dots, \nu$$

- Expressed as a linear problem (complete master problem)

$$\begin{aligned} \theta_2(x_1) = \quad & \min_{\theta_2} \theta_2 \\ & \theta_2 \ge (b_2 - B_1 x_1)^T \pi_2^1 \\ & \vdots \\ & \theta_2 \ge (b_2 - B_1 x_1)^T \pi_2^\nu \end{aligned}$$

Cuts
Cutting planes
Support hyperplanes

Bd Algorithm deduction (iii)

- One cut is introduced in each iteration
- Relaxed Master problem

$$\begin{aligned}
&\min_{x_1,\theta_2} c_1^T x_1 + \theta_2 \\
&A_1 x_1 = b_1 \\
&\pi_2^{lT} B_1 x_1 + \theta_2 \geq \pi_2^{lT} b_2 \qquad l = 1,\dots,j \\
&x_1 \geq 0
\end{aligned}$$

- Dual variable generated by the subproblem is different in each iteration. As the number of vertices is finite, the number of algorithm iterations is also finite
- Valid cut: outer cut of the recourse function (not necessarily tangent). It is NOT necessary to solve the subproblem up to optimality

Bd Algorithm deduction (iv)

- Alternative formulation of Benders cuts

$$\theta_2 \geq \pi_2^{jT}(b_2 - B_1 x_1) = \pi_2^{jT}(b_2 - B_1 x_1 + B_1 x_1^j - B_1 x_1^j) =$$
$$= \pi_2^{jT}\left[b_2 - B_1 x_1^j - B_1(x_1 - x_1^j)\right] = \pi_2^{jT}\left[b_2 - B_1 x_1^j\right] + \pi_2^{jT}\left[-B_1(x_1 - x_1^j)\right]$$

- Being $f_2^j = \pi_2^{jT}\left[b_2 - B_1 x_1^j\right]$
- Benders cuts can also be expressed as (linearization around a point)

$$\theta_2 - f_2^l \geq \pi_2^{lT} B_1 (x_1^l - x_1)$$

$$\theta_2 + \pi_2^{lT} B_1 x_1 \geq f_2^l + \pi_2^{lT} B_1 x_1^l$$

- $\pi_2^{jT} B_1$ is a subgradient of $\theta_2(x_1)$

Bd Relaxed Master problem and Subproblem

- Bd Relaxed Master: first stage + cuts

$$\min_{x_1,\theta_2} c_1^T x_1 + \theta_2$$
$$A_1 x_1 = b_1$$
$$\pi_2^{lT} B_1 x_1 + \theta_2 \geq f_2^l + \pi_2^{lT} B_1 x_1^l \qquad l = 1,\dots,j$$
$$x_1 \geq 0$$

- Bd Subproblem: second stage with known decisions of the first stage

$$f_2^j = \min_{x_2} c_2^T x_2$$
$$A_2 x_2 = b_2 - B_1 x_1^j \qquad : \pi_2^j$$
$$x_2 \geq 0$$

Bd Feasibility cuts (i)

- So far, we have assumed a feasible subproblem for master decisions (complete recourse or relatively complete)
- If the subproblem is
  - Feasible we build optimality cuts
  - Infeasible we build feasibility cuts
  - Unbounded, then PL-2 is unbounded
- Phase I of the simplex method (minimize the sum of infeasibilities)

$$\min_{x_2, v^+, v^-} e^T v^+ + e^T v^-$$
$$A_2 x_2 + I v^+ - I v^- = b_2 - B_1 x_1 \quad : \pi_2$$
$$x_2, v^+, v^- \geq 0$$

Farkas' lemma

- Let's define primal and dual problems

$$\begin{aligned} &\min c^T x \\ &Ax = b \\ &x \geq 0 \end{aligned} \qquad \begin{aligned} &\max b^T y \\ &A^T y \leq c \end{aligned}$$

- Exactly one of these systems of equations has a solution
  – If the first is true, necessarily the second is false

Primal feasibility condition

$$\begin{aligned} &A^T x = b \\ &x \geq 0 \end{aligned} \qquad \begin{aligned} &A^T y \leq 0 \\ &b^T y > 0 \end{aligned}$$

Bd Feasibility cuts (ii)

- Subproblem

$$f_2^j = \min_{x_2} c_2^T x_2$$
$$A_2 x_2 = b_2 - B_1 x_1^j \quad : \pi_2^j$$
$$x_2 \geq 0$$

will be feasible if the dual variables that satisfy

$A_2^T \pi_2 \leq 0$ also met $(b_2 - B_1 x_1)^T \pi_2 \leq 0$

- According to Farka's lemma, it can then be formulated as

$$\max_{\pi_2} (b_2 - B_1 x_1)^T \pi_2$$
$$A_2^T \pi_2 \leq 0$$

- If this o.f. is strictly positive, a feasibility cut will be generated to avoid this (infeasible subproblem)

$$(b_2 - B_1 x_1)^T \pi_2^j \leq 0$$

Bd Feasibility cuts (iii)

- As the subproblem is a cone, it could be unbounded; some bounds on the dual variables are introduced

$$\max_{\pi_2} (b_2 - B_1 x_1)^T \pi_2$$
$$A_2^T \pi_2 \leq 0$$
$$-1 \leq \pi_2 \leq 1$$

- The dual problem of this problem is the infeasibility minimization subproblem

$$\min_{x_2, v^+, v^-} e^T v^+ + e^T v^-$$
$$A_2 x_2 + I v^+ - I v^- = b_2 - B_1 x_1 \quad : \pi_2$$
$$x_2, v^+, v^- \geq 0$$

Bd Feasibility cuts (iv)

- Relaxed Master problem

$$\min_{x_1,\theta_2} c_1^T x_1 + \theta_2$$
$$A_1 x_1 = b_1$$
$$\pi_2^{lT} B_1 x_1 + \delta_1^l \theta_2 \geq \pi_2^{lT} b_2 \qquad l = 1,\ldots,j$$
$$x_1 \geq 0$$

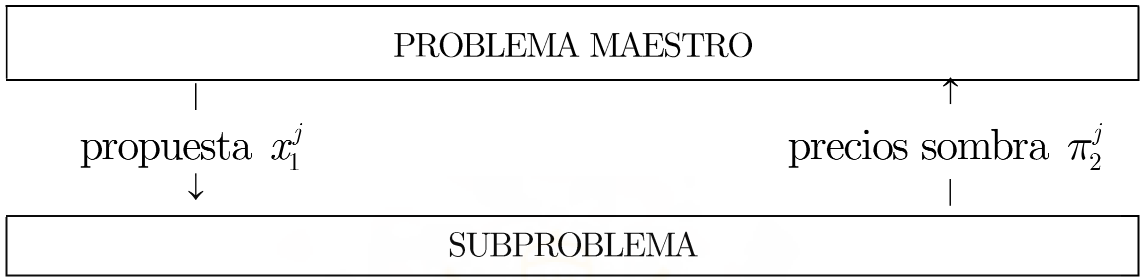
$\delta_1^l = 1$ for optimality cuts and $\delta_1^l = 0$ for feasibility cuts

- Feasibility cuts are optimality cuts with infinite slope
- These cuts avoid master solutions that make infeasible the subproblem and allow to keep those that are feasible
- Alternatively, instead of feasibility cuts, we can penalize the infeasibilities in the subproblem

Bd Relaxed Master and Subproblem

- Bd Relaxed Master
  - One cut is added in each iteration
  - Each cut defines a new feasible region
  - Optimal solution of the previous iteration becomes infeasible
  - It is worth using the dual simplex method or lazy constraints option in Gurobi and CPLEX
  - Size $(m_1 + j) \times (n_1 + 1)$
  - It can be nonconvex (MIP, NLP)
- Bd Subproblem
  - Each iteration modifies the RHS of the constraints
  - It is worth using the primal simplex method (if the size is reasonable) $m_2 \times n_2$
  - Size
  - It must be convex (LP, NLP). Generalized Benders decomposition (GBD)

Bd Convergence



- Upper bound of the optimal value of the o.f. of problem PL-2 $$\overline{z} = c_1^T x_1^j + c_2^T x_2^j$$
- Lower bound of the problem, the value obtained by the o.f. of the relaxed master problem $$\underline{z} = c_1^T x_1^j + \theta_2^j$$
- Convergence condition

$$\frac{|\overline{z} - \underline{z}|}{|\overline{z}|} = \frac{\left|c_2^T x_2^j - \theta_2^j\right|}{\left|c_1^T x_1^j + c_2^T x_2^j\right|} \le \varepsilon$$

or repetition of the last master proposal

Bd Algorithm (i)

- Successive approximation of the second-stage objective function by cuts.
- Benders cuts (cutting planes, support hyperplanes) are an outer linearization of the recourse function.
- Lower bound is monotonically increasing.

  Upper bound is not necessarily monotonically decreasing.
  - Upper bound is the minimum of previous upper bounds
- In the first iteration, the value of x_1^0 can be fixed if the problem nature is known or by solving the master problem without cuts $\theta_2 = 0$
- In each iteration, we have a quasi-optimal feasible solution

Bd Algorithm (ii)

1. Initialization: $j=0$ $\underline{z}=-\infty$ $\overline{z}=\infty$ $\varepsilon=10^{-4}$
2. Solving the Bd Relaxed Master problem

$$
\begin{aligned}
\min_{x_1,\theta_2}\ & c_1^T x_1 + \theta_2 \\
& A_1 x_1 = b_1 \\
& \pi_2^{lT} B_1 x_1 + \delta_1^l \theta_2 \ge \pi_2^{lT} b_2 \qquad l = 1,\dots,j \\
& x_1 \ge 0
\end{aligned}
\qquad
\begin{aligned}
\min_{x_1,\theta_2}\ & c_1^T x_1 + \theta_2 \\
& A_1 x_1 = b_1 \\
& \pi_2^{lT} B_1 x_1 + \delta_1^l \theta_2 \ge f_2^l + \pi_2^{lT} B_1 x_1^l \qquad l = 1,\dots,j \\
& x_1 \ge 0
\end{aligned}
$$

Determine the solution (x_1^j, θ_2^j) and the lower bound

If no optimality cuts $\theta_2 = 0$

3. Solving the Bd Subproblem of sum of infeasibilities

$$
\begin{aligned}
f_2^j = \min_{x_2, v^+, v^-}\ & e^T v^+ + e^T v^- \\
& A_2 x_2 + I v^+ - I v^- = b_2 - B_1 x_1^j \qquad : \pi_2^j \\
& x_2, v^+, v^- \ge 0
\end{aligned}
$$

If $f_2^j \ge 0$ feasibility cut

If $f_2^j = 0$ go to step 4.

4. Solving the Bd Subproblem

$$
\begin{aligned}
f_2^j = \min_{x_2}\ & c_2^T x_2 \\
& A_2 x_2 = b_2 - B_1 x_1^j \qquad : \pi_2^j \\
& x_2 \ge 0
\end{aligned}
$$

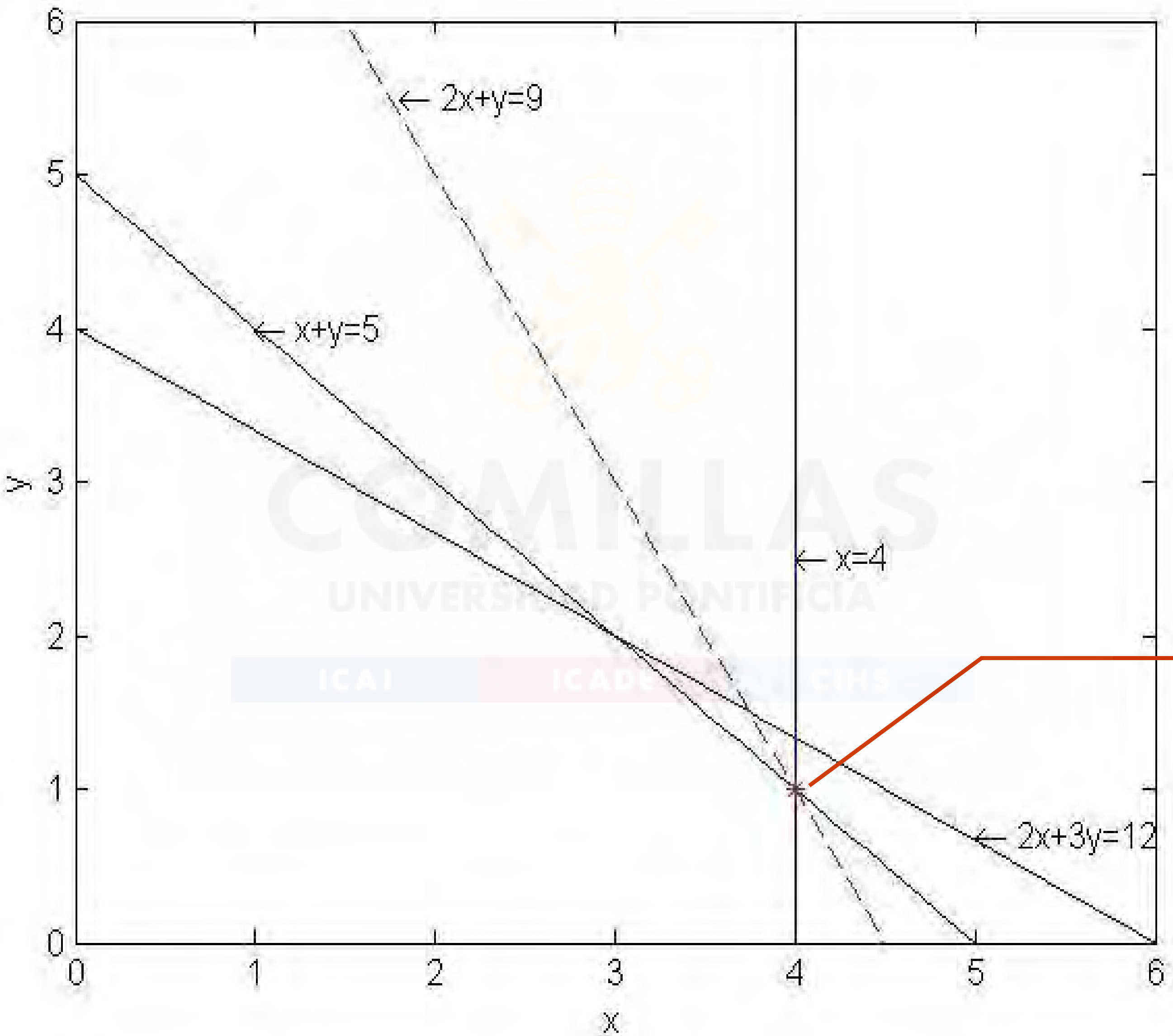
Obtain x_2^j and update the upper bound.

3. If stopping rule is met

$$
\frac{|\overline{z} - \underline{z}|}{|\overline{z}|} = \frac{|c_2^T x_2^j - \theta_2^j|}{|c_1^T x_1^j + c_2^T x_2^j|} \le \varepsilon
$$

If not go to step 2.

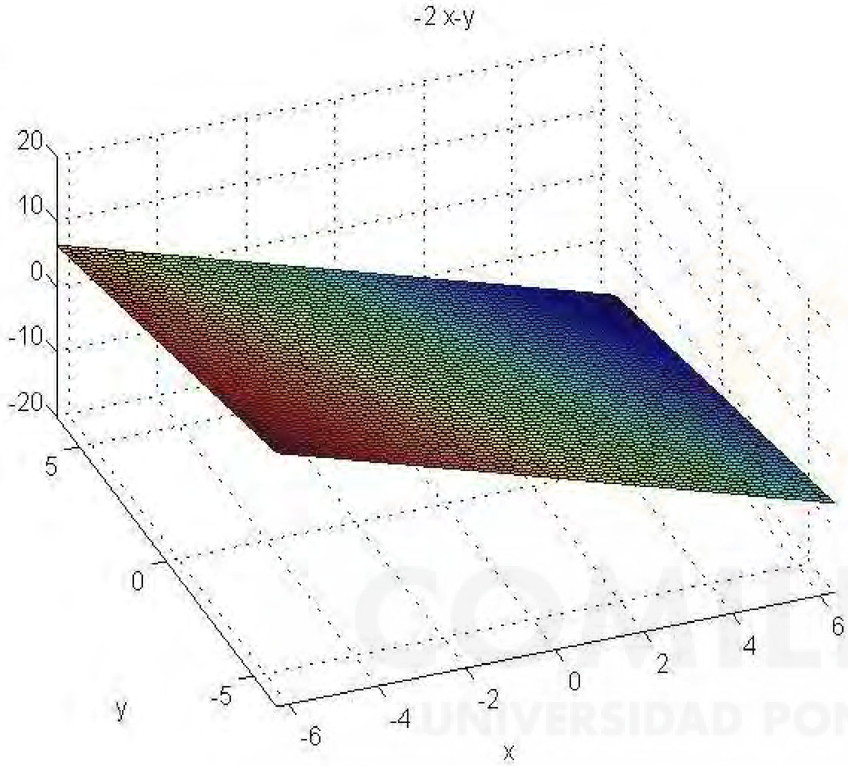
Case study 1: complete problem



$$\min_{x,y} -2x - y$$
$$\begin{array}{lll} x & & \leq 4 \\ x & +y & \leq 5 \\ 2x & +3y & \leq 12 \\ x, & y & \geq 0 \end{array}$$

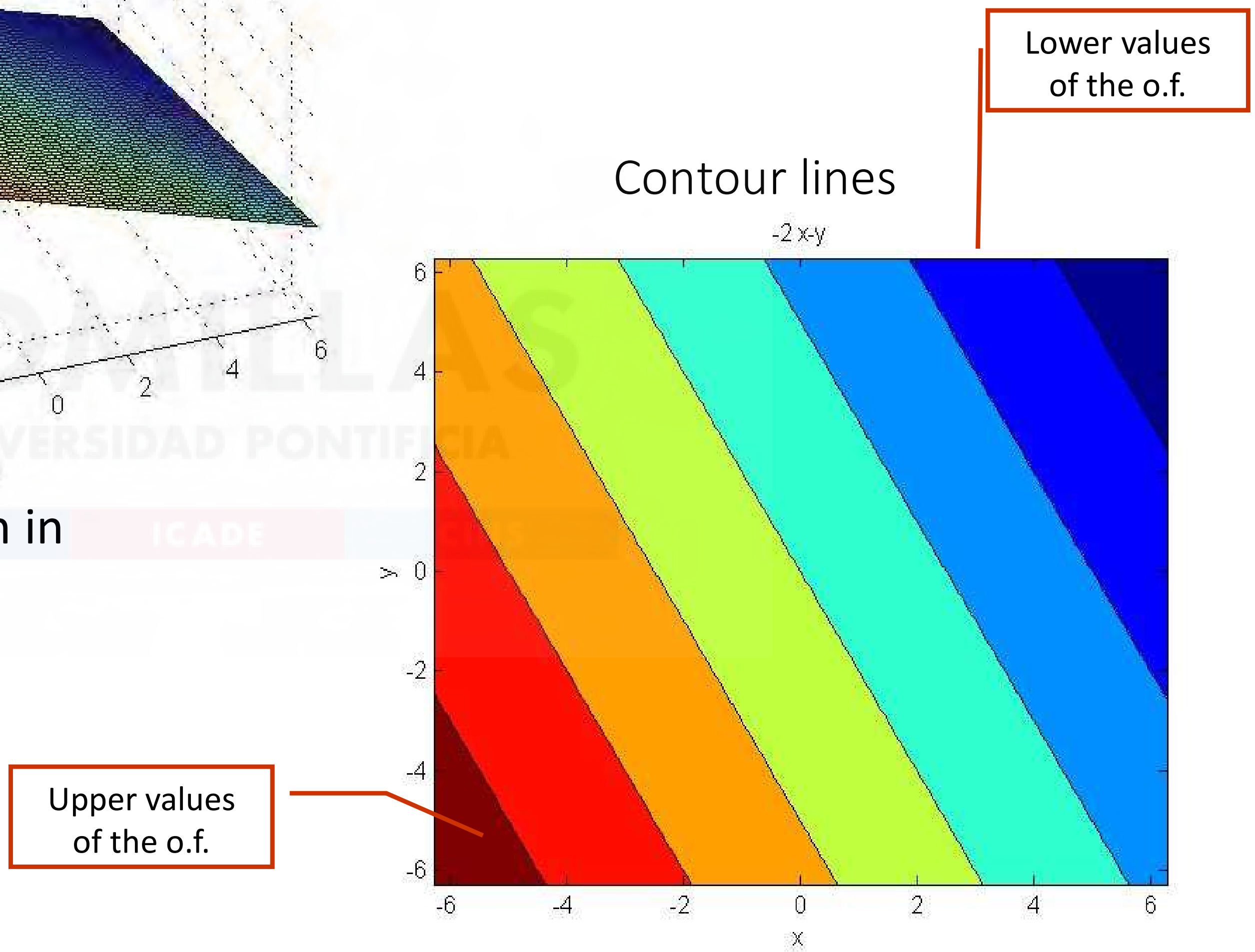
Optimal solution (4,1)
Value of the o.f. -9

Case study 1

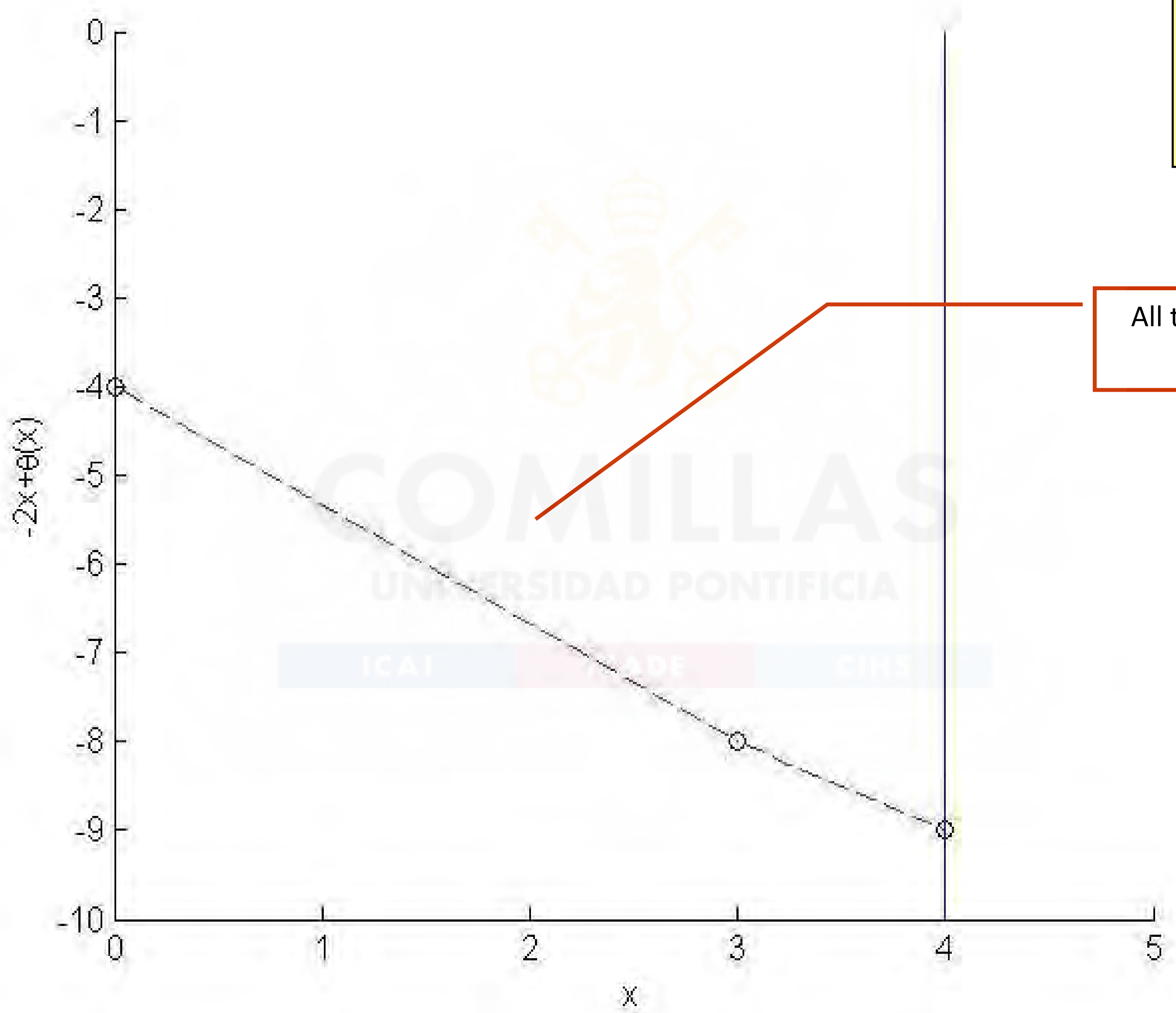


Objective function in 3D

Contour lines



Case study 1: complete master problem

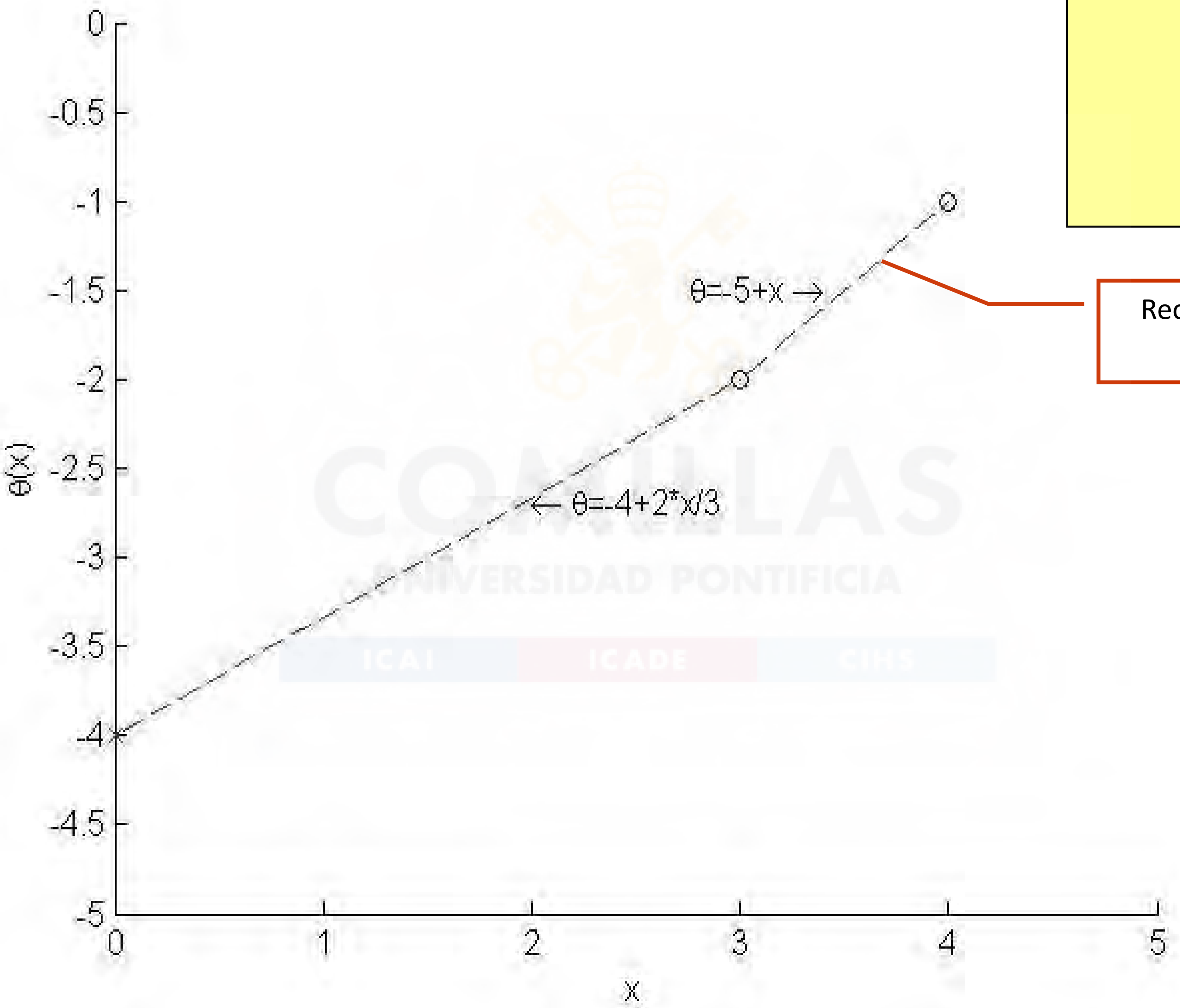


$$\min_x -2x + \theta(x)$$
$$x \leq 4$$
$$x \geq 0$$

All the Benders cuts

¡UNKNOWN!

Case study 1: recourse function (to be discovered iteratively)

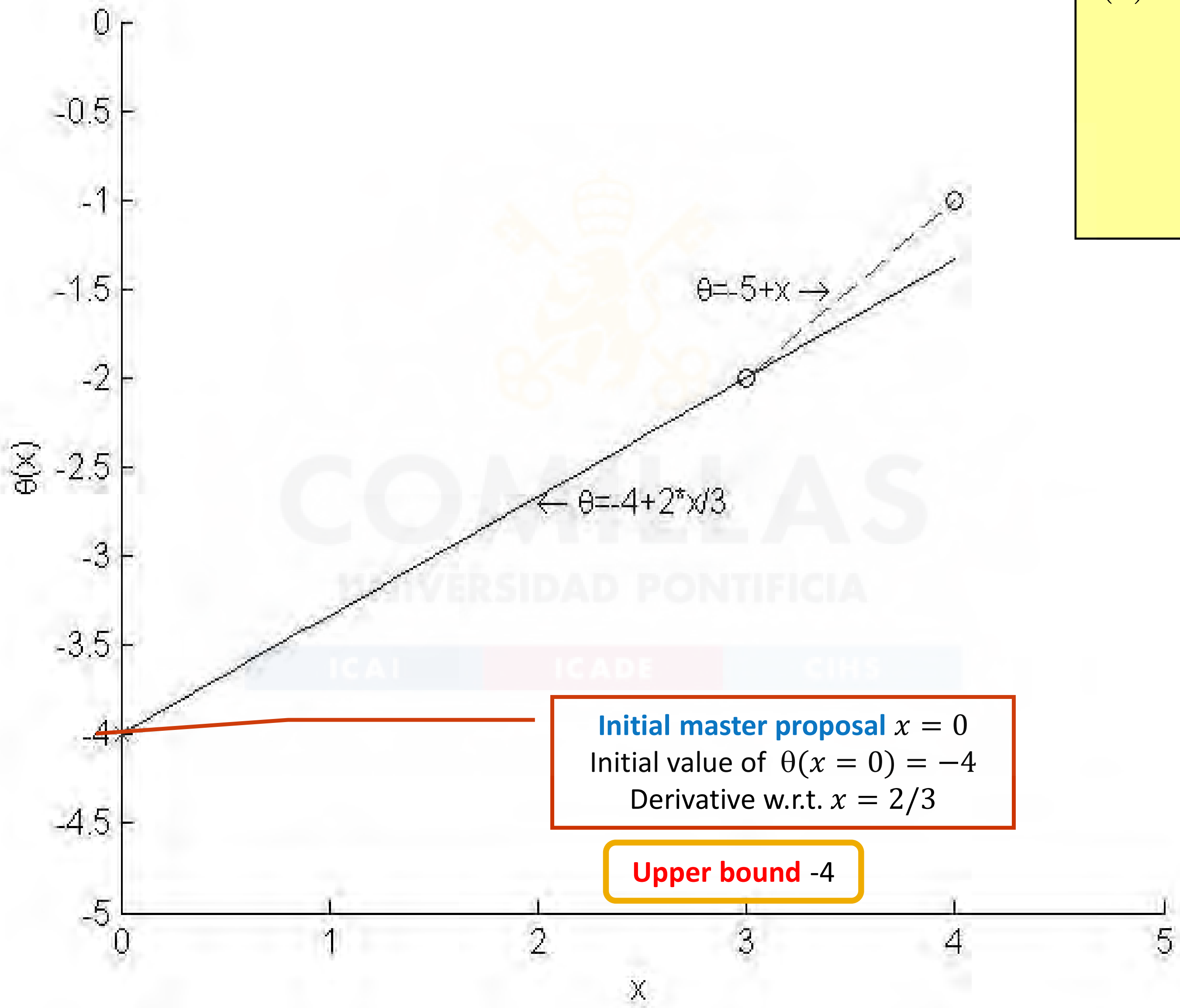


$$\theta(x) = \min_y -y$$
$$y \le 5 - x$$
$$3y \le 12 - 2x$$
$$y \ge 0$$

Recourse function $\theta(x)$
¡UNKNOWN!

Case study 1: recourse function iteration 1

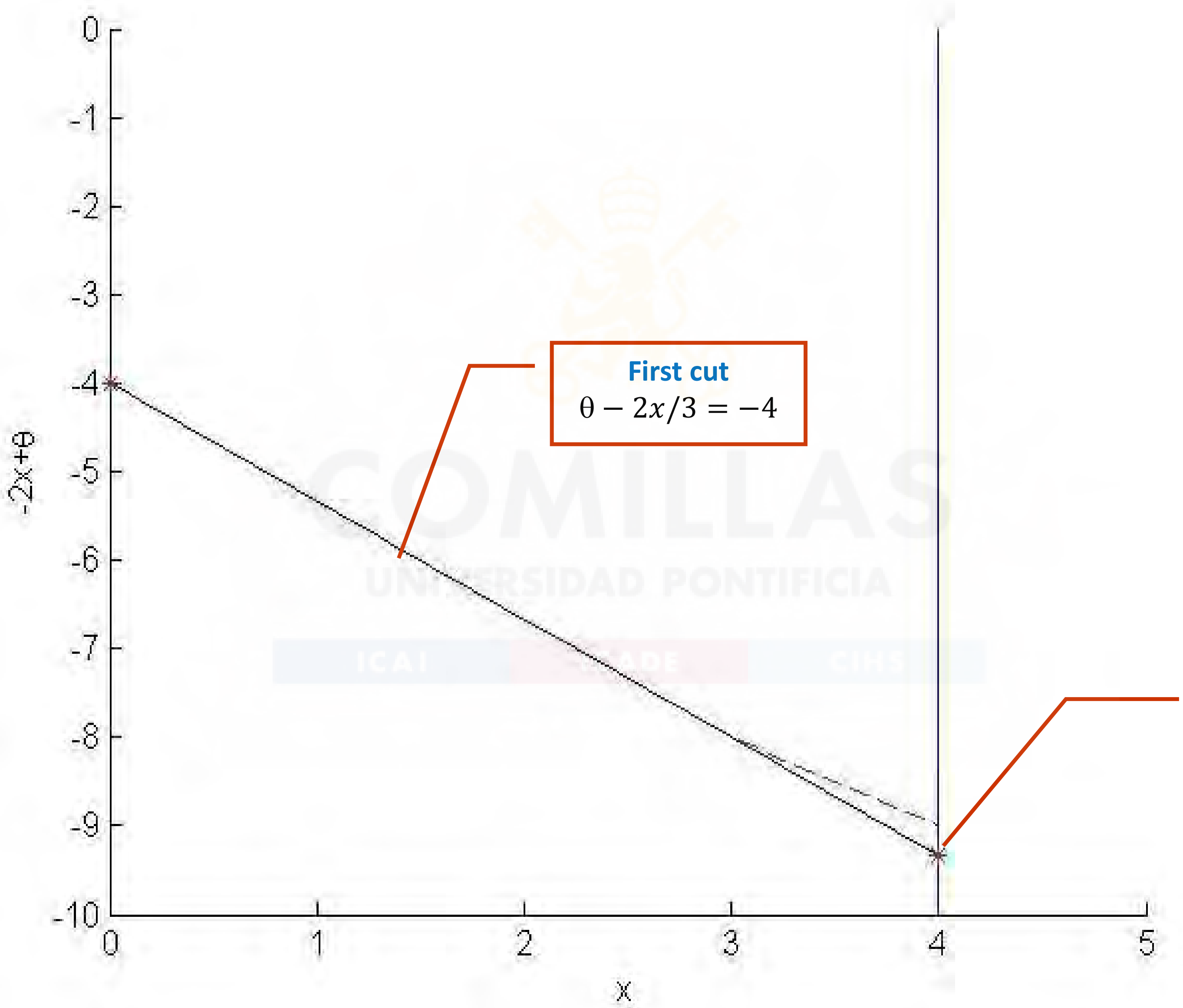
$$\theta(x) = \min_y -y$$
$$y \leq 5 - 0$$
$$3y \leq 12 - 2 \cdot 0$$
$$y \geq 0$$



Initial master proposal $x = 0$
Initial value of $\theta(x = 0) = -4$
Derivative w.r.t. $x = 2/3$

Upper bound -4

Case study 1: master problem iteration 1



$$\min_{x,\theta} -2x+\theta$$
$$x \le 4$$
$$\theta - \frac{2}{3}x \ge -4$$
$$x \ge 0$$

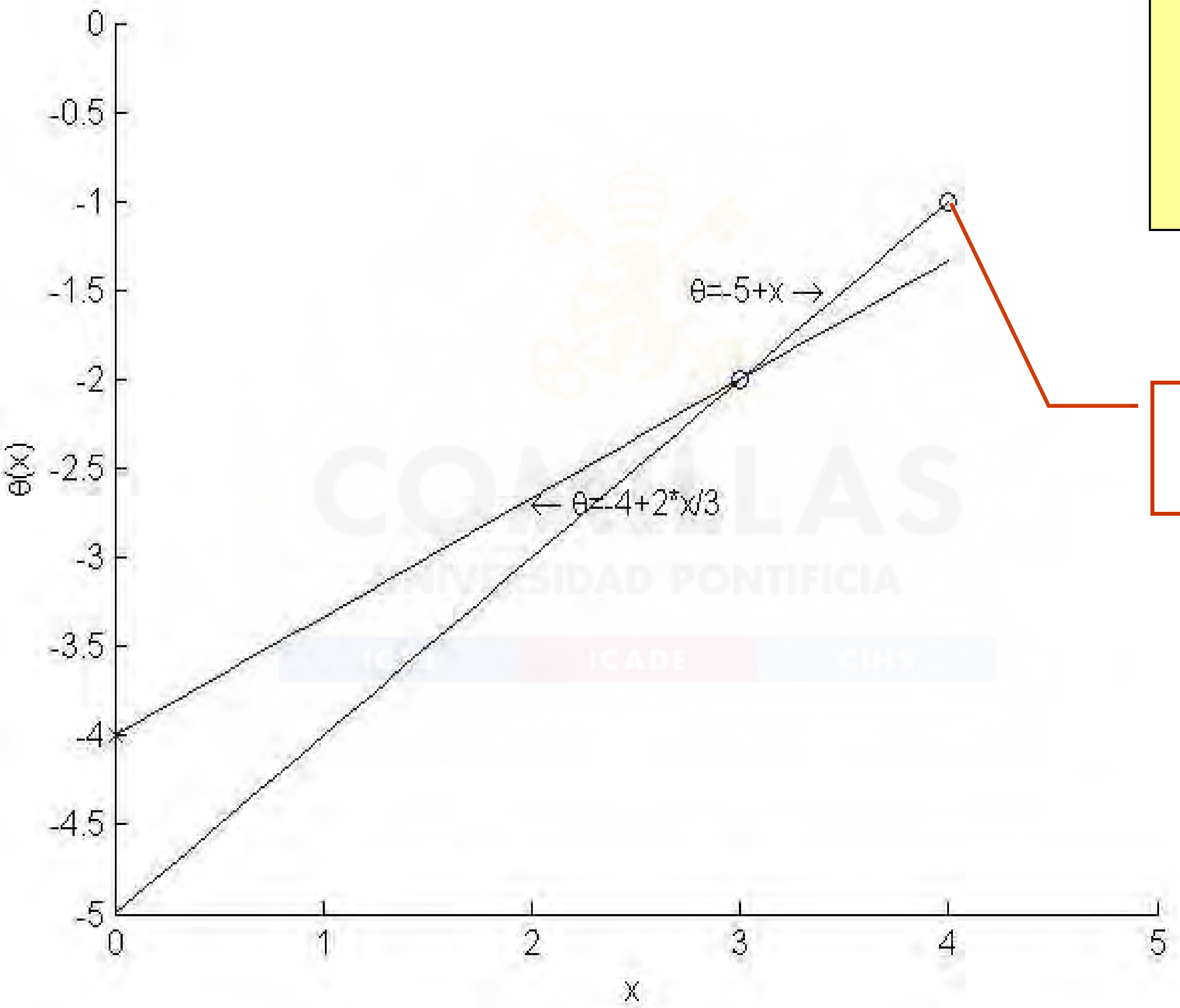
First cut
$\theta - 2x/3 = -4$

Second master proposal $x = 4$
Value of $\theta = -4/3$

Lower bound = -28/3

Case study 1: recourse function iteration 2

$$\theta(x) = \min_y -y$$
$$y \leq 5 - 4$$
$$3y \leq 12 - 2 \cdot 4$$
$$y \geq 0$$

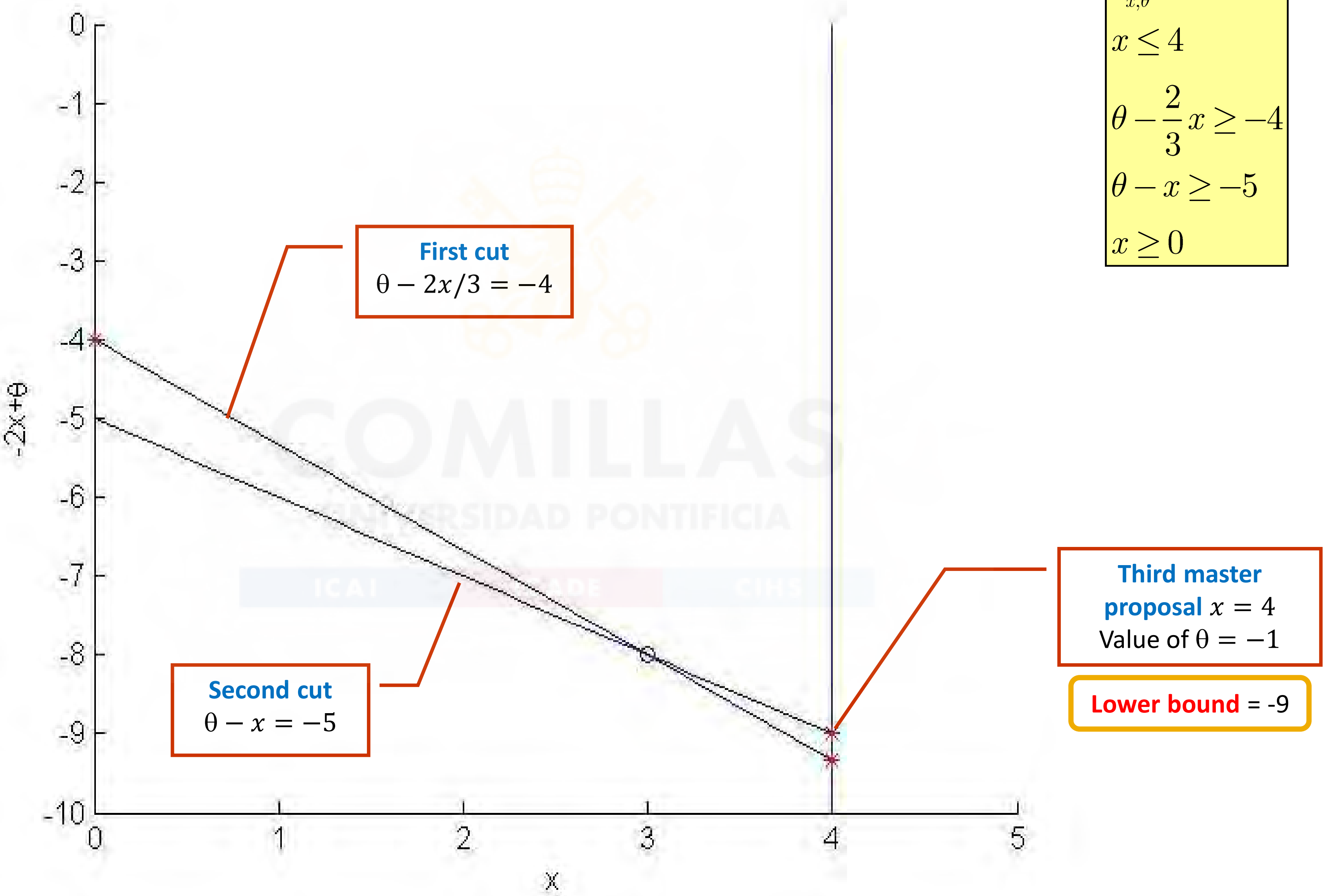


Solution $x = 4$
Value of $\theta(x = 4) = -1$
Derivative w.r.t. $x = 1$

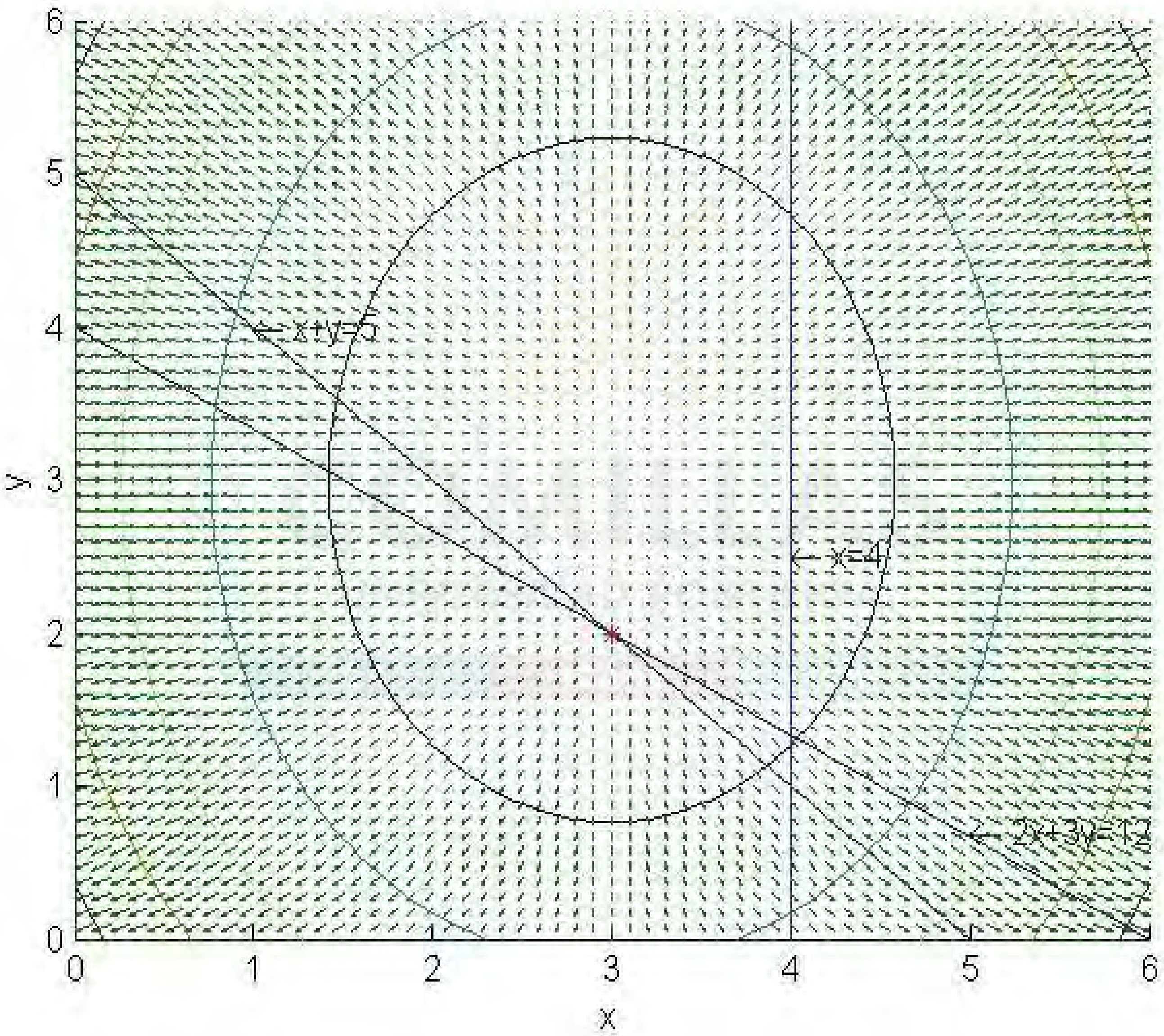
Upper bound -9

Case study 1: master problem iteration 2

$$\min_{x,\theta} -2x + \theta$$
$$x \leq 4$$
$$\theta - \frac{2}{3}x \geq -4$$
$$\theta - x \geq -5$$
$$x \geq 0$$

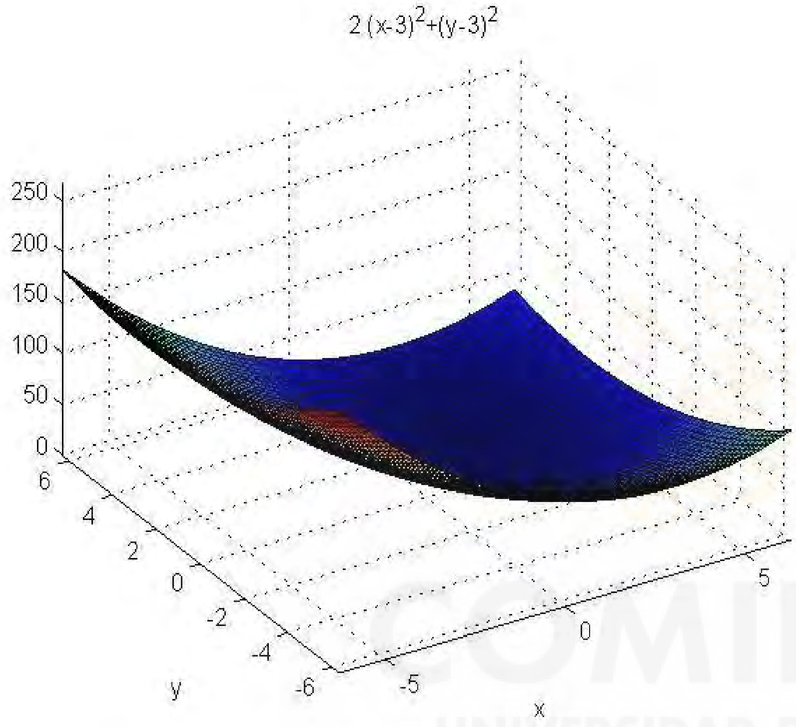


Case study 2: complete problem



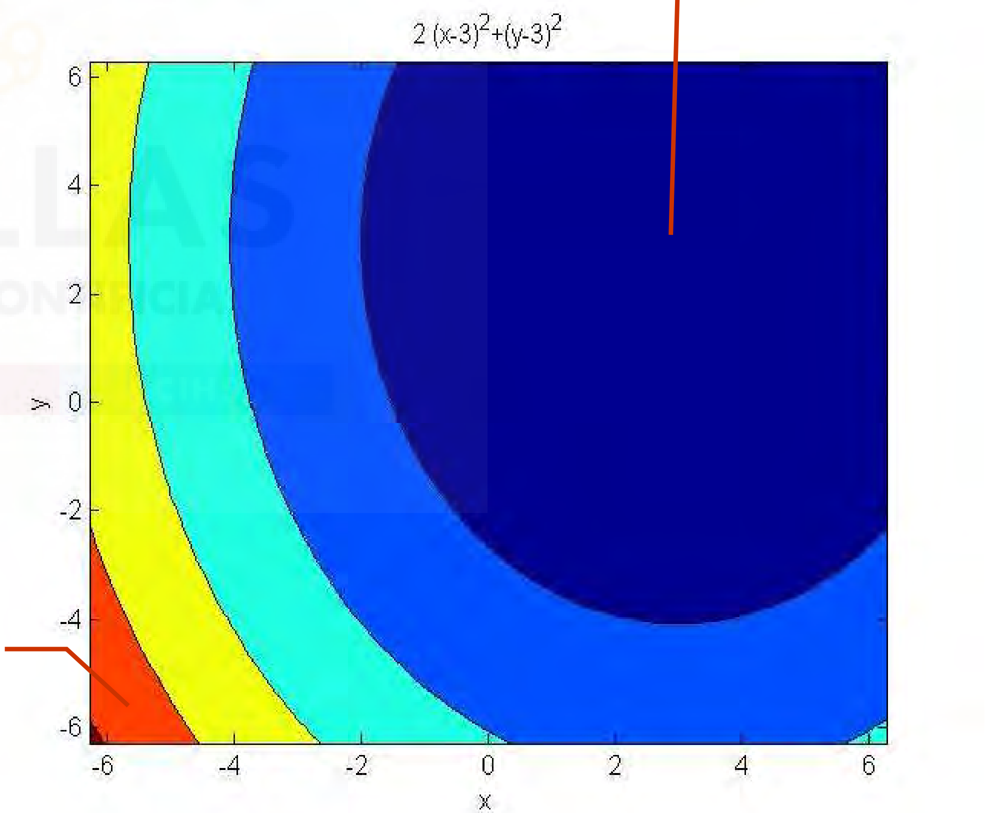
$$
\begin{aligned}
\min_{x,y}\ & 2(x-3)^2 + (y-3)^2 \\
x & \le 4 \\
x \quad + y & \le 5 \\
2x \quad + 3y & \le 12 \\
x, \quad y & \ge 0
\end{aligned}
$$

Case study 2

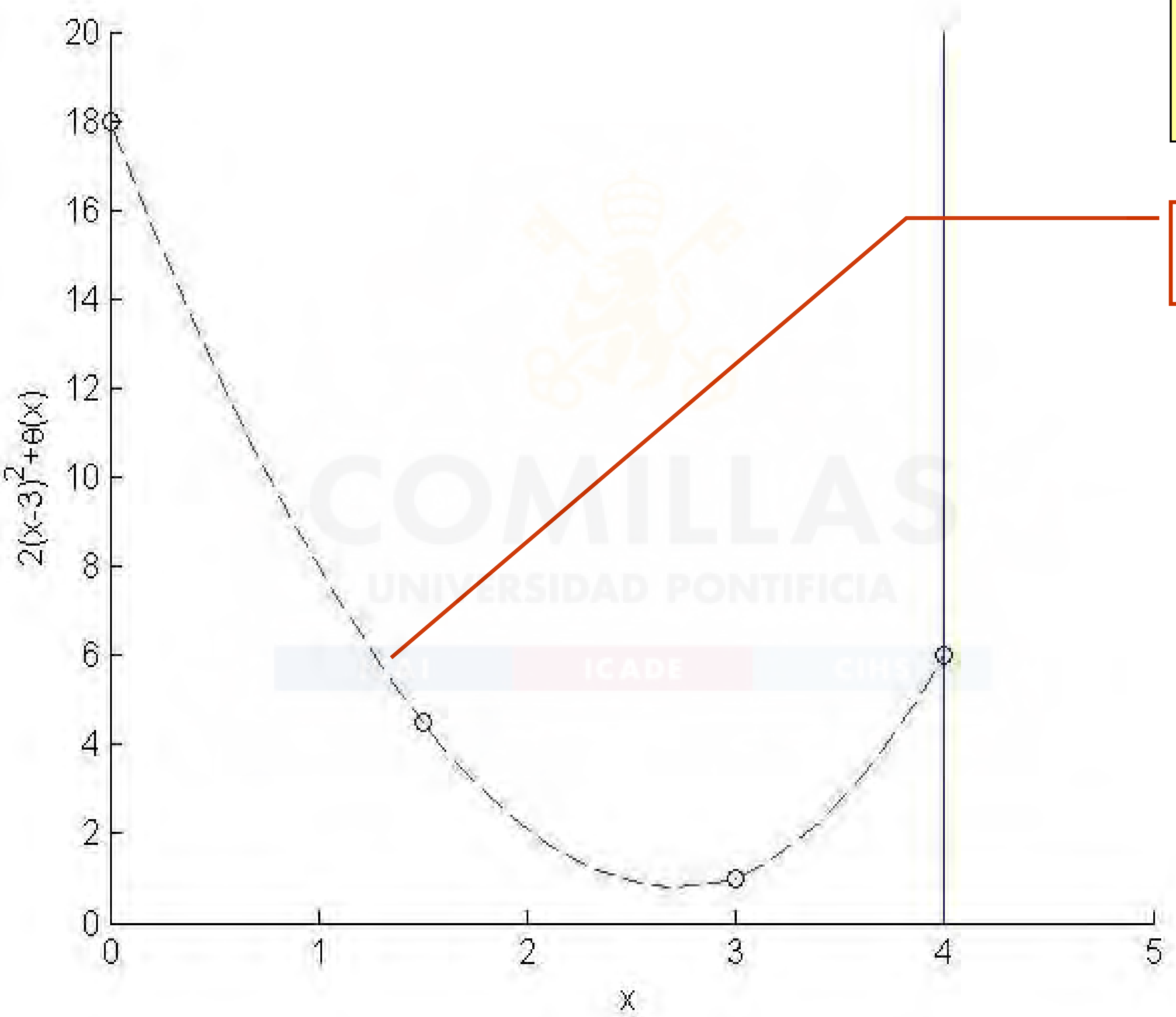


Objective function in 3D

Contour lines



Case study 2: complete master problem

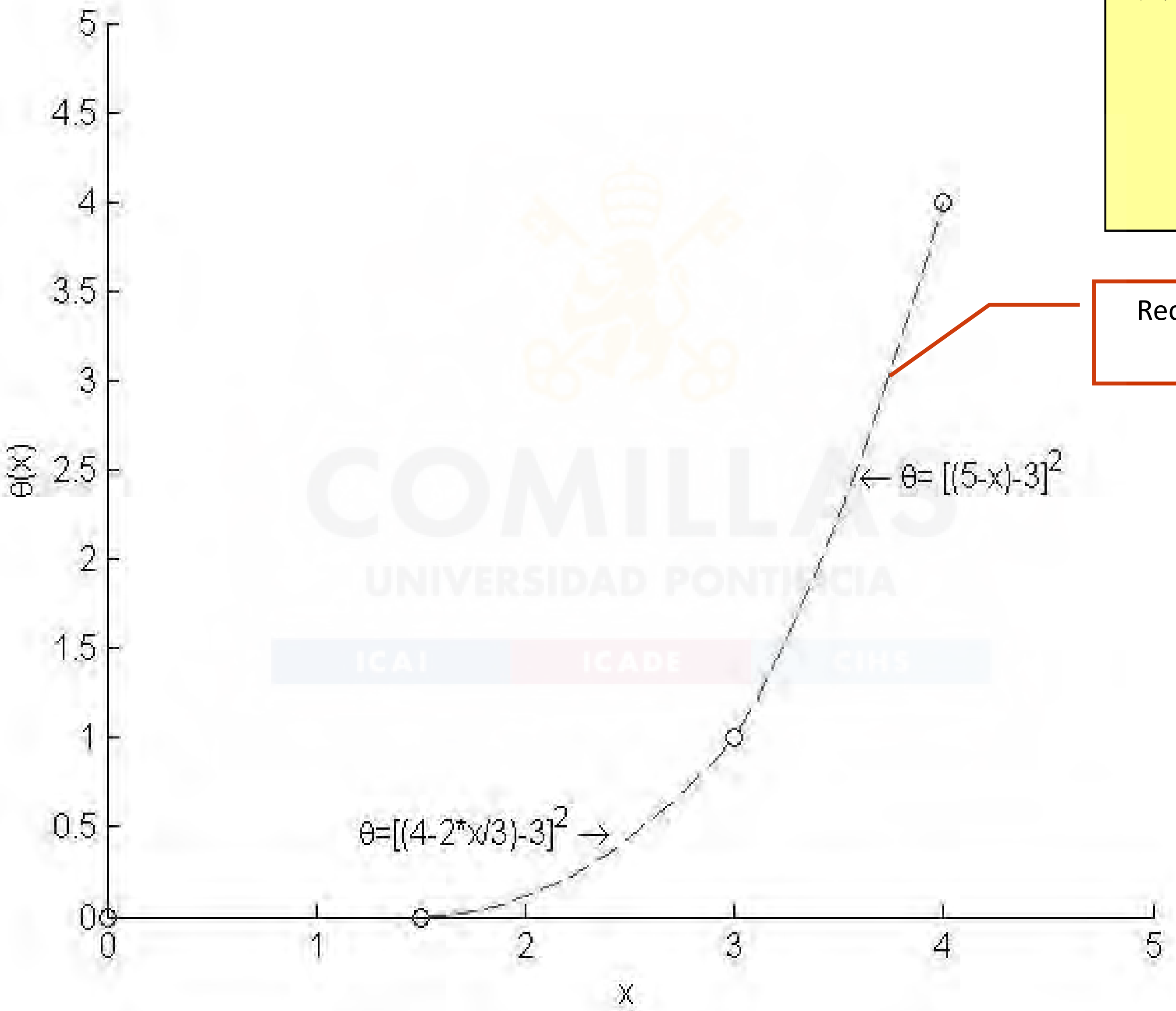


$$\min_x 2(x-3)^2 + \theta(x)$$
$$x \le 4$$
$$x \ge 0$$

All the Benders cuts
¡UNKNOWN!

Case study 2: recourse function

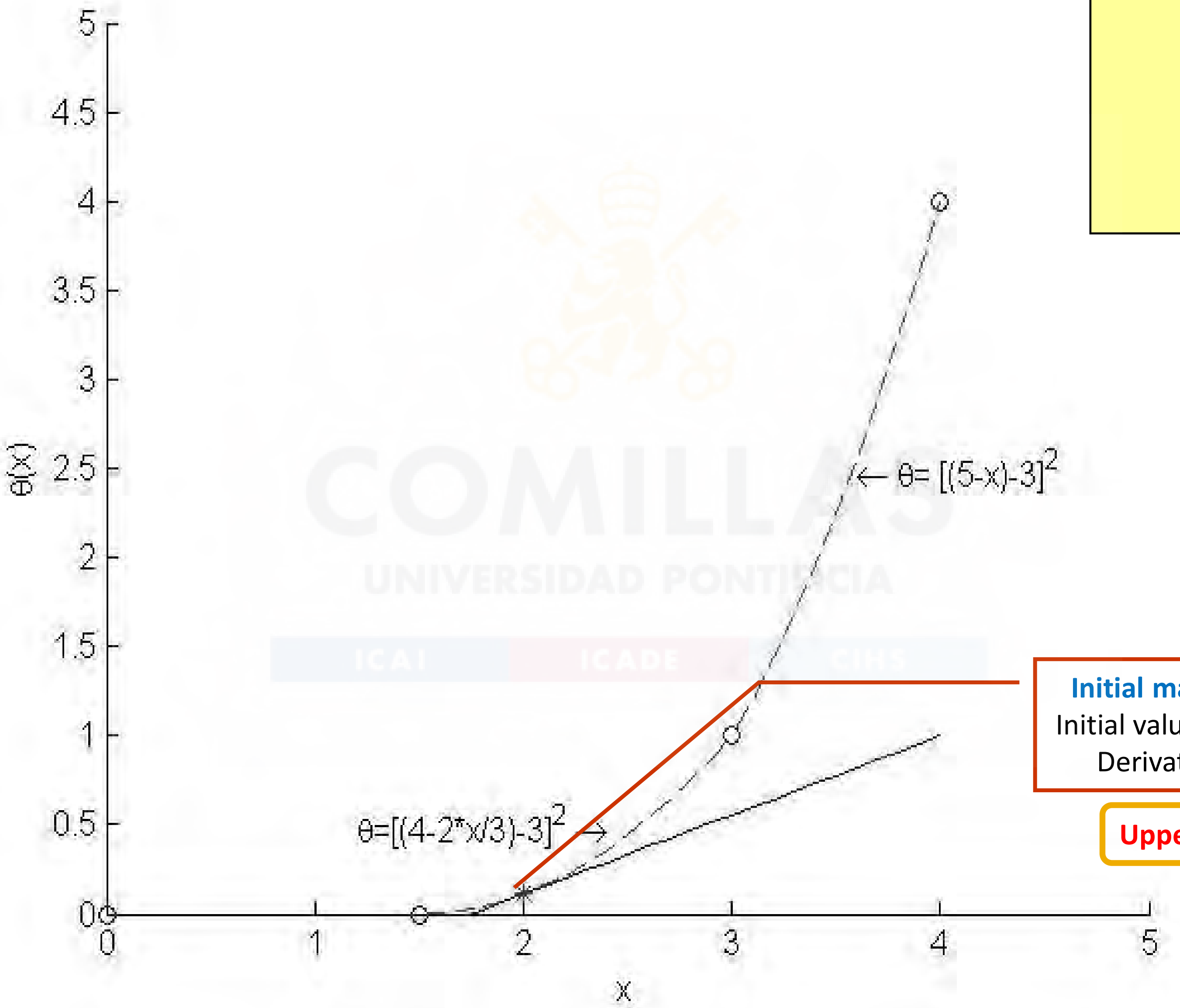
$$\theta(x) = \min_y (y-3)^2$$
$$y \le 5 - x$$
$$3y \le 12 - 2x$$
$$y \ge 0$$



Recourse function $\theta(x)$
¡UNKNOWN!

Case study 2: recourse function iteration 1

$$\theta(x) = \min_y (y-3)^2$$
$$y \le 5 - 2$$
$$3y \le 12 - 2 \cdot 2$$
$$y \ge 0$$

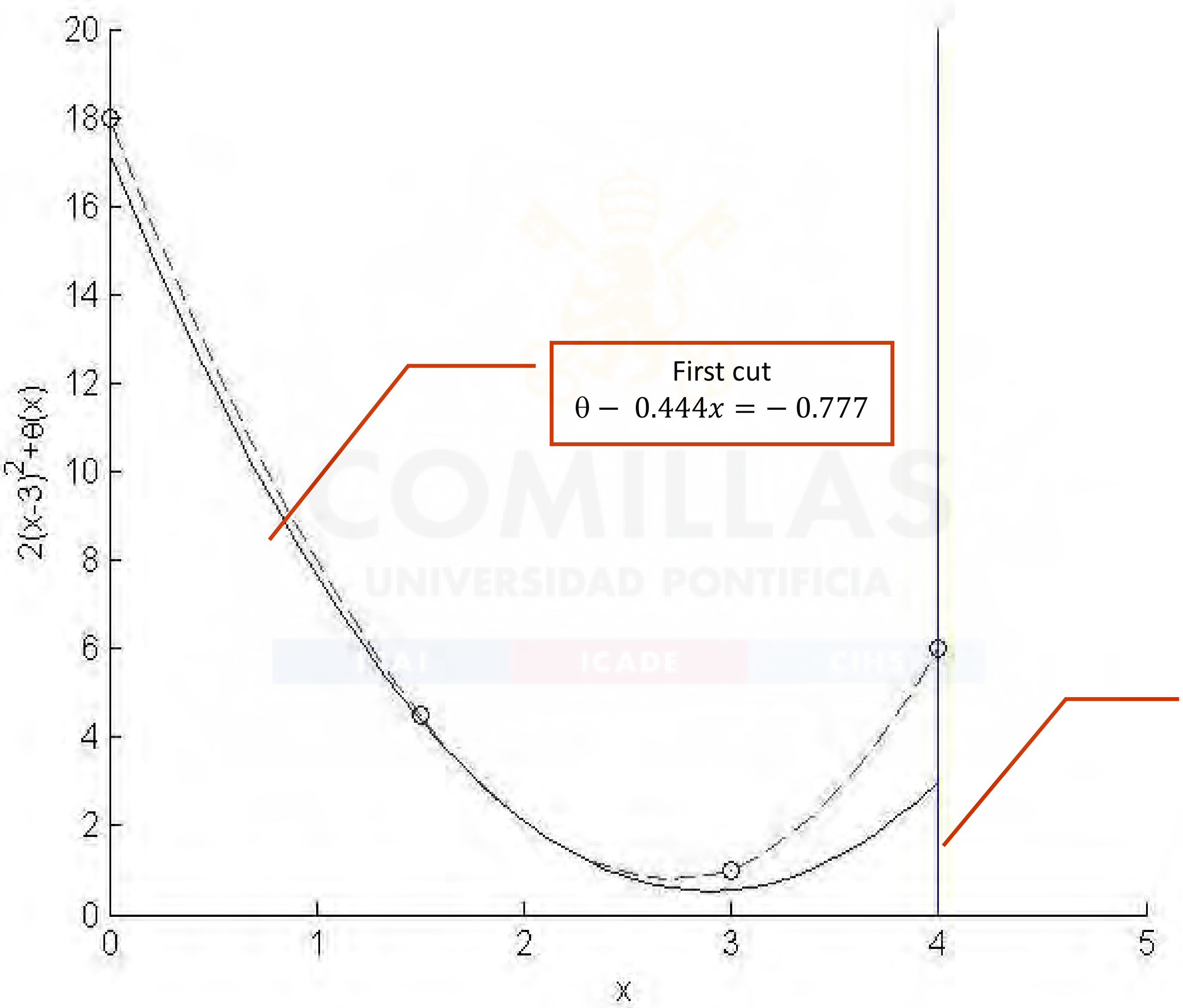


Initial master proposal $x = 2$
Initial value of $\theta(x = 2) = 0.111$
Derivative w.r.t. $x = 0.444$

Upper bound 2.111

Case study 2: master problem iteration 1

$$\min_{x,\theta} 2(x-3)^2 + \theta$$
$$x \le 4$$
$$\theta - 0.\widehat{4}x \ge -0.\widehat{7}$$
$$x \ge 0$$



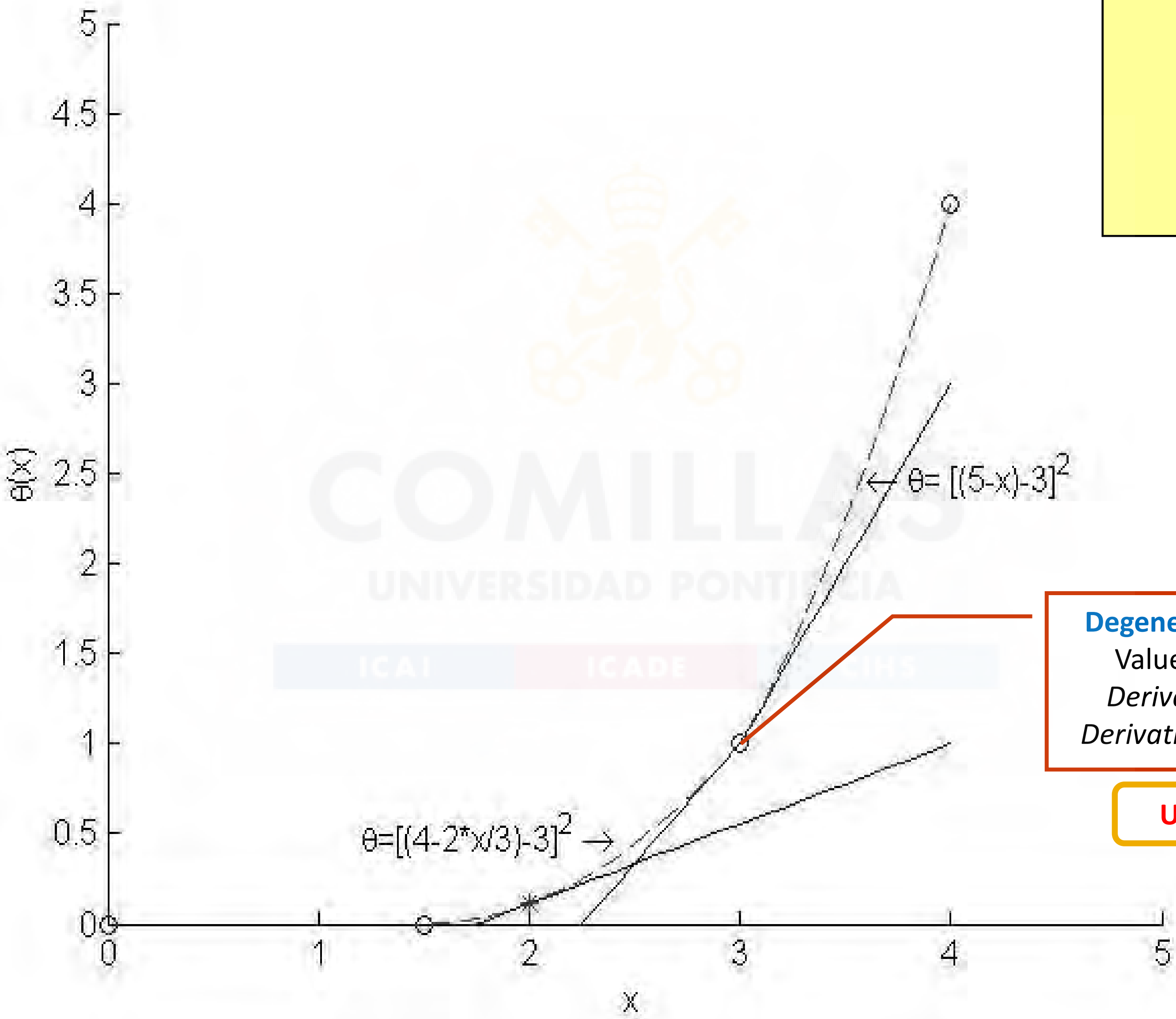
First cut
$\theta - 0.444x = -0.777$

Second master proposal $x = 3$
Value of $\theta = 0.555$

Lower bound = 0.555

Case study 2: recourse function iteration 2

$$\theta(x) = \min_y (y-3)^2$$
$$y \le 5-3$$
$$3y \le 12 - 2\cdot 3$$
$$y \ge 0$$

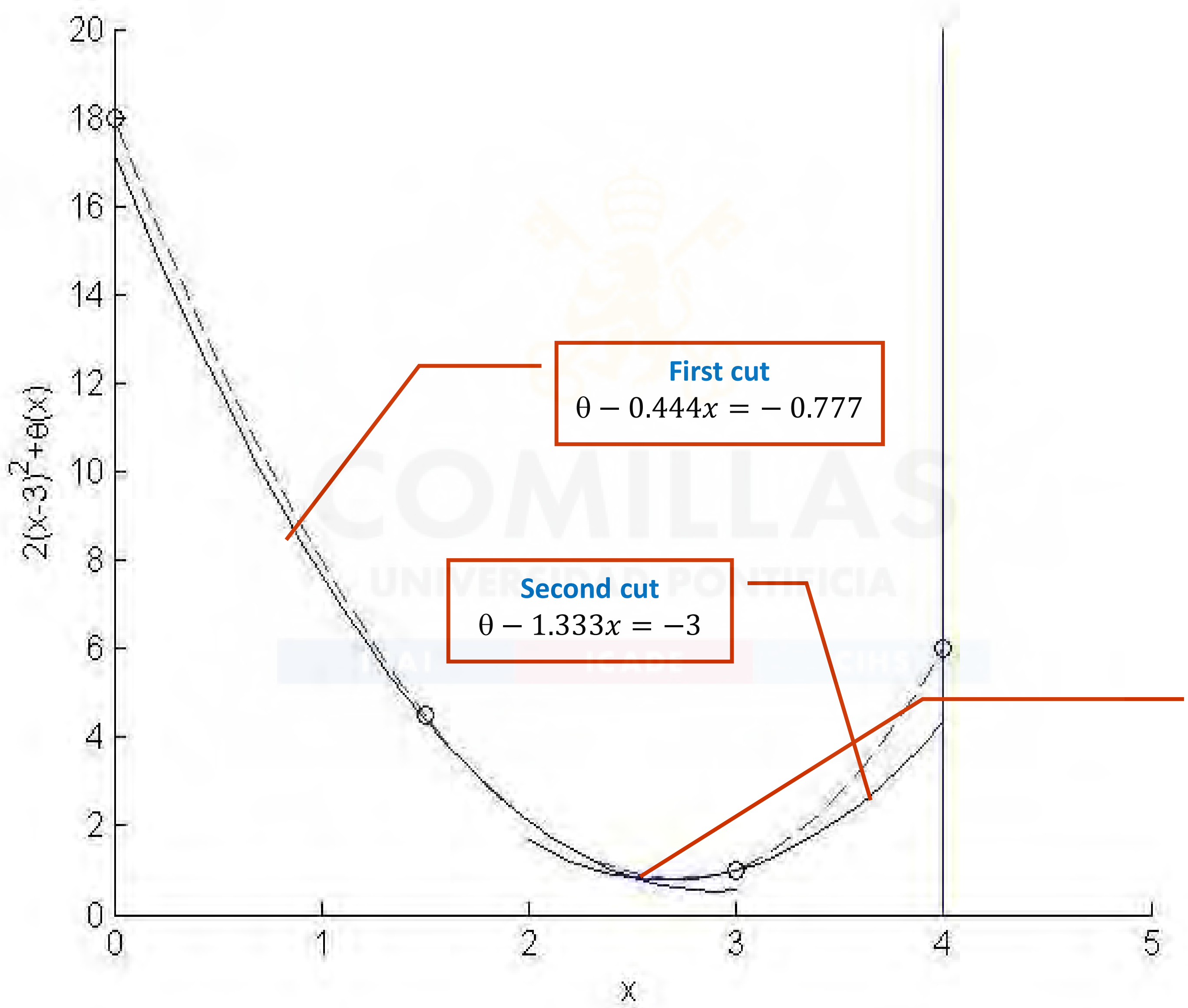


Degenerate solution $x = 3$
Value of $\theta(x=3) = 1$
Derivative + w.r.t. $x = 2$
Derivative - w.r.t. $x = 1.333$

Upper bound 1

Case study 2: master problem iteration 2

$$\min_{x,\theta} 2(x-3)^2 + \theta$$
$$x \le 4$$
$$\theta - 0.\widehat{4}x \ge -0.\widehat{7}$$
$$\theta - 1.\widehat{3}x \ge -3$$
$$x \ge 0$$



Third master proposal
$x = 2.5$
Value of $\theta = 0.333$

Lower bound = 0.8333

Fixed Cost Transportation Problem (FCTP)

- It is a transportation problem where the arc connecting two nodes (i and j) has a fixed cost f_{ij} associated with its installation and a variable cost c_{ij} by the use. We want to minimize the total fixed (investment) and variable (transportation) costs subject to the constraints of demand supply b_j at the destinations and maximum capacity at the origins a_i.

Fixed-Charge Transportation Problem (FCTP)

- Bd Relaxed Master

$$\min_{y_{ij},\theta} \theta + \sum_{ij}(f_{ij}y_{ij})$$

$$\delta^l\theta - \theta^l \geq \sum_{ij} \pi_{ij}^l M_{ij}(y_{ij}^l - y_{ij}) \quad l = 1, \dots, k$$

$$y_{ij} \in \{0,1\}$$

O.F. of the subproblem at iteration l

Dual variables of linking constraints at iteration l

Master proposal at iteration l

Change in the o.f. proportional to changes in master proposals

Complete problem

Flows
(second stage)

Investment decisions
(first stage)

$$\min_{x_{ij},y_{ij}} \sum_{ij}(f_{ij}y_{ij} + c_{ij}x_{ij})$$

$$\sum_j x_{ij} \leq a_i \quad \forall i$$

$$\sum_i x_{ij} \geq b_j \quad \forall j$$

$$x_{ij} \leq M_{ij}y_{ij} \quad \forall ij$$

$$x_{ij} \geq 0, y_{ij} \in \{0,1\}$$

Capacity of each origin

Demand of each destination

Flow can pass only for installed connections

- Bd Subproblem

$$\theta^k = \min_{x_{ij}} \sum_{ij}(c_{ij}x_{ij})$$

$$\sum_j x_{ij} \leq a_i \quad \forall i$$

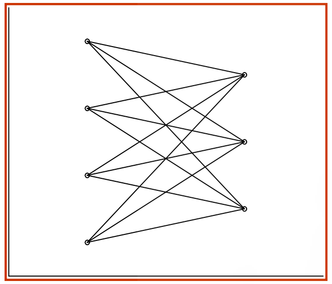
$$\sum_i x_{ij} \geq b_j \quad \forall j$$

$$x_{ij} \leq M_{ij}y_{ij}^k \quad \forall ij \quad : \pi_{ij}^k$$

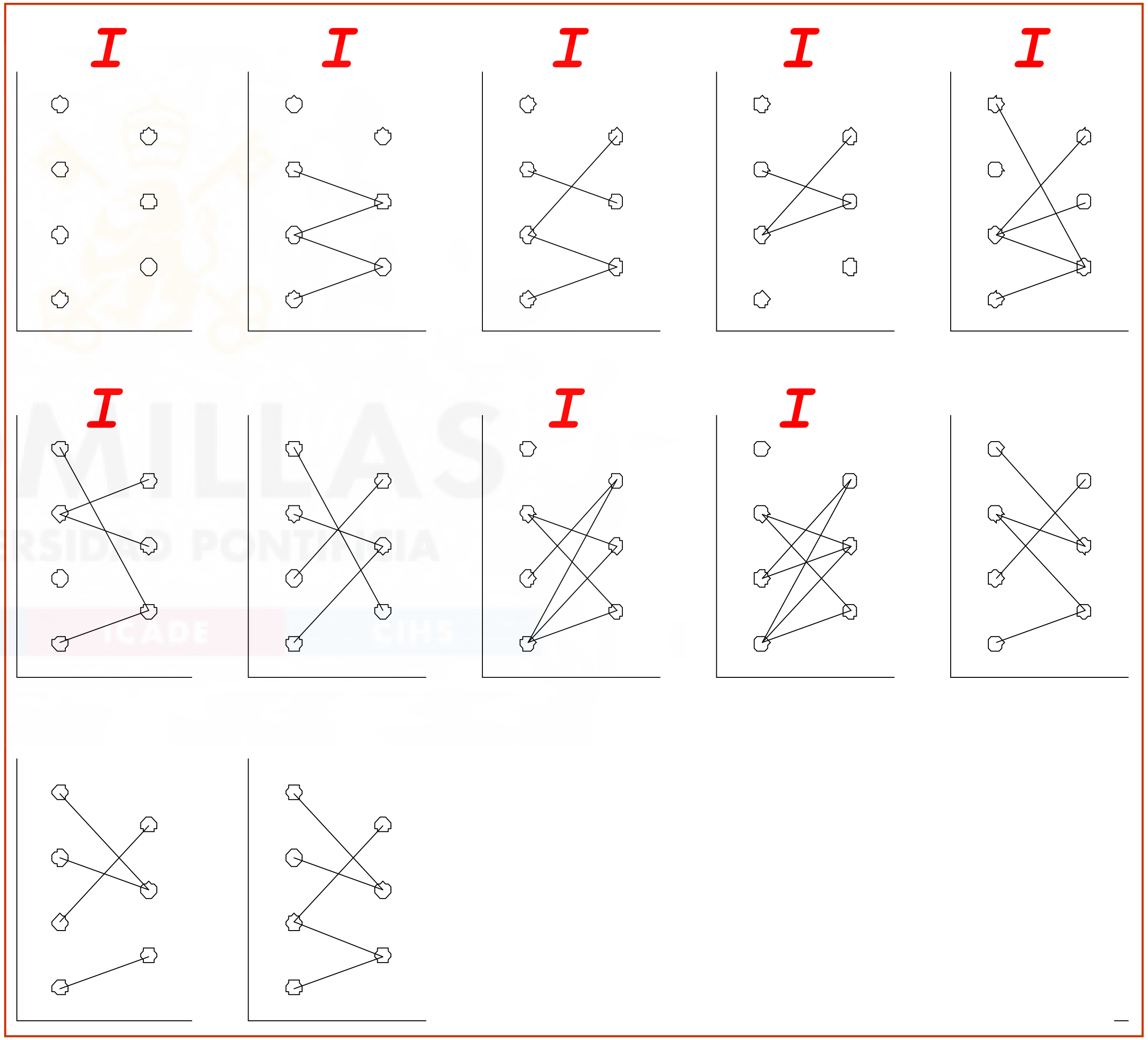
$$x_{ij} \geq 0$$

Fixed-charge transportation problem. Bd Solution

- Possible arcs

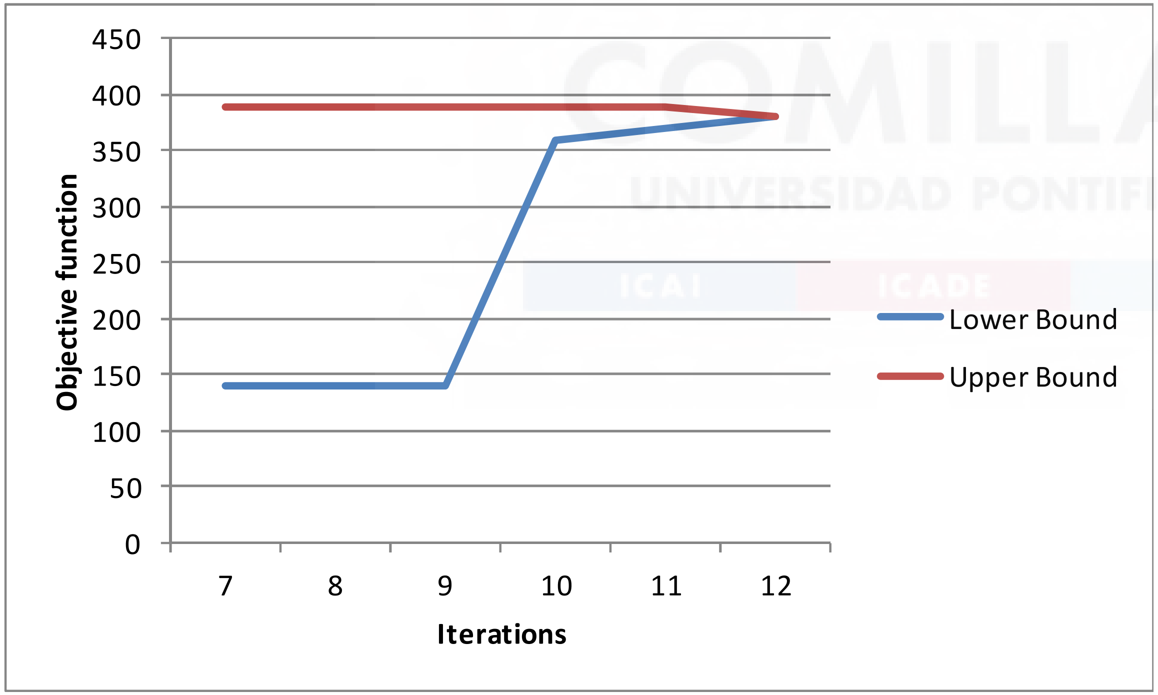


- Solutions along Bd decomposition iterations



Fixed-charge transportation problem. Bd Convergence

Iteration	Lower Bound	Upper Bound
1 a 6	$-\infty$	∞
7	140	390
8	140	390
9	140	390
10	360	390
11	370	390
12	380	380



FCTP solved by Benders decomposition (i)

```
$title Fixed-charge transportation problem (FCTP) solved by Benders decomposition

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
  L      iterations       / l1 * l20 /
  LL(l)  iterations subset
  I      origins          / i1 * i4  /
  J      destinations     / j1 * j3  /

* Begin problem data

parameters
  A(i)   product offer
         / i1 20, i2 30, i3 40, i4 20 /
  B(j)   product demand
         / j1 20, j2 50, j3 30 /

table C(i,j) per unit variable transportation cost
     j1  j2  j3
  i1  1   2   3
  i2  3   2   1
  i3  2   3   4
  i4  4   3   2

table F(i,j) fixed transportation cost
     j1  j2  j3
  i1  10  20  30
  i2  20  30  40
  i3  30  40  50
  i4  40  50  60

* End problem data

abort $(sum[i, A(i)] < sum[j, B(j)]) 'Infeasible problem'

parameters
  BdTol      relative Benders tolerance / 1e-6 /
  Z_Lower    lower bound        / -inf /
  Z_Upper    upper bound        /  inf /
  Y_L  (l,i,j) first stage variables values          in iteration l
  PI_L (l,i,j) dual variables of second stage constraints in iteration l
  Delta(l)   cut type (feasibility 0 optimality 1)    in iteration l
  Z2_L (l)   subproblem objective function value     in iteration l
```

FCTP solved by Benders decomposition (ii)

```
positive variable
   X(i,j)        arc flow

binary   variable
   Y(i,j)        arc investment decision

variables
   Z1            first  stage objective function
   Z2            second stage objective function
   Theta         recourse function

equations
   EQ_Z1           first  stage     objective function
   EQ_Z2           second stage     objective function
   EQ_OBJ          complete problem objective function
   Offer     (i  ) offer  at origin
   Demand    (  j) demand at destination
   FlowLimit(i,j) arc flow limit
   Bd_Cuts     (l) Benders cuts ;

EQ_Z1          .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + Theta ;

EQ_Z2          .. Z2 =e=                             sum[(i,j), C(i,j)*X(i,j)] ;

EQ_OBJ         .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(i,j), C(i,j)*X(i,j)] ;

Offer    (i  ) .. sum[j, X(i,j)] =l= A(i) ;

Demand   (  j) .. sum[i, X(i,j)] =g= B(j) ;

FlowLimit(i,j) .. X(i,j) =l= min[A(i),B(j)] * Y(i,j) ;

Bd_Cuts(ll)    .. Delta(ll) * Theta =g= Z2_L(ll) -
                  sum[(i,j), PI_L(ll,i,j) * min[A(i),B(j)] * (Y_L(ll,i,j) - Y(i,j))] ;

model Master_Bd     / EQ_Z1  Bd_Cuts                / 
model Subproblem_Bd / EQ_Z2  Offer Demand FlowLimit /
model Complete      / EQ_OBJ Offer Demand FlowLimit / ;

X.up(i,j) = min[A(i),B(j)]

* to allow CPLEX correctly detect rays in an infeasible problem
* only simplex method can be used and no preprocessing neither scaling options
* optimality and feasibility tolerances are very small to avoid primal degeneration

file COPT / cplex.opt /
put  COPT putclose 'ScaInd -1' / 'LPMethod 1' / 'PreInd 0' / 'EpOpt 1e-9' / 'EpRHS 1e-9' / ;

Subproblem_Bd.OptFile = 1 ;
```

FCTP solved by Benders decomposition (iii)

```
* parameter initialization

LL      (l) = no ;
Theta.fx    =  0 ;
Delta   (l) =  0 ;
Z2_L    (l) =  0 ;
PI_L(l,i,j) =  0 ;
Y_L (l,i,j) =  0 ;

* Benders algorithm iterations
loop (l $(abs(1-Z_Lower/Z_Upper) > BdTol),

*  solving master problem
   solve Master_Bd using MIP minimizing Z1 ;

*  storing the master solution
   Y_L(l,i,j) = Y.l(i,j) ;

*  fixing first-stage variables and solving subproblem
   Y.fx (i,j) = Y.l(i,j) ;

*  solving subproblem
   solve Subproblem_Bd using RMIP minimizing Z2 ;

*  storing parameters to build a new Benders cut
   if (Subproblem_Bd.ModelStat = 4,
      Delta(l) = 0 ;
      Z2_L (l) = Subproblem_Bd.SumInfes ;
   else
*     updating lower and upper bound
      Z_Lower =              Z1.l ;
      Z_Upper = min(Z_Upper, Z1.l - Theta.l + Z2.l) ;

      Theta.lo = -inf ;
      Theta.up =  inf ;

      Delta(l) =    1 ;
      Z2_L (l) = Subproblem_Bd.ObjVal ;
   ) ;

   PI_L(l,i,j) = FlowLimit.m(i,j) ;

   Y.lo(  i,j) = 0 ;
   Y.up(  i,j) = 1 ;

*  increase the set of Benders cuts
   LL(l) = yes ;
) ;

solve Complete using MIP minimizing Z1
```

Deterministic & Stochastic FCTP

Flows
(second stage)

Investment decisions
(first stage)

Capacity of each origin

Demand of each destination

Flow can pass only for installed connections

Deterministic

$$\min_{x_{ij},y_{ij}} \sum_{ij} (f_{ij}y_{ij} + c_{ij}x_{ij})$$

$$\sum_j x_{ij} \le a_i \quad \forall i$$

$$\sum_i x_{ij} \ge b_j \quad \forall j$$

$$x_{ij} \le M_{ij}y_{ij} \quad \forall ij$$

$$x_{ij} \ge 0, y_{ij} \in \{0,1\}$$

Stochastic

$$\min_{x_{ij}^\omega,y_{ij}} \sum_{ij} \left(f_{ij}y_{ij} + \sum_\omega p_\omega c_{ij}x_{ij}^\omega \right)$$

$$\sum_j x_{ij}^\omega \le a_i \quad \forall i\omega$$

$$\sum_i x_{ij}^\omega \ge b_j^\omega \quad \forall j\omega$$

$$x_{ij}^\omega \le M_{ij}y_{ij} \quad \forall ij\omega$$

$$x_{ij}^\omega \ge 0, y_{ij} \in \{0,1\}$$

Deterministic & Stochastic FCTP

```
$title Deterministic fixed-charge transportation problem (DFCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins           / i1 * i4 /
   J         destinations      / j1 * j3 /


parameters
   A(i)      product offer
             / i1 20, i2 30, i3 40, i4 20 /
   B(j)      product demand
             / j1 20, j2 50, j3 30 /





table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

abort $(sum[i, A(i)] < sum[j, B(j)]) 'Infeasible problem'

positive variable
   X(i,j)       arc flow
binary   variable
   Y(i,j)       arc investment decision
variables
   Z1           objective function

equations
   EQ_OBJ         complete problem objective function
   Offer     (i  ) offer  at origin
   Demand    (  j) demand at destination
   FlowLimit(i,j) arc flow limit ;

EQ_OBJ         .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(i,j), C(i,j)*X(i,j)] ;
Offer    (i  ) .. sum[j, X(i,j)] =l= A(i) ;
Demand   (  j) .. sum[i, X(i,j)] =g= B(j) ;
FlowLimit(i,j) .. X(i,j) =l= min[A(i),B(j)] * Y(i,j) ;

model Complete      / EQ_OBJ Offer Demand FlowLimit / ;

X.up(i,j) = min[A(i),B(j)]

solve Complete using MIP minimizing Z1
```

```
$title Stochastic fixed-charge transportation problem (SFCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins           / i1 * i4 /
   J         destinations      / j1 * j3 /
   S         scenarios         / s1 * s3 /

parameters
   A(i)      product offer        / i1 20, i2 30, i3 40, i4 20 /
   P(s)      scenario probability / s1 0.5, s2 0.3, s3 0.2 /

table B(s,j) product demand
       j1  j2  j3
   s1  21  51  31
   s2  32  22  52
   s3  53  33  23

table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

loop (s, abort $(sum[i, A(i)] < sum[j, B(s,j)]) 'Infeasible problem' )

positive variable
   X(s,i,j)     arc flow
binary   variable
   Y(i,j  )     arc investment decision
variables
   Z1           objective function

equations
   EQ_OBJ           complete problem objective function
   Offer    (s,i  ) offer  at origin
   Demand   (s,  j) demand at destination
   FlowLimit(s,i,j) arc flow limit ;

EQ_OBJ           .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(s,i,j), P(s)*C(i,j)*X(s,i,j)] ;
Offer    (s,i  ) .. sum[j, X(s,i,j)]  =l= A(  i) ;
Demand   (s,  j) .. sum[i, X(s,i,j)]  =g= B(s,j) ;
FlowLimit(s,i,j) .. X(s,i,j) =l= min[A(i),B(s,j)] * Y(i,j) ;

model Complete      / EQ_OBJ Offer Demand FlowLimit / ;

X.up(s,i,j) = min[A(i),B(s,j)]

solve Complete using MIP minimizing Z1
```

Stochastic FCTP with EMP

```
$title Deterministic fixed-charge transportation problem (FCTP)

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
   I         origins           / i1 * i4 /
   J         destinations      / j1 * j3 /

parameters
   A(i)      product offer
             / i1 20, i2 30, i3 40, i4 20 /
   B(j)      product demand
             / j1 11, j2 44, j3 66 /

table C(i,j) per unit variable transportation cost
       j1  j2  j3
   i1   1   2   3
   i2   3   2   1
   i3   2   3   4
   i4   4   3   2

table F(i,j) fixed transportation cost
       j1  j2  j3
   i1  10  20  30
   i2  20  30  40
   i3  30  40  50
   i4  40  50  60

positive variable
   X(i,j)       arc flow

binary   variable
   Y(i,j)       arc investment decision

variables
   Z1           objective function

equations
   EQ_OBJ         complete problem objective function
   Offer    (i  ) offer  at origin
   Demand   ( j) demand at destination
   FlowLimit(i,j) arc flow limit ;

EQ_OBJ         .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(i,j), C(i,j)*X(i,j)] ;

Offer    (i  ) .. sum[j, X(i,j)] =l= A(i) ;

Demand   (  j) .. sum[i, X(i,j)] =g= B(j) ;

FlowLimit(i,j) .. X(i,j) =l= 100 * Y(i,j) ;

model Complete      / all / ;

X.up(i,j) = 100
```

```
set          S      scenarios             / s1 * s3 /
parameter P(s)      scenario probability / s1 0.5, s2 0.3, s3 0.2 /
          YS(s,i,j) arc investment decision
          XS(s,i,j) arc flow

table  BS(s,j) product demand
       j1  j2  j3
   s1  21  51  31
   s2  32  22  52
   s3  53  33  23

file     emp / '%emp.info%' / ; emp.pc=2
put      emp
put '* problem %gams.i%' / "jrandvar "
loop (j,
   put B.tn(j) " "
)
loop (s,
   put P(s)
   loop (j,
      put BS(s,j)
   )
)
put / "stage 2 B X Offer Demand FlowLimit"
putclose emp

set dict / s . scenario . ''
           B . randvar  . BS
           X . level    . XS
           Y . level    . YS /

loop (s, abort $(sum[i, A(i)] < sum[j, BS(s,j)]) 'Infeasible problem' ) ;

solve Complete minimizing Z1 using emp scenario dict

display XS, YS
```

Stochastic FCTP solved by Benders decomposition (i)

```
$title Stochastic Fixed-charge transportation problem (FCTP) solved by Benders decomposition

* relative optimality tolerance in solving MIP problems
option OptcR = 0

sets
  L      iterations       / l1 * l70 /
  LL(l)   iterations subset
  I      origins          / i1 * i4  /
  J      destinations     / j1 * j3  /
  S      scenarios        / s1 * s3  /

* Begin problem data

parameters
  A(i)    product offer
          / i1 20, i2 30, i3 40, i4 30 /
  P(s)    scenario probability / s1 0.5, s2 0.3, s3 0.2 /

table B(s,j) product demand
     j1  j2  j3
  s1  21  51  31
  s2  32  22  52
  s3  53  33  23

table C(i,j) per unit variable transportation cost
     j1  j2  j3
  i1  1   2   3
  i2  3   2   1
  i3  2   3   4
  i4  4   3   2

table F(i,j) fixed transportation cost
     j1  j2  j3
  i1  10  20  30
  i2  20  30  40
  i3  30  40  50
  i4  40  50  60

* End problem data

loop (s, abort $(sum[i, A(i)] < sum[j, B(s,j)]) 'Infeasible problem' )

parameters
  BdTol        relative Benders tolerance / 1e-6 /
  Z_Lower       lower bound       / -inf /
  Z_Upper       upper bound       /  inf /
  Y_L  (l,  i,j) first stage variables values            in iteration l
  PI_L (l,s,i,j) dual variables of second stage constraints in iteration l
  Delta(l)     cut type (feasibility 0 optimality 1)     in iteration l
  Z2_L (l)     subproblem objective function value       in iteration l
```

Stochastic FCTP solved by Benders decomposition (ii)

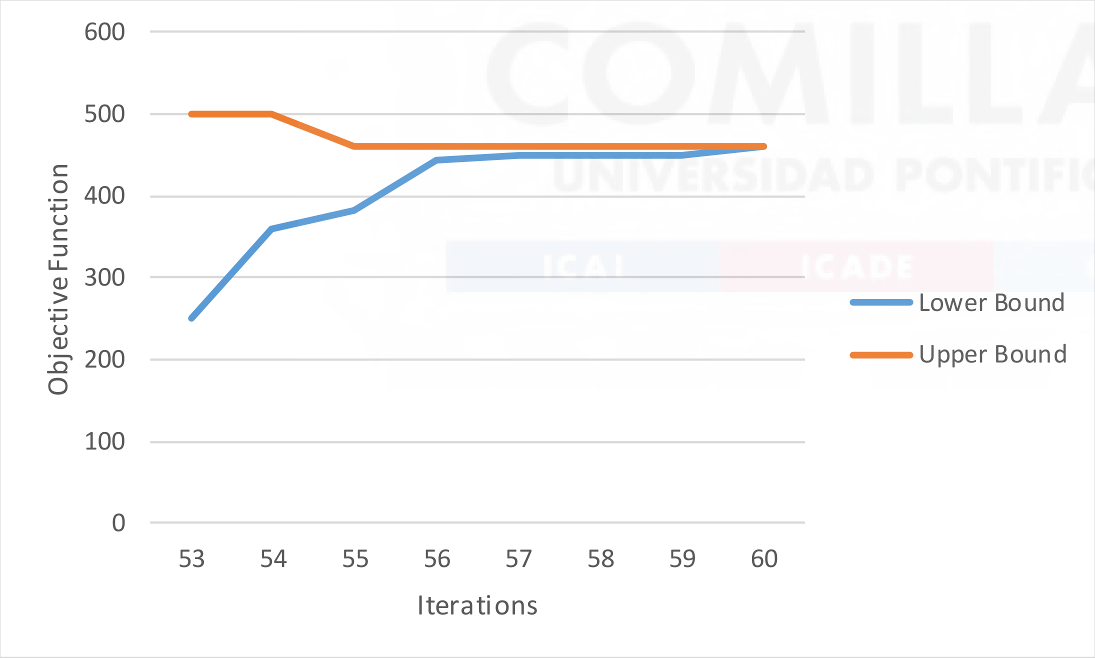
```
positive variable
   X(s,i,j)     arc flow

binary   variable
   Y(i,j)       arc investment decision

variables
   Z1           first  stage objective function
   Z2           second stage objective function
   Theta        recourse function

equations
   EQ_Z1            first  stage     objective function
   EQ_Z2            second stage     objective function
   EQ_OBJ           complete problem objective function
   Offer    (s,i  ) offer  at origin
   Demand   (s,   j) demand at destination
   FlowLimit(s,i,j) arc flow limit
   Bd_Cuts      (l) Benders cuts ;

EQ_Z1            .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + Theta ;

EQ_Z2            .. Z2 =e=                            sum[(s,i,j), P(s)*C(i,j)*X(s,i,j)] ;

EQ_OBJ           .. Z1 =e= sum[(i,j), F(i,j)*Y(i,j)] + sum[(s,i,j), P(s)*C(i,j)*X(s,i,j)] ;

Offer    (s,i  ) .. sum[j, X(s,i,j)]  =l= A(  i) ;

Demand   (s,  j) .. sum[i, X(s,i,j)]  =g= B(s,j) ;

FlowLimit(s,i,j) .. X(s,i,j) =l= min[A(i),B(s,j)] * Y(i,j) ;

Bd_Cuts(ll)      .. Delta(ll) * Theta =g= Z2_L(ll) -
                    sum[(s,i,j), PI_L(ll,s,i,j) * min[A(i),B(s,j)] * (Y_L(ll,i,j) - Y(i,j))] ;

model Master_Bd      / EQ_Z1  Bd_Cuts                /
model Subproblem_Bd / EQ_Z2  Offer Demand FlowLimit /
model Complete       / EQ_OBJ Offer Demand FlowLimit / ;

X.up(s,i,j) = min[A(i),B(s,j)]

* to allow CPLEX correctly detect rays in an infeasible problem
* only simplex method can be used and no preprocessing neither scaling options
* optimality and feasibility tolerances are very small to avoid primal degeneration

file COPT / cplex.opt /
put  COPT putclose 'ScaInd -1' / 'LPMethod 1' / 'PreInd 0' / 'EpOpt 1e-9' / 'EpRHS 1e-9' / ;

Subproblem_Bd.OptFile = 1 ;
```

Stochastic FCTP solved by Benders decomposition (iii)

```
* parameter initialization

LL         (l) = no ;
Theta.fx       =  0 ;
Delta      (l) =  0 ;
Z2_L       (l) =  0 ;
PI_L(l,s,i,j) =  0 ;
Y_L (l,  i,j) =  0 ;

* Benders algorithm iterations
loop (l $(abs(1-Z_Lower/Z_Upper) > BdTol),

*  solving master problem
   solve Master_Bd using MIP minimizing Z1 ;

*  storing the master solution
   Y_L(l,i,j) = Y.l(i,j) ;

*  fixing first-stage variables and solving subproblem
   Y.fx (i,j) = Y.l(i,j) ;

*  solving subproblem
   solve Subproblem_Bd using RMIP minimizing Z2 ;

*  storing parameters to build a new Benders cut
   if (Subproblem_Bd.ModelStat = 4,
      Delta(l) = 0 ;
      Z2_L (l) = Subproblem_Bd.SumInfes ;
   else
*     updating lower and upper bound
      Z_Lower =                Z1.l ;
      Z_Upper = min(Z_Upper, Z1.l - Theta.l + Z2.l) ;

      Theta.lo = -inf ;
      Theta.up =  inf ;

      Delta(l) =     1 ;
      Z2_L (l) = Subproblem_Bd.ObjVal ;
   ) ;

   PI_L(l,s,i,j) = FlowLimit.m(s,i,j) ;

   Y.lo(    i,j) = 0 ;
   Y.up(    i,j) = 1 ;

*  increase the set of Benders cuts
   LL(l) = yes ;
) ;

solve Complete using MIP minimizing Z1 ;
```

Stochastic FCTP. Bd Convergence

Iteration	Lower Bound	Upper Bound
1 a 52	-inf	+inf
53	250.0	498.9
54	359.5	498.9
55	381.5	461.5
56	442.9	461.3
57	449.3	461.3
58	449.3	461.3
59	449.3	461.3
60	461.3	461.3



Benders decomposition in Julia

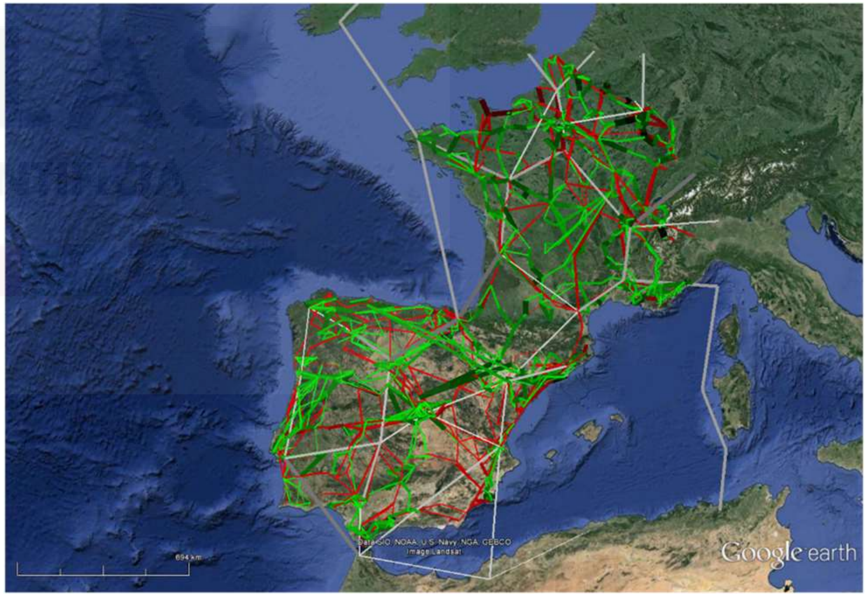
```
# Benders algorithm iterations
for l in 1:L
  if abs(1-Z_Lower/Z_Upper) > BdTol

    # solving master problem
    #print("\nThe current master problem model is \n", Master_Bd)
    Master_Bd_status = solve(Master_Bd)
    Z1   = getobjectivevalue(Master_Bd)

    for i in 1:I
      for j in 1:J
        # storing the master solution
        Y_L[l,i,j] = getvalue(Y[i,j])
        Y_J[  i,j] = getvalue(Y[i,j])
        chgConstrRHS(FlowLimit[i,j], min(A[i],B[j])*Y_J[i,j])
      end
    end

    # solving subproblem
    #print("\nThe current subproblem model is \n", Subproblem_Bd)
    Subproblem_Bd_status = solve(Subproblem_Bd)
    Z2       = getobjectivevalue(Subproblem_Bd)
    Z2_L[l]  = Z2

    # storing parameters to build a new Benders cut
    if Subproblem_Bd_status == :Infeasible
      # the problem has to be feasible because I am not able to obtain the sum of infeasibilities of the phase I
      Delta[l] =  0
    else
      # updating lower and upper bound
      Z_Lower =              Z1
      Z_Upper = min(Z_Upper, Z1 - getvalue(Theta) + Z2)
      println("Iteration ", l, " Status of the master problem is ", Master_Bd_status, " with o.f. Master_Bd ",     Z1, " Z_Lower = ", Z_Lower)
      println("Iteration ", l, " Status of the subproblem is ", Subproblem_Bd_status, " with o.f. Subproblem_Bd ", Z2, " Z_Upper = ", Z_Upper)

      setlowerbound(Theta, - Inf)
      setupperbound(Theta,   Inf)

      Delta[l] =  1
    end

    for i in 1:I
      for j in 1:J
        # storing the master solution
        PI_L[l,i,j] = getdual(FlowLimit[i,j])
      end
    end

    # Benders cuts
    @constraint(Master_Bd, Bd_Cuts[l], Delta[l] * Theta >= Z2_L[l] - sum(PI_L[l,i,j] * min(A[i],B[j]) * (Y_L[l,i,j] - Y[i,j]) for i=1:I,j=1:J))

  end
end
```

Benders decomposition in Python

```
# Benders algorithm
mMaster_Bd.vTheta.fix(0)
for l in mMaster_Bd.l:
    if abs(1-Z_Lower/Z_Upper) > BdTol or l == mMaster_Bd.l.first():

        # solving master problem
        SolverResultsMst = Solver.solve(mMaster_Bd)
        Z1               = mMaster_Bd.eCostMst.expr()

        for i,j in mFCTP.i*mFCTP.j:
            # storing the master solution
            Y_L[l,i,j] = pyo.value(mMaster_Bd.vY[i,j])
            # fix investment decision for the subproblem
            mFCTP.vY[i,j].fix(Y_L[l,i,j])

        # solving subproblem
        SolverResultsSbp    = Solver.solve(mFCTP)
        Z2                  = mFCTP.eCostSubp.expr()
        Z2_L[l]             = Z2

        # storing parameters to build a new Benders cut
        if SolverResultsSbp.solver.termination_condition == TerminationCondition.infeasible:
            # the problem has to be feasible because I am not able to obtain the sum of infeasibilities of the phase I
            Delta[l] =  0
        else:
            # updating lower and upper bound
            Z_Lower =              Z1
            Z_Upper = min(Z_Upper, Z1 - pyo.value(mMaster_Bd.vTheta) + Z2)
            print('Iteration ', l, ' Z_Lower ... ', Z_Lower)
            print('Iteration ', l, ' Z_Upper ... ', Z_Upper)

            mMaster_Bd.vTheta.free()

            Delta[l] =  1

        for i,j in mFCTP.i*mFCTP.j:
            PI_L[l,i,j] = mFCTP.dual[mFCTP.eFlowLimit[i,j]]

        mMaster_Bd.vY.unfix()

        # add one cut
        mMaster_Bd.ll.add(l)
        ll = mMaster_Bd.ll
        mMaster_Bd.eBd_Cuts = Constraint(mMaster_Bd.ll, rule=eBd_Cuts, doc='Benders cuts')
```

Transmission expansion planning

- G. Micheli, M.T. Vespucci, M. Stabil, C. Puglisi, A. Ramos *A two-stage stochastic MILP model for generation and transmission expansion planning with high shares of renewables* Energy Systems October 2020 10.1007/s12667-020-00404-w
- S. Lumbreras, A. Ramos *How to Solve the Transmission Expansion Planning Problem Faster: Acceleration Techniques Applied to Benders Decomposition* IET Generation, Transmission & Distribution 10: 2351-2359, Jul 2016 10.1049/iet-gtd.2015.1075
- S. Lumbreras, A. Ramos *Transmission Expansion Planning using an Efficient Version of Benders' Decomposition. A Case Study* IEEE PowerTech. Grenoble, France. June 2013 10.1109/PTC.2013.6652091



Electrical layout of an offshore wind farm

- Th. Marge, S. Lumbreras, A. Ramos, and B.F. Hobbs *Integrated offshore wind farm design: Optimizing micro-siting and cable layout simultaneously* Wind Energy 22, 1684–1698 November 2019 10.1002/we.2396
- S. Lumbreras and A. Ramos *Optimal Design of the Electrical Layout of an Offshore Wind Farm: a Comprehensive and Efficient Approach Applying Decomposition Strategies* IEEE Transactions on Power Systems 28 (2): 1434-1441, May 2013 10.1109/TPWRS.2012.2204906
- S. Lumbreras and A. Ramos *Optimal Design of the Electrical Layout of an Offshore Wind Farm: a Comprehensive and Efficient Approach Applying Decomposition Strategies* IEEE Transactions on Power Systems 28 (2): 1434-1441, May 2013 10.1109/TPWRS.2012.2204906
- S. Lumbreras and A. Ramos *Offshore Wind Farm Electrical Design: A Review* Wind Energy 16 (3): 459-473 April 2013 10.1002/we.1498
- S. Lumbreras, A. Ramos *A Benders' Decomposition Approach for Optimizing the Electric System of Offshore Wind Farms* IEEE PowerTech. Trondheim, Norway June 2011 (Presentation) 10.1109/PTC.2011.6019371
- M. Banzo and A. Ramos *Stochastic Optimization Model for Electric Power System Planning of Offshore Wind Farms* IEEE Transactions on Power Systems 26 (3): 1338-1348 Aug 2011 10.1109/TPWRS.2010.2075944

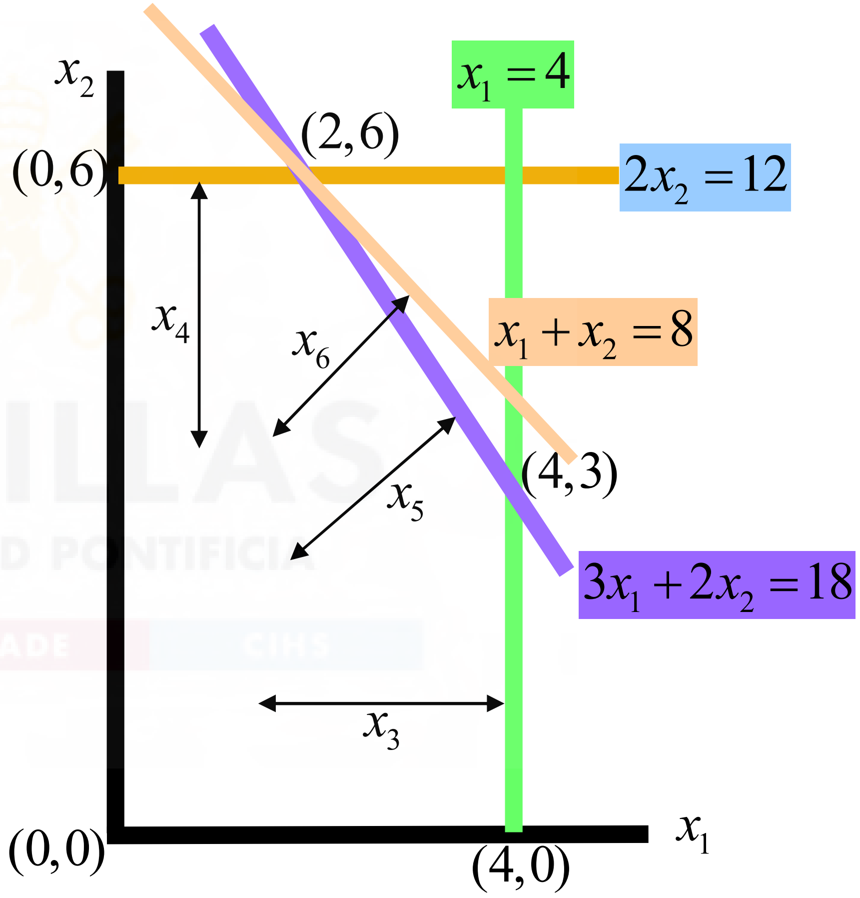
Achtung! Achtung!

- Degeneracy in LP problems
  - In real cases, finding multiple optima (degeneracy in primal problem or multiple dual solutions) with the same or different basis is frequent. Given that decomposition techniques are based on dual variables, you must be very careful in their computation
    - Careful implementation (scaling) is crucial to avoid numerical problems
  - For example, in the hydrothermal scheduling model formulated as LP it can exist spatial degeneracy (the system can produce with one hydro plant or another) and temporal (the system can produce now or in the future)

Primal degeneracy

- Variable x_6 is degenerate (basic variable with value 0)

$$
\begin{aligned}
&\min z = -3x_1 - 5x_2 \\
&x_1 + x_3 = 4 \\
&2x_2 + x_4 = 12 \\
&3x_1 + 2x_2 + x_5 = 18 \\
&x_1 + x_2 + x_6 = 8 \\
&x_1,\ x_2,\ x_3,\ x_4,\ x_5,\ x_6 \geq 0
\end{aligned}
$$



When to use Benders decomposition for computational advantage?

- Variables x_1 complicate the solution of the problem
- Implicitly $n_1 \ll n_2$
- Number of iterations related to n_1
- Matrix structure induces separability of subproblems
- Master and subproblem have different natures
  - Master in discrete variables (MIP)
  - Subproblem with (convex) nonlinear objective function (NLP)
- Benders decomposition needs convex o.f. and convex feasible region of the subproblem

$$
\begin{array}{ll}
c_1 \in \mathbb{R}^{n_1} & c_2 \in \mathbb{R}^{n_2} \\
A_1 \in \mathbb{R}^{m_1 \times n_1} & A_2 \in \mathbb{R}^{m_2 \times n_2} \\
b_1 \in \mathbb{R}^{m_1} & b_2 \in \mathbb{R}^{m_2} \\
x_1 \in \mathbb{R}^{n_1} & x_2 \in \mathbb{R}^{n_2}
\end{array}
$$

$$
\begin{array}{lllr}
\min_{x_1, x_2} (c_1^T x_1 + c_2^T x_2) & & & \\
A_1 x_1 & & & = b_1 \\
B_1 x_1 & + A_2 x_2 & & = b_2 \\
x_1, & x_2 & & \geq 0
\end{array}
$$

Pros and cons

- Main advantage
  - At any iteration, the current solution is valid
- Main drawback
  - Slow convergence, i.e., a large number of iterations to converge to a small tolerance

6

Nested Benders decomposition

Nested Benders decomposition (i)

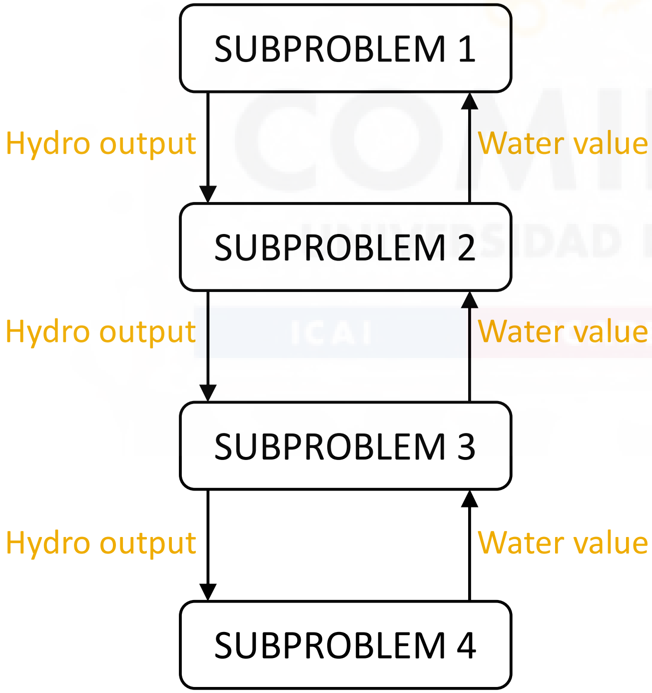
- Recursive application of the decomposition technique.
- Let us see the PL-P problem:

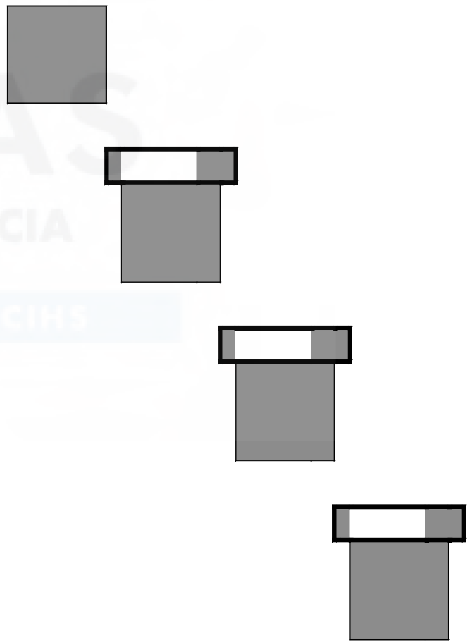
$$\min_{x_p} \sum_{p=1}^{P} c_p^T x_p$$
$$B_{p-1} x_{p-1} + A_p x_p = b_p \quad p = 1, \ldots, P$$
$$x_p \geq 0$$
$$B_0 \equiv 0$$

- We apply Benders decomposition:
  - Stage 1 master, stages 2 to *P* subproblem
  - We decompose the subproblem that begins in stage 2
    - Stage 2 master, stages 3 to *P* subproblem
    - We decompose the subproblem that begins in stage 3
      - Stage 3 master, stages 4 to *P* subproblem
      - We decompose the subproblem that begins in stage 4

Nested Benders decomposition (ii)

- At any stage *p*
  - The problem of this stage is solved
  - As a master, it receives cuts from *p*+1 and passes the solution to *p*+1,
  - As a subproblem, it builds cuts from *p*-1 and receives the solution from *p*-1.





Nested Benders decomposition. Cut deduction (i)

- Let it be this problem with 4 stages

$$
\begin{aligned}
\min_{x_1,x_2,x_3,x_4} \; & c_1^T x_1 + c_2^T x_2 + c_3^T x_3 + c_4^T x_4 \\
& A_1 x_1 = b_1 \\
& B_1 x_1 + A_2 x_2 = b_2 \\
& B_2 x_2 + A_3 x_3 = b_3 \\
& B_3 x_3 + A_4 x_4 = b_4 \\
& x_1, \; x_2, \; x_3, \; x_4 \geq 0
\end{aligned}
$$

- Solve stage 4 for a proposal from stage 3

$$
\begin{aligned}
\min_{x_4} \; & c_4^T x_4 \\
& A_4 x_4 = b_4 - B_3 x_3^l \quad : \pi_4 \\
& x_4 \geq 0
\end{aligned}
$$

Nested Benders decomposition. Cut deduction (ii)

- Master problem of stage 3 will be

$$\min_{x_3,\theta_4} c_3^T x_3 + \theta_4$$
$$A_3 x_3 = b_3 - B_2 x_2^l \quad : \pi_3$$
$$\theta_4 + \pi_4^T B_3 x_3 \geq \pi_4^T b_4 \quad : \eta_3$$
$$x_3 \geq 0$$

- Objective function of dual problem is

$$\max_{\pi_3,\eta_3} \pi_3^T \left(b_3 - B_2 x_2^l\right) + \eta_3^T \left(\pi_4^T b_4\right)$$

- Master problem of simultaneous stages 3 and 4 will be

$$\min_{x_3,x_4} c_3^T x_3 + c_4^T x_4$$
$$A_3 x_3 = b_3 - B_2 x_2^l \quad : \pi_3'$$
$$B_3 x_3 + A_4 x_4 = b_4 \quad : \mu_3$$
$$x_3, x_4 \geq 0$$

- Objective function of dual problem is

$$\max_{\pi_3',\mu_3} \pi_3'^T \left(b_3 - B_2 x_2^l\right) + \mu_3^T b_4$$

Nested Benders decomposition. Cut deduction (iii)

- Objective function of problem of stage 3 is a lower bound of the objective function of problem of stages 3 and 4 for each value of x_2

$$\pi_3'^T (b_3 - B_2 x_2) + \mu_3^T b_4 \geq \pi_3^T (b_3 - B_2 x_2) + \eta_3^T (\pi_4^T b_4)$$

- The cut to introduce in the stage 2 assuming stages 3 and 4 are a single subproblem would be

$$\theta_3 \geq \pi_3'^T (b_3 - B_2 x_2) + \mu_3^T b_4 \geq \pi_3^T (b_3 - B_2 x_2) + \eta_3^T (\pi_4^T b_4)$$

- The following cut is a valid cut

$$\theta_3 + \pi_3^T B_2 x_2 \geq \pi_3^T b_3 + \eta_3^T (\pi_4^T b_4)$$

Nested Benders decomposition. Cut deduction (iv)

- Master problem of stage 2 is

$$\begin{aligned}
&\min_{x_2,\theta_3} c_2^T x_2 + \theta_3 \\
&A_2 x_2 = b_2 - B_1 x_1^l \qquad : \pi_2 \\
&\theta_3 + \pi_3^T B_2 x_2 \ge \pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right) \quad : \eta_2 \\
&x_2 \ge 0
\end{aligned}$$

- Objective function of dual problem is

$$\max_{\pi_2,\eta_2} \pi_2^T \left(b_2 - B_1 x_1^l\right) + \eta_2^T \left[\pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right)\right]$$

- Master problem of simultaneous stages 2, 3 and 4 and the objective function of the dual is

$$\begin{aligned}
&\min_{x_2,x_3,x_4} c_2^T x_2 + c_3^T x_3 + c_4^T x_4 \\
&A_2 x_2 = b_2 - B_1 x_1^l \quad : \pi_2' \\
&B_2 x_2 + A_3 x_3 = b_3 \quad : \mu_2 \\
&B_3 x_3 + A_4 x_4 = b_4 \quad : \mu_3 \\
&x_2, x_3, x_4 \ge 0
\end{aligned}$$

$$\max_{\pi_2',\mu_2,\upsilon_2} \pi_2'^T \left(b_2 - B_1 x_1^l\right) + \mu_2^T b_3 + \mu_3^T b_4$$

Nested Benders decomposition. Cut deduction (v)

- The cut to introduce in the stage 1 would be

$$\theta_2 \geq \pi_2'^T (b_2 - B_1 x_1) + \mu_2^T b_3 + \mu_3^T b_4 \geq \pi_2^T (b_2 - B_1 x_1) + \eta_2^T \left[\pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right)\right]$$

- The following cut is a valid cut

$$\theta_2 + \pi_2^T B_1 x_1 \geq \pi_2^T b_2 + \eta_2^T \left[\pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right)\right]$$

- Cuts for stages 2, 3 and 4 are

$$\begin{aligned}
&\theta_4 + \pi_4^T B_3 x_3 \geq \pi_4^T b_4 \\
&\theta_3 + \pi_3^T B_2 x_2 \geq \pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right) \\
&\theta_2 + \pi_2^T B_1 x_1 \geq \pi_2^T b_2 + \eta_2^T \left[\pi_3^T b_3 + \eta_3^T \left(\pi_4^T b_4\right)\right]
\end{aligned}$$

$$\theta_{p+1} + \pi_{p+1}^T B_p x_p \geq q_p = \pi_{p+1}^T b_{p+1} + \eta_{p+1}^T q_{p+1}$$

Nested Benders decomposition. Alternative cuts

- Writing the expressions of linearization around a point

$$\theta_4 + \pi_4^T B_3 x_3 \geq f_4^l + \pi_4^T B_3 x_3^l$$
$$\theta_4 \geq f_4^l + \pi_4^T B_3 (x_3^l - x_3)$$

$$\theta_3 \geq \pi_3^T (b_3 - B_2 x_2) + \eta_3^T (\pi_4^T b_4) =$$
$$= \pi_3^T (b_3 - B_2 x_2 + B_2 x_2^l - B_2 x_2^l) + \eta_3^T (\pi_4^T b_4) =$$
$$= \pi_3^T (b_3 - B_2 x_2^l) + \pi_3^T B_2 (x_2^l - x_2) + \eta_3^T (\pi_4^T b_4) =$$
$$= f_3^l + \pi_3^T B_2 (x_2^l - x_2)$$

$$\theta_{p+1} \geq f_{p+1}^l + \pi_{p+1}^T B_p (x_p^l - x_p)$$
$$\theta_{p+1} + \pi_{p+1}^T B_p x_p \geq f_{p+1}^l + \pi_{p+1}^T B_p x_p^l$$

Nested Benders decomposition

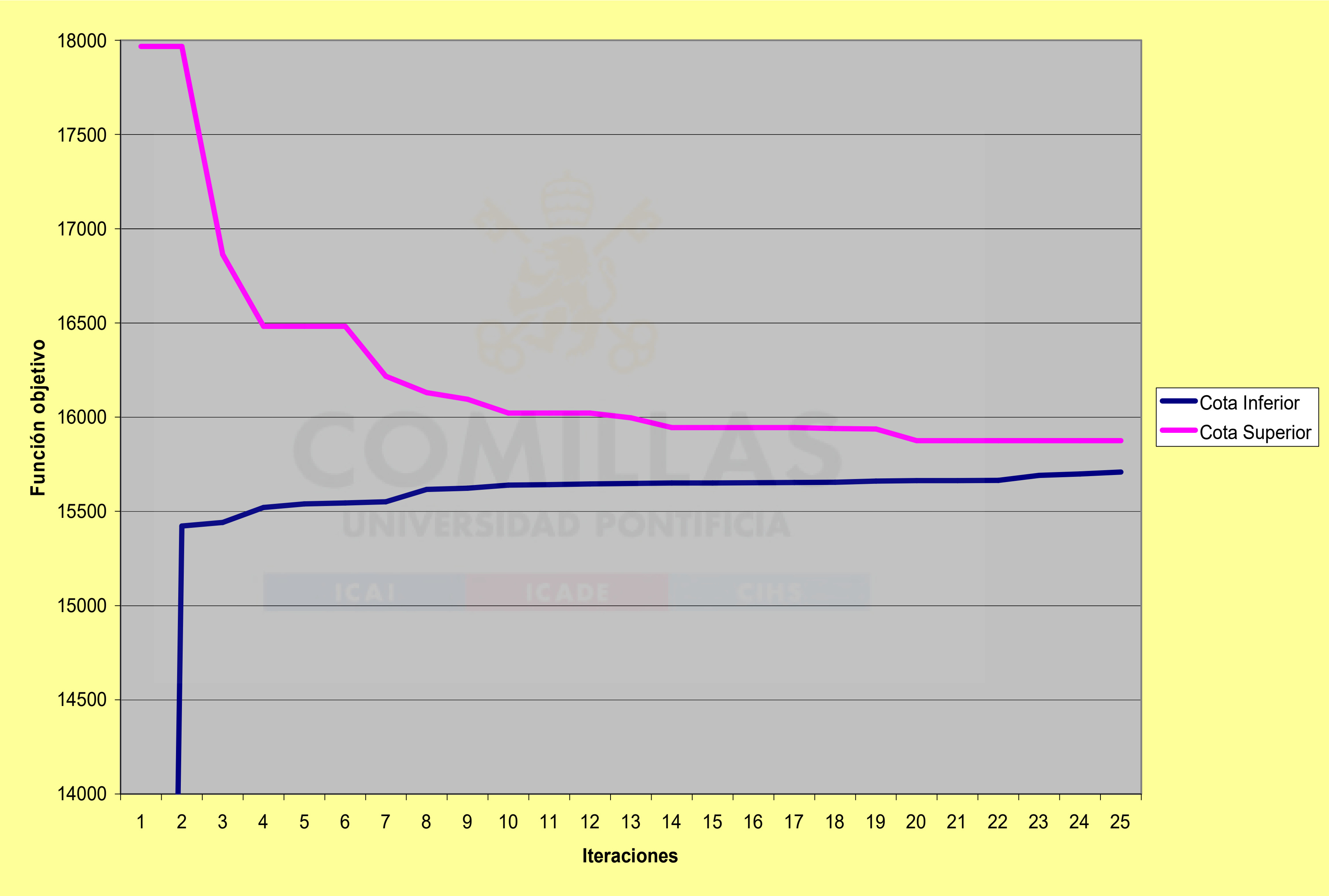
- Generic problem to solve

$$\begin{aligned}
&\min_{x_p,\theta_{p+1}} c_p^T x_p + \theta_{p+1} \\
&A_p x_p = b_p - B_{p-1} x_{p-1}^l && : \pi_p \\
&\pi_{p+1}^{lT} B_p x_p + \theta_{p+1} \ge q_p = \pi_{p+1}^{lT} b_{p+1} + \eta_{p+1}^{lT} q_{p+1} && : \eta_p \quad l = 1,\dots,j \\
&x_p \ge 0 \\
&\theta_{P+1} \equiv 0 \\
&B_0 \equiv 0 \\
&\pi_{P+1}^l \equiv 0 \\
&\eta_{P+1}^l \equiv 0
\end{aligned}$$

$$\begin{aligned}
&\min_{x_p,\theta_{p+1}} c_p^T x_p + \theta_{p+1} \\
&A_p x_p = b_p - B_{p-1} x_{p-1}^l && : \pi_p \\
&\pi_{p+1}^{lT} B_p x_p + \theta_{p+1} \ge f_{p+1}^l + \pi_{p+1}^{lT} B_p x_p^l && : \eta_p \quad l = 1,\dots,j \\
&x_p \ge 0 \\
&\theta_{P+1} \equiv 0 \\
&B_0 \equiv 0 \\
&\pi_{P+1}^l \equiv 0 \\
&\eta_{P+1}^l \equiv 0
\end{aligned}$$

- Problem converges when first stage does

Convergence of a hydrothermal scheduling model



7

Dantzig-Wolfe decomposition

Dantzig-Wolfe decomposition

- Dantzig-Wolfe decomposition or
  - Dual decomposition (because dual information is sent),
  - Price decomposition (because master assigns prices),
  - Column generation (master increases the number of variables in each iteration),
  - Inner linearization (points are linear combination of the vertices)
- Splits PL-2 in master and subproblem

Dantzig-Wolfe decomposition

- First-stage constraints complicate the solution

$$\min_{x_1} c_1^T x_1$$
$$A_1 x_1 = b_1$$
$$A_2 x_1 = b_2$$
$$x_1 \geq 0$$

- Implicitly $m_1 \ll m_2$
- It needs that o. f. and feasible region of the first-stage constraints will be convex
- Subproblems can be nonconvex (MIP, NLP)
- Applications:
  - Decentralized planning with central coordination
  - Unit commitment

DW Algorithm deduction (i)

- Two-stage linear programming PL-2 can be expressed as

$$\min_{x_1} c_1^T x_1$$
$$A_1 x_1 = b_1$$
$$x_1 \in K$$

Complicating (first stage)

Being K the region defined as $K = \{x_1 \mid A_2 x_1 = b_2, x_1 \geq 0\}$

- Every point of the polyhedron can be written as linear convex combination of their vertices

$$K = \left\{ \sum_{l=1}^{\nu} x_1^l \lambda_l \mid \sum_{l=1}^{\nu} \lambda_l = 1, \lambda_l \geq 0 \right\}$$

DW Algorithm deduction (ii)

- Therefore, the complete (master) problem PL-2 can be expressed as

$$
\begin{aligned}
&\min_{\lambda_l} \sum_{l=1}^{\nu} (c_1^T x_1^l)\lambda_l \\
&\sum_{l=1}^{\nu} (A_1 x_1^l)\lambda_l = b_1 \quad : \pi_2 \\
&\sum_{l=1}^{\nu} \lambda_l = 1 \quad : \mu \\
&\lambda_l \geq 0 \quad l = 1, \ldots, \nu
\end{aligned}
$$

- Instead of computing all the vertices they are introduced iteratively
- The original complete problem is solved over a feasible region K bigger at each iteration, corresponding to the second set of constraints (the complicating constraints).
- Objective function decreases monotonically in each iteration when introducing a new variable. It is an upper bound of that of the complete problem.

DW Algorithm deduction (iii)

- Optimality condition for the master problem: reduced costs of the nonbasic variables ≥ 0 in a minimization problem

$$c_1^T x_1^* - \begin{pmatrix} \pi_2^T & \mu \end{pmatrix} \begin{pmatrix} A_1 x_1^* \\ 1 \end{pmatrix} \geq 0$$

$$\theta_2 = \min_{x_1 \in K} \left(c_1^T - \pi_2^T A_1 \right) x_1 - \mu$$

$$\begin{aligned} \theta_2 = \quad & \min_{x_1} \left(c_1^T - \pi_2^{jT} A_1 \right) x_1 - \mu^j \\ & A_2 x_1 = b_2 \\ & x_1 \geq 0 \end{aligned}$$

Second stage

- Subproblem obtains in each iteration the vertex with lower reduced cost, and it is incorporated into the master. When the objective function of the subproblem is positive or zero the optimum has been reached.

DW Relaxed Master and Subproblem

- DW Relaxed Master

$$\min_{\lambda_l} \sum_{l=1}^{j} (c_1^T x_1^l) \lambda_l$$

$$\sum_{l=1}^{j} (A_1 x_1^l) \lambda_l = b_1 \quad : \pi_2$$

$$\sum_{l=1}^{j} \lambda_l = 1 \quad : \mu$$

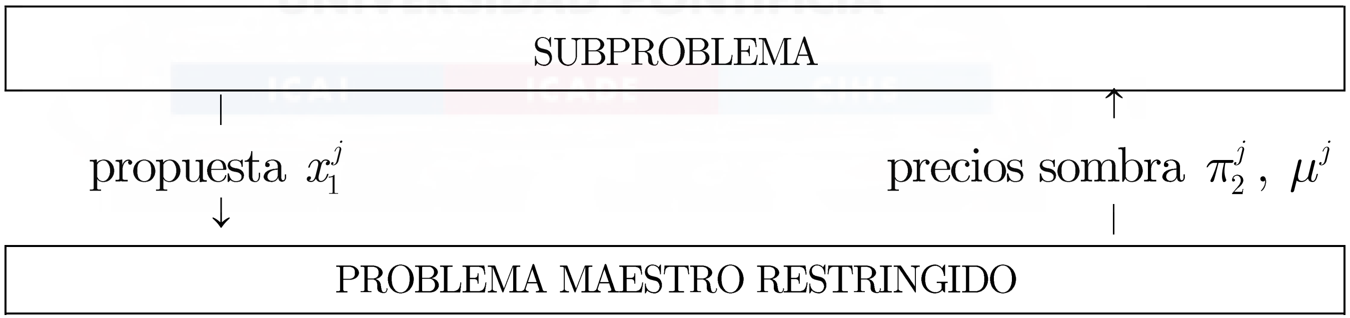
$$\lambda_l \geq 0 \quad l = 1, \ldots, j$$

DW Subproblem

$$\theta_2 = \min_{x_1} \left(c_1^T - \pi_2^{jT} A_1 \right) x_1 - \mu^j$$

$$A_2 x_1 = b_2$$

$$x_1 \geq 0$$



DW Relaxed Master and Subproblem

- DW Relaxed Master:
  - Linear combination of the first-stage solutions and constraints (inner linearization)
  - Generates economic signals (dual variables)
  - Adds a new variable at each iteration
  - May have feasibility problems in the first iterations
  - Must be convex
- DW Subproblem:
  - Objective function is reduced costs
  - Second-stage constraints
  - Plans decentralizedly internalizing the economic signals
  - Each iteration changes the objective function
  - Can be nonconvex

DW Optimal solution

- The optimal solution is the linear combination of the solutions of all the iterations

$$x_1^* = \sum_{l=1}^{j} x_1^l \lambda_l$$

$$z_1^* = \sum_{l=1}^{j} (c_1^T x_1^l) \lambda_l$$

Comparison between Bd and DW

- Original linear problem PL-2: $(m_1 + m_2) \times (n_1 + n_2)$
- For each Bd iteration
  - Increases in 1 the number of constraints of the master problem
  - Modifies the RHS of the constraints of the subproblem
  - Bd Relaxed Master: $(m_1 + j) \times (n_1 + 1)$
  - Bd Subproblem: $m_2 \times n_2$
- For each DW iteration
  - Increases in 1 the number of variables of the master problem
  - Modifies the objective function of the subproblem
  - DW Relaxed Master: $(m_1 + 1) \times j$
  - DW Subproblem: $m_2 \times n_1$

DW Algorithm deduction (i)

- Original problem

$$\min_{x_1} c_1^T x_1$$
$$A_1 x_1 = b_1 \quad : \pi_2'^T$$
$$x_1 \in K$$

- Lagrangian

$$L(x_1, \pi_2') = c_1^T x_1 + \pi_2'^T (A_1 x_1 - b_1)$$

- The dual function (concave) will be un upper bound

$$\begin{aligned} \theta_2(\pi_2') &= \min_{x_1 \in K} L(x_1, \pi_2') = c_1^T x_1 + \pi_2'^T (A_1 x_1 - b_1) \\ &= -\pi_2'^T b_1 + \min_{x_1 \in K} (c_1^T + \pi_2'^T A_1) x_1 \end{aligned}$$

$$\pi_2 = -\pi_2'$$

or

$$\begin{aligned} \theta_2(\pi_2) = \quad & \min_{x_1} (c_1^T - \pi_2^T A_1) x_1 + \pi_2^T b_1 \\ & A_2 x_1 = b_2 \\ & x_1 \geq 0 \end{aligned}$$

SUBPROBLEM

DW Algorithm deduction (ii)

- The optimum of the dual function is reached in one of the vertices

$$\theta_2(\pi_2) = \pi_2^T b_1 + \min_{x_1^l}(c_1^T - \pi_2^T A_1)x_1^l \qquad l = 1,\ldots,\nu$$

- It can be expressed as

$$\begin{aligned}
&\theta_2(\pi_2) \le \pi_2^T b_1 + (c_1^T - \pi_2^T A_1)x_1^1 \\
&\theta_2(\pi_2) \le \pi_2^T b_1 + (c_1^T - \pi_2^T A_1)x_1^2 \\
&\vdots \\
&\theta_2(\pi_2) \le \pi_2^T b_1 + (c_1^T - \pi_2^T A_1)x_1^\nu
\end{aligned}$$

$$\begin{aligned}
&\max_{\theta_2,\pi_2} \theta_2 \\
&\theta_2 + (A_1 x_1^1 - b_1)^T \pi_2 \le c_1^T x_1^1 \quad : \lambda_1 \\
&\theta_2 + (A_1 x_1^2 - b_1)^T \pi_2 \le c_1^T x_1^2 \quad : \lambda_2 \\
&\vdots \\
&\theta_2 + (A_1 x_1^\nu - b_1)^T \pi_2 \le c_1^T x_1^\nu \quad : \lambda_\nu
\end{aligned}$$

- Taking the dual we obtain the complete master problem

$$\begin{aligned}
&\min_{\lambda_l} \sum_{l=1}^{\nu} (c_1^T x_1^l)\lambda_l \\
&\sum_{l=1}^{\nu} \lambda_l = 1 \qquad : \mu = \theta_2 \\
&\sum_{l=1}^{\nu} (A_1 x_1^l - b_1)\lambda_l = 0 \quad : \pi_2 \\
&\lambda_l \ge 0 \quad l = 1,\ldots,\nu
\end{aligned}$$

COMPLETE MASTER PROBLEM

8

Lagrangian relaxation

LR Algorithm deduction (i)

- Joseph-Louis Lagrange (https://en.wikipedia.org/wiki/Joseph-Louis_Lagrange)
- We take the optimum of the dual function

$$
\begin{aligned}
&\max_{\theta_2,\pi_2} \theta_2 \\
&\theta_2 + (A_1 x_1^1 - b_1)^T \pi_2 \le c_1^T x_1^1 \quad : \lambda_1 \\
&\theta_2 + (A_1 x_1^2 - b_1)^T \pi_2 \le c_1^T x_1^2 \quad : \lambda_2 \\
&\vdots \\
&\theta_2 + (A_1 x_1^\nu - b_1)^T \pi_2 \le c_1^T x_1^\nu \quad : \lambda_\nu
\end{aligned}
$$

or doing the variable change $\theta_2 = b_1^T \pi_2 + \mu$

$$
\begin{aligned}
&\max_{\pi_2,\mu} b_1^T \pi_2 + \mu \\
&(A_1 x_1^1)^T \pi_2 + \mu \le c_1^T x_1^1 \quad : \lambda_1 \\
&(A_1 x_1^2)^T \pi_2 + \mu \le c_1^T x_1^2 \quad : \lambda_2 \\
&\vdots \\
&(A_1 x_1^\nu)^T \pi_2 + \mu \le c_1^T x_1^\nu \quad : \lambda_\nu
\end{aligned}
$$

LR Algorithm deduction (ii)

- LR Relaxed Master problem

$$\max_{\theta_2,\pi_2} \theta_2$$
$$\theta_2 + (A_1 x_1^l - b_1)^T \pi_2 \le c_1^T x_1^l \quad : \lambda_l \quad l = 1,\dots,j$$

$$\max_{\pi_2,\mu} b_1^T \pi_2 + \mu$$
$$(A_1 x_1^l)^T \pi_2 + \mu \le c_1^T x_1^l \quad : \lambda_l \quad l = 1,\dots,j$$

- Constraints are called dual cuts or Lagrangian optimality cuts.
- The formulation is the method of Kelly's cutting planes.
- To avoid unbounded subproblems, bounding cuts are introduced

$$\max_{\theta_2,\pi_2} \theta_2$$
$$\delta_1^l \theta_2 + (A_1 x_1^l - \delta_1^l b_1)^T \pi_2 \le c_1^T x_1^l \quad : \lambda_l \quad l = 1,\dots,j$$

$$\max_{\pi_2,\mu} b_1^T \pi_2 + \mu$$
$$(A_1 x_1^l)^T \pi_2 + \delta_1^l \mu \le c_1^T x_1^l \quad : \lambda_l \quad l = 1,\dots,j$$

- $\delta_1^l = 1$ dual cuts, $\delta_1^l = 0$ bounding cuts

LR Algorithm deduction (iii)

- LR Subproblem

$$\begin{aligned}\theta_2 = \quad & \min_{x_1} (c_1^T - \pi_2^{jT} A_1) x_1 + \pi_2^{jT} b_1 \\ & A_2 x_1 = b_2 \\ & x_1 \geq 0\end{aligned}$$

$$\begin{aligned}\theta_2 = \quad & \min_{x_1} \left(c_1^T - \pi_2^{jT} A_1\right) x_1 - \mu^j \\ & A_2 x_1 = b_2 \\ & x_1 \geq 0\end{aligned}$$

LR Relaxed Master and Subproblem

- LR Relaxed Master:
  - Dual of DW Relaxed Master
  - Generates economic signals (primal variables)
  - Adds a new constraint at each iteration (dual cut)
  - Objective function decreasing monotonically
  - Outer linearization of the subproblem
- LR Subproblem:
  - Objective function is reduced costs or total costs minus recourse price
  - Plans decentralizedly internalizing the economic signals
  - Second-stage constraints
  - Each iteration changes the objective function
  - Can be nonconvex

Comparison between DW and LR

- Original linear problem PL-2: $(m_1 + m_2) \times n_1$
- For each LR iteration
  - Increases in 1 the number of constraints of the master problem
  - Dual simplex method
  - LR Relaxed Master: $j \times (m_1 + 1)$
  - Modifies the objective function of the LR subproblem
  - Primal simplex method
  - LR Subproblem: $m_2 \times n_1$
- Comparison with DW
  - DW: dual variables are dual for the master (degeneracy)
  - LR: dual variables are primal for the master

LR Optimal solution

- LR optimal solution is not necessarily feasible in the complicating constraints $A_1 x_1 = b_1$
  - Introduce a penalty in the o.f. of the subproblem (augmented Lagrangian). Besides, differentiability is achieved
  - Postprocessing the optimal solution

Subgradient method

- Use NLP techniques to obtain the Lagrange multipliers

$$\pi_2^{j+1} = \pi_2^j + \alpha_j p_j$$

- Search direction: gradient of the Lagrangian w.r.t. π_2^j

$$p_j = A_1 x_1^j - b_1$$

- Updating the step length α_j

$$\alpha_j = \frac{\beta_j \left[\theta_2(\pi_2^j) - c_1^T x_1^*\right]}{\sum_m \left(\sum_n a_{mn} x_n^j - b_m\right)^2}$$

Penalty for distance to the best optimal solution

Penalty for constraint violation

- β_j decreasing sequence 1, ½, ¼, 1/8

Subgradient method

- Search direction: gradient of the Lagrangian w.r.t. x_1^j

$$p_j = c_1^T - \pi_2^{jT} A_1$$

- Updating mechanism

$$\pi_2^{j+1} = \pi_2^j + \alpha_j p_j$$

- Step length

$$\alpha_j = \frac{\beta_j \left[c_1^T x_1 - \pi_2^{jT} (A_1 x_1 - b_1) - c_1^T x_1^* \right]}{\sum_j \left(p_j \right)^2}$$

- β_j decreasing sequence 1, ½, ¼, 1/8

LR Algorithm

1. Initialization: $j=0$ $\underline{z}=-\infty$ $\overline{z}=\infty$ $\varepsilon=10^{-4}$
2. Solving the LR Relaxed Master problem

$$\max_{\theta_2,\pi_2} \theta_2$$
$$\delta_1^l\theta_2+(A_1x_1^l-\delta_1^l b_1)^T\pi_2 \le c_1^T x_1^l \quad :\lambda_l \quad l=1,\dots,j$$

$$\max_{\pi_2,\mu} b_1^T\pi_2+\mu$$
$$(A_1x_1^l)^T\pi_2+\delta_1^l\mu \le c_1^T x_1^l \quad :\lambda_l \quad l=1,\dots,j$$

Obtain the solution π_2

3. Bounding LR Subproblem solution

$$\theta_2^*(\pi_2)=\min_{x_1}(c_1^T-\pi_2^T A_1)x_1$$
$$A_2x_1\le 0$$
$$0\le x_1\le 1$$

If $\theta_2^*(\pi_2)\ge 0$ go to step 4.

If not, build a bounding cut. $(A_1x_1^j)^T\pi_2\le c_1^T x_1^j$

4. Solving the LR Subproblem

$$\theta_2=\min_{x_1}(c_1^T-\pi_2^{jT}A_1)x_1+\pi_2^{jT}b_1$$
$$A_2x_1=b_2$$
$$x_1\ge 0$$

$$\theta_2=\min_{x_1}\left(c_1^T-\pi_2^{jT}A_1\right)x_1-\mu^j$$
$$A_2x_1=b_2$$
$$x_1\ge 0$$

Obtain x_1^j and build a dual cut. $\theta_2+(A_1x_1^j-b_1)^T\pi_2\le c_1^T x_1^j$

5. If stopping rule $d(\pi_2^j-\pi_2^{j-1})<\varepsilon$ is satisfied stops. If not go to step 2.

Fixed-charge transportation problem

- Complete problem

$$\min_{x_{ij}, y_{ij}} \sum_{ij} \left(c_{ij} x_{ij} + f_{ij} y_{ij} \right)$$

$$\sum_j x_{ij} \le a_i \qquad \forall i$$

$$\sum_i x_{ij} \ge b_j \qquad \forall j$$

$$x_{ij} \le M_{ij} y_{ij} \qquad \forall ij$$

Complicating constraints

$$x_{ij} \ge 0, y_{ij} \in \{0,1\}$$

Fixed-charge transportation problem

$$\min_{x_{ij},y_{ij}} \sum_{ij}\left(c_{ij}x_{ij} + f_{ij}y_{ij}\right) + \lambda_{ij}^k\left(x_{ij} - M_{ij}y_{ij}\right)$$
$$\sum_j x_{ij} \le a_i$$
$$\sum_i x_{ij} \ge b_j$$
$$x_{ij} \ge 0, y_{ij} \in \{0,1\}$$

- LR Subproblem

$$\min_{x_{ij},y_{ij}} \sum_{ij}\left(c_{ij} + \lambda_{ij}^k\right)x_{ij} + \left(f_{ij} - \lambda_{ij}^k M_{ij}\right)y_{ij}$$
$$\sum_j x_{ij} \le a_i$$
$$\sum_i x_{ij} \ge b_j$$
$$x_{ij} \ge 0, y_{ij} \in \{0,1\}$$

Separable

$$\min_{x_{ij}} \sum_{ij}\left(c_{ij} + \lambda_{ij}^k\right)x_{ij}$$
$$\sum_j x_{ij} \le a_i$$
$$\sum_i x_{ij} \ge b_j$$
$$x_{ij} \ge 0$$

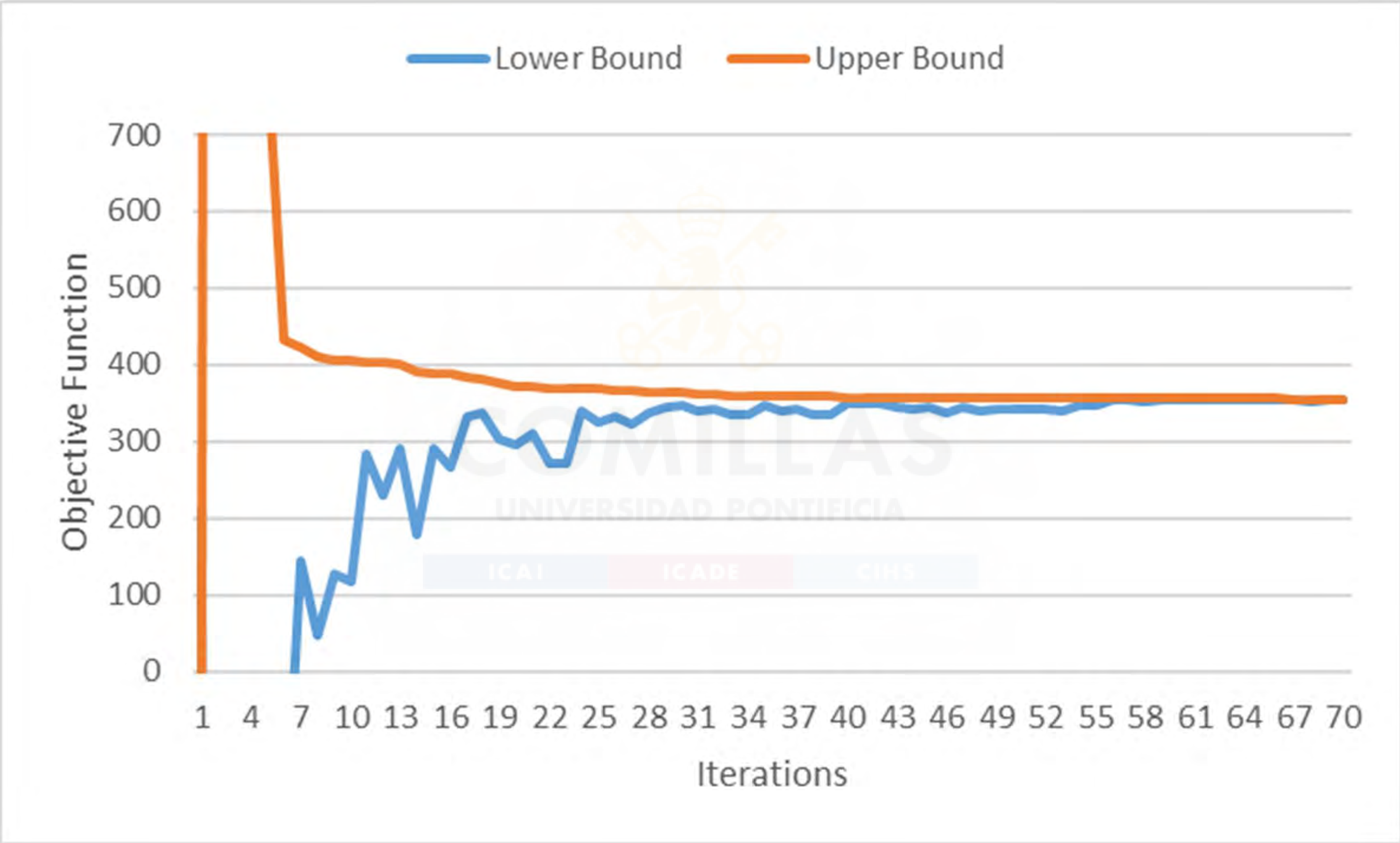
$$\min_{y_{ij}} \sum_{ij}\left(f_{ij} - \lambda_{ij}^k M_{ij}\right)y_{ij}$$
$$y_{ij} \in \{0,1\}$$

Fixed-charge transportation problem

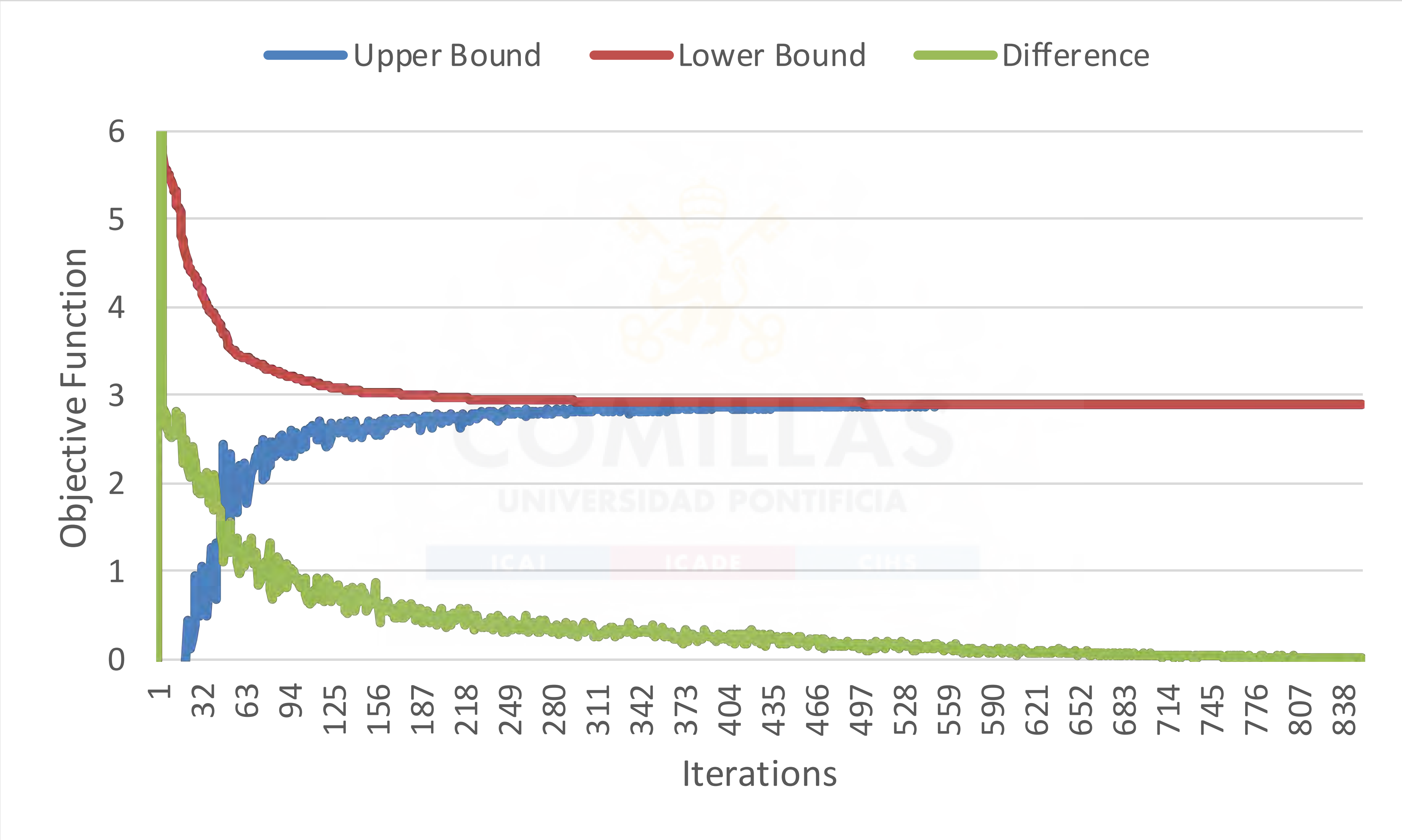
- LR Master Problem

$$
\begin{aligned}
&\max_{\theta_2, \lambda_{ij}} \theta_2 \\
&\theta_2 \le \sum_{ij} \left(c_{ij} x_{ij}^k + f_{ij} y_{ij}^k \right) + \lambda_{ij} \left(x_{ij}^k - M_{ij} y_{ij}^k \right) \quad \forall k \\
&\lambda_{ij} \ge 0
\end{aligned}
$$

Fixed-charge transportation problem. LR Convergence



Unit Commitment. LR convergence



9

Scenario tree

Objectives

- To understand
  - How uncertainty is represented
  - What is a scenario tree
  - How it is generated

Stochasticity or uncertainty

- Origin
  - Future information (e.g., prices or future demand)
  - Lack of reliable data
  - Measurement errors
- In electric energy systems planning
  - Demand (yearly, seasonal or daily variation, load growth)
  - Hydro inflows
  - Availability of generation or network elements
  - Electricity or fuel prices
- Uncertainty relevance for each time scale

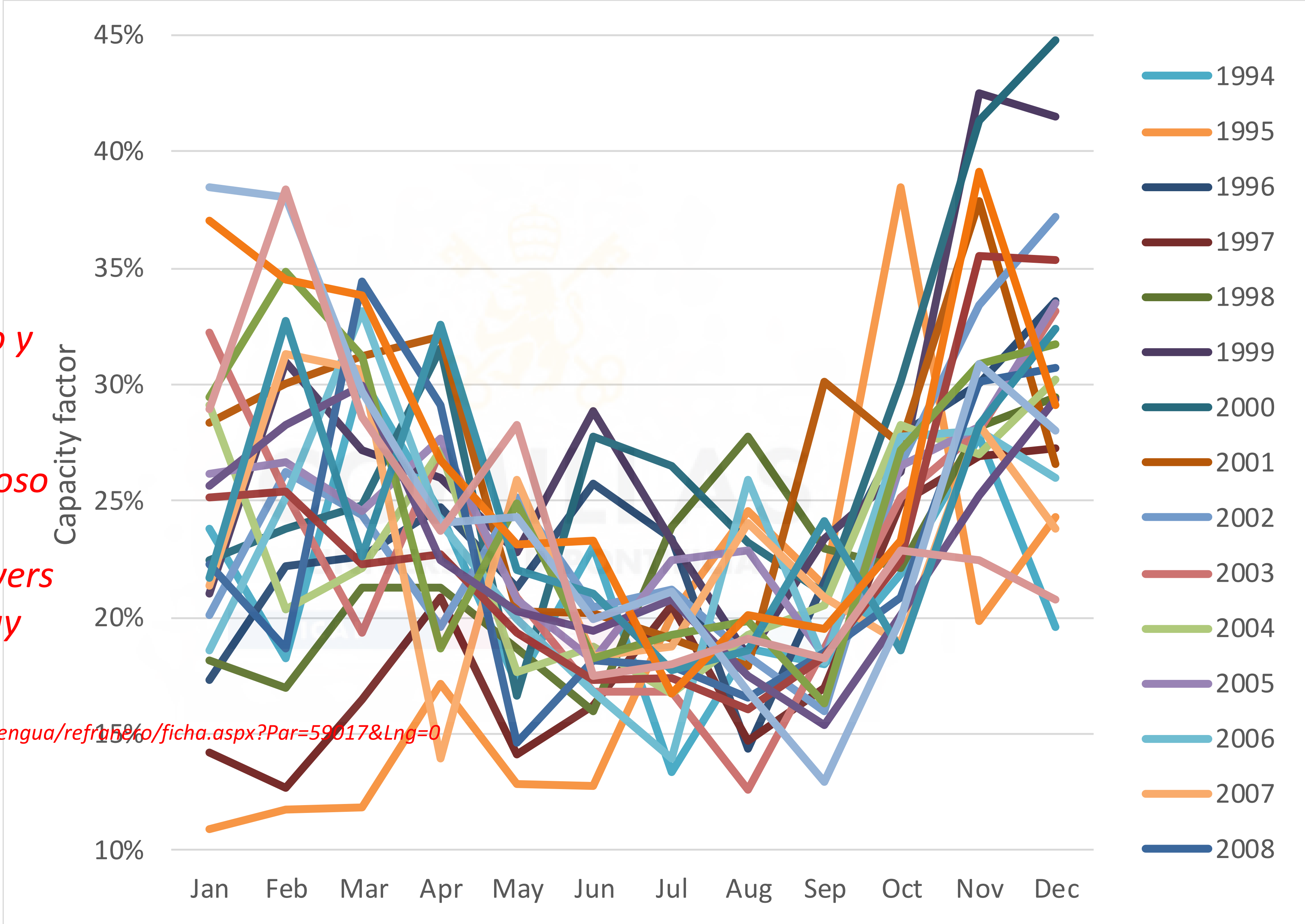
Long term load forecasting methods (i)

- Final use models
  - Explain the direct use of electricity for the different users
  - Require lot of data and are dependent on their quality
- Econometric models
  - Use economic data to explain the demand variation

Short term load forecasting methods (ii)

- Regression models
  - Determine the relation of the demand with factors as humidity, temperature, day of the week
- Time series analysis
  - Detect the intrinsic structure of the demand: trend, seasonal and daily variation
- Artificial neural networks
  - Perform an nonlinear adjust of the demand as a function of previous factors
- Fuzzy logic
  - Introduce qualitative aspects by means of fuzzy numbers

Monthly WP capacity factor

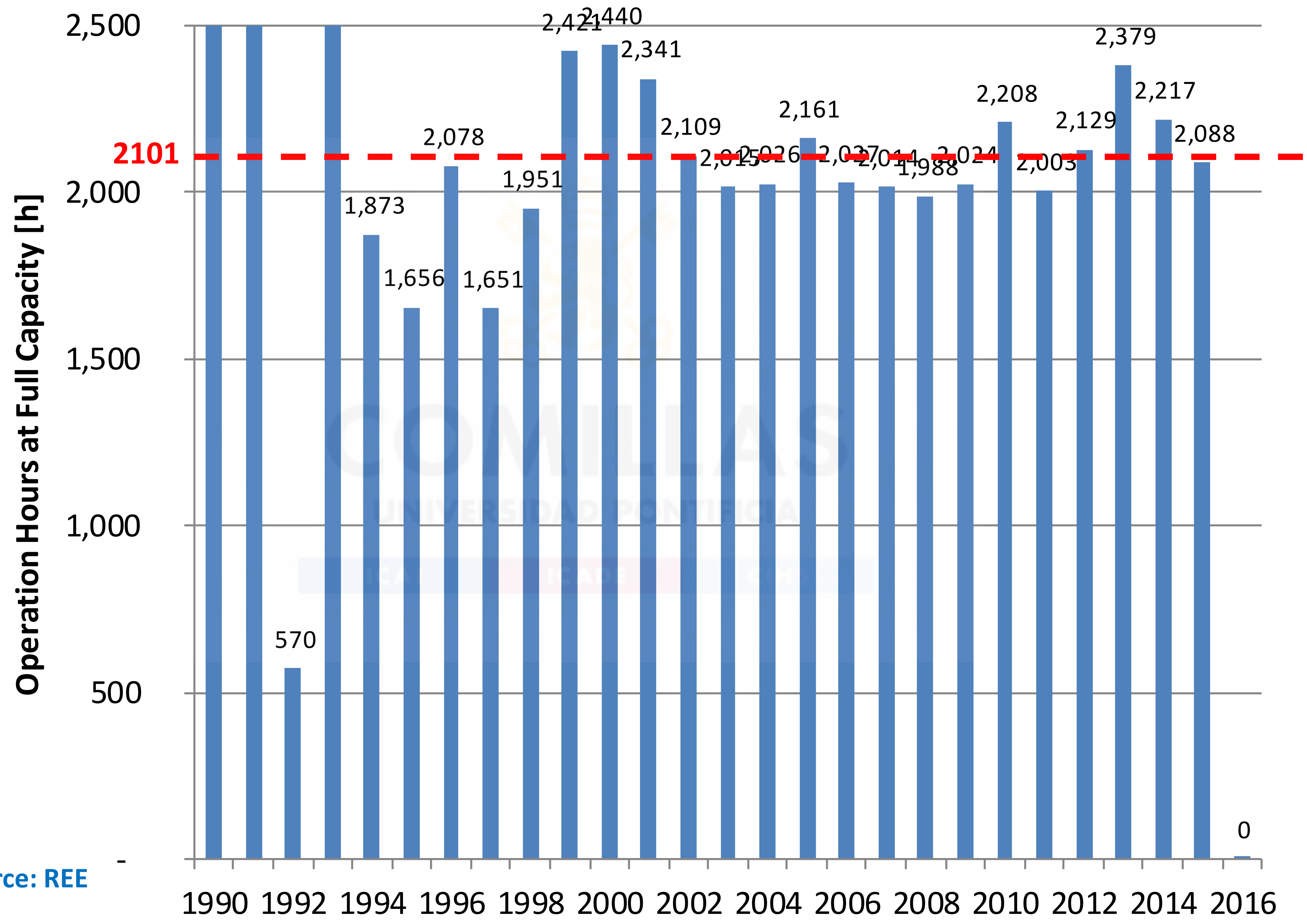


A saying:

Marzo ventoso y abril lluvioso sacan a mayo florido y hermoso (March winds and April showers bring forth May flowers)

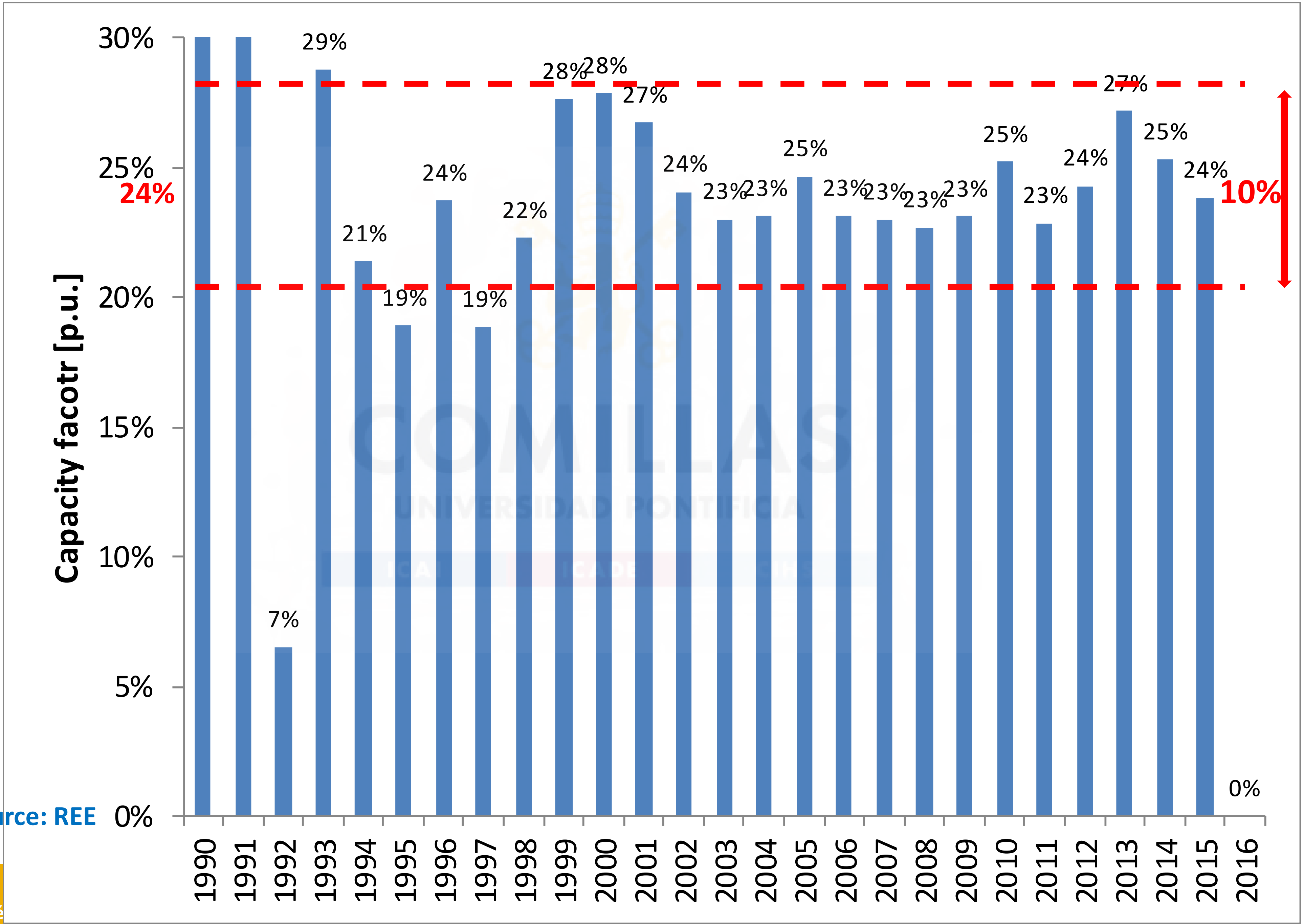
http://cvc.cervantes.es/lengua/refranero/ficha.aspx?Par=59017&Lng=0

WP Operation hours at full capacity



Source: REE

Yearly WP Capacity factor



Source: REE

Stochastic hydro inflows

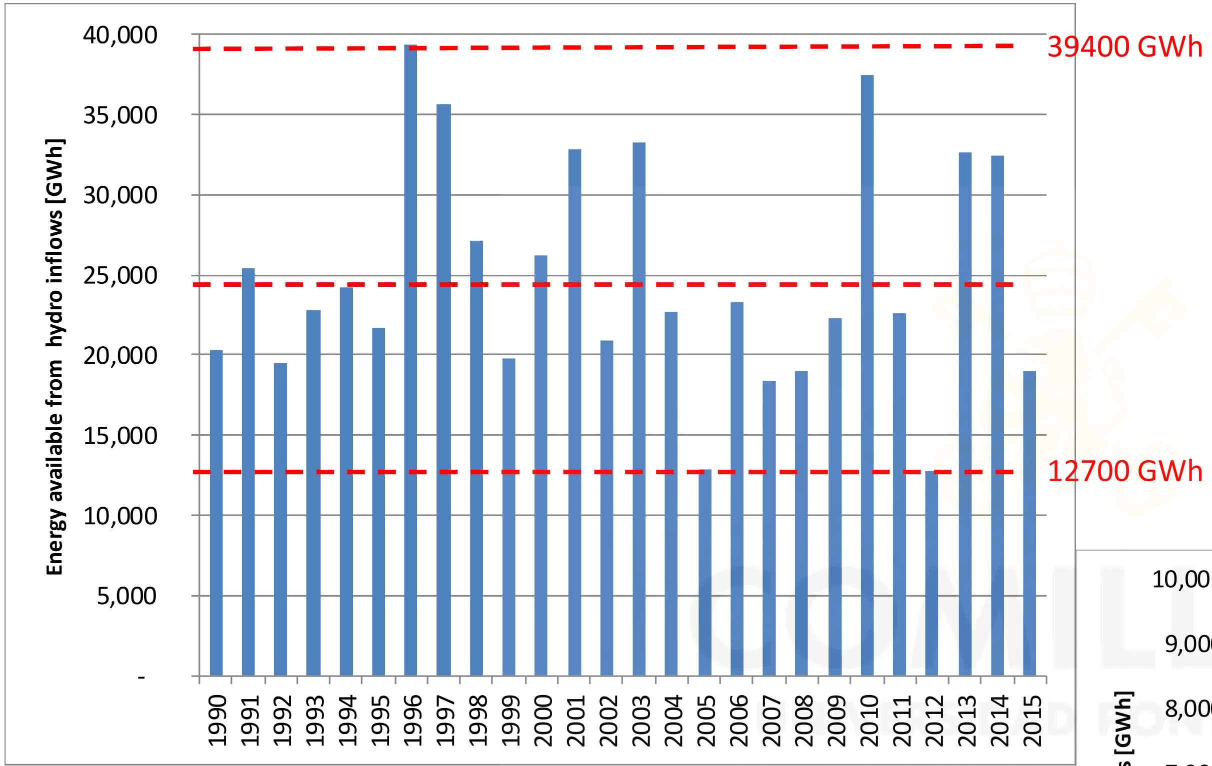
- Natural hydro inflows (clearly the most important factor in the Spanish electric system)

Year	Hydro energy TWh	Index	% of being exceeded
1990	20.3	0.57	98%
1991	25.4	0.84	76%
1992	19.5	0.64	95%
1993	22.8	0.75	85%
1994	24.2	0.80	29%
1995	21.7	0.72	88%
1996	39.4	1.30	17%
1997	35.6	1.19	27%
1998	27.1	0.90	64%
1999	19.8	0.67	92%
2000	26.2	0.90	64%
2001	32.9	1.13	32%
2002	20.9	0.72	87%
2003	33.2	1.15	30%
2004	22.7	0.79	80%
2005	12.9	0.45	100%
2006	23.3	0.82	74%
2007	18.4	0.65	92%
2008	18.9	0.67	90%
2009	22.3	0.79	76%
2010	37.4	1.34	13%
2011	22.6	0.81	74%
2012	12.7	0.46	100%
2013	32.6	1.18	25%
2014	32.4	1.17	26%
2015	18.9	0.69	88%

Source: REE

- Changes in reservoir volumes are significant because of:
  - stochasticity in hydro inflows
  - chronological pattern of inflows and
  - capacity of the reservoir with respect to the inflows

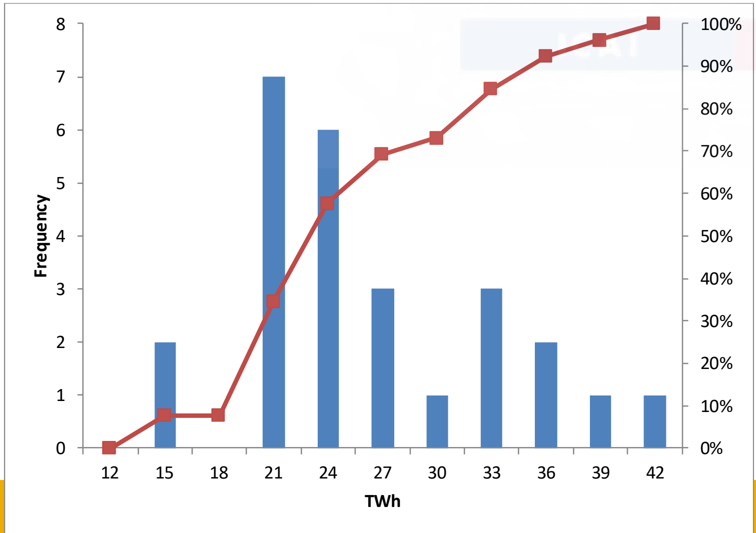
How uncertain are hydro inflows in Spain?

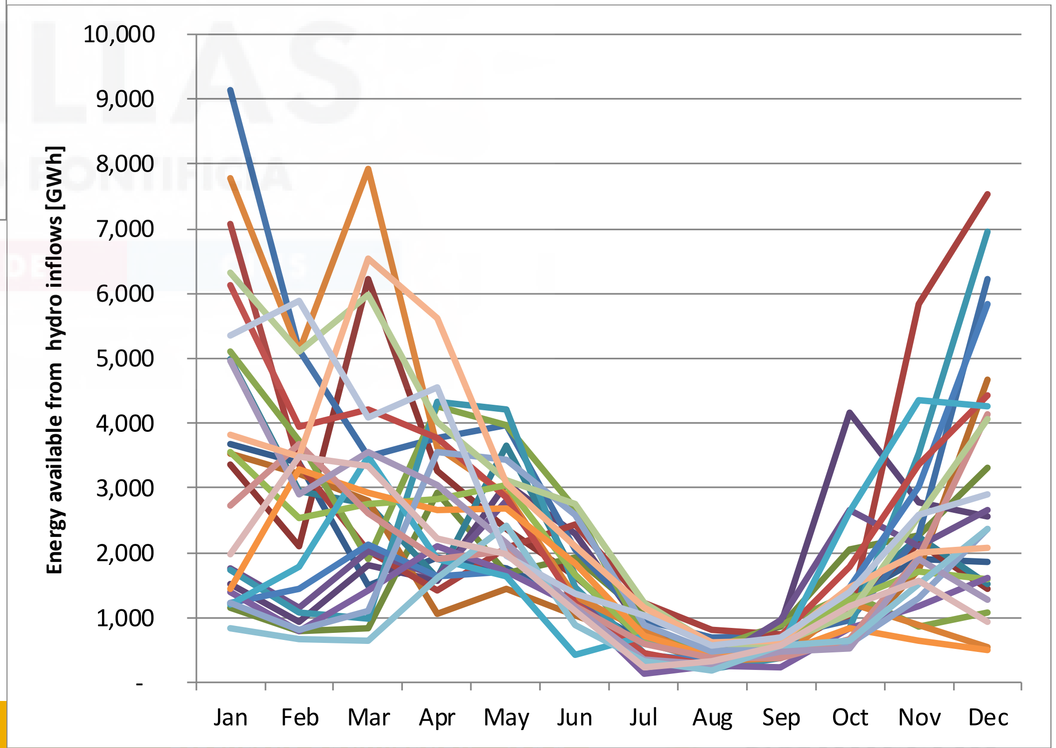


- Mean values

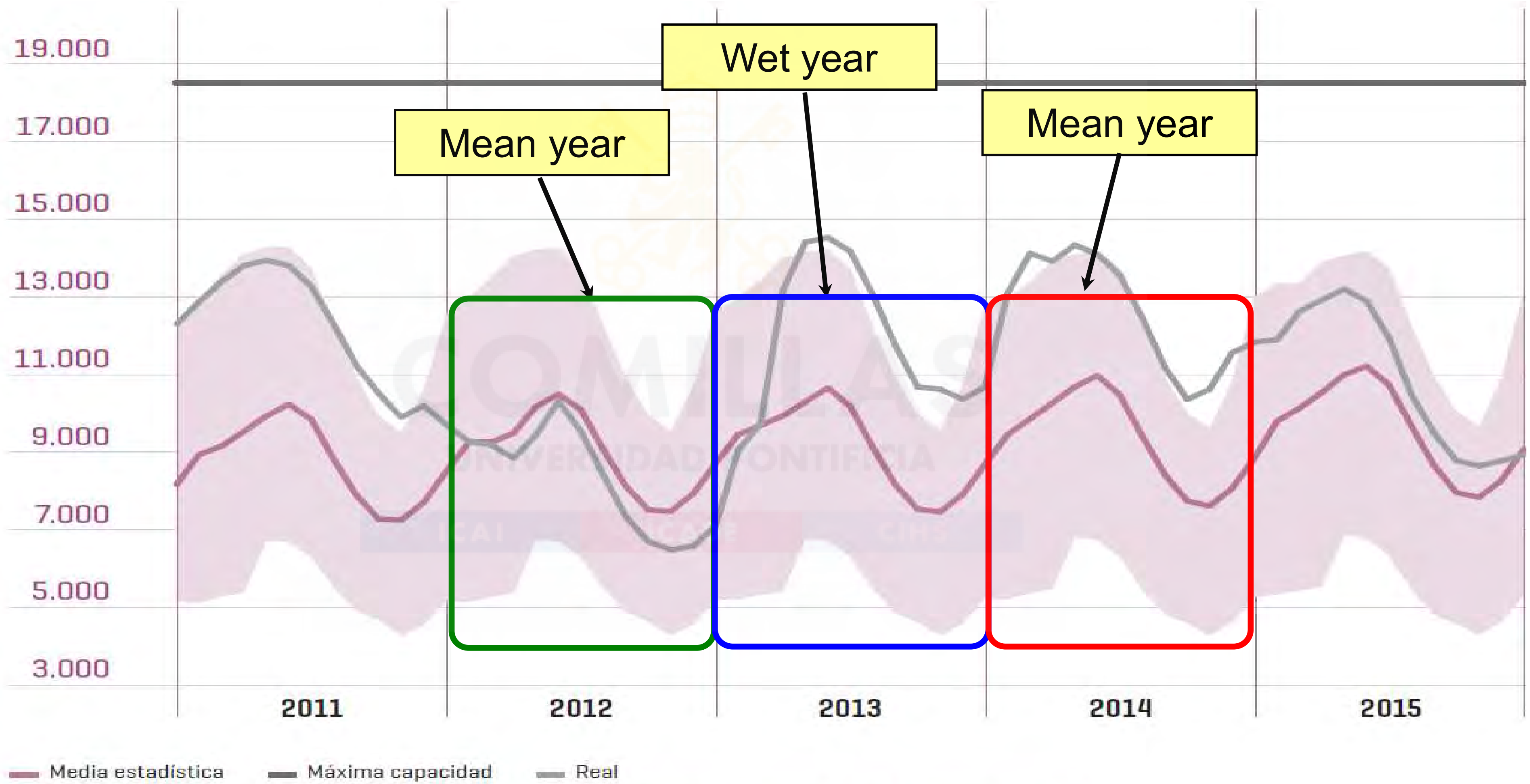
  24768 GWh

  2827 MW



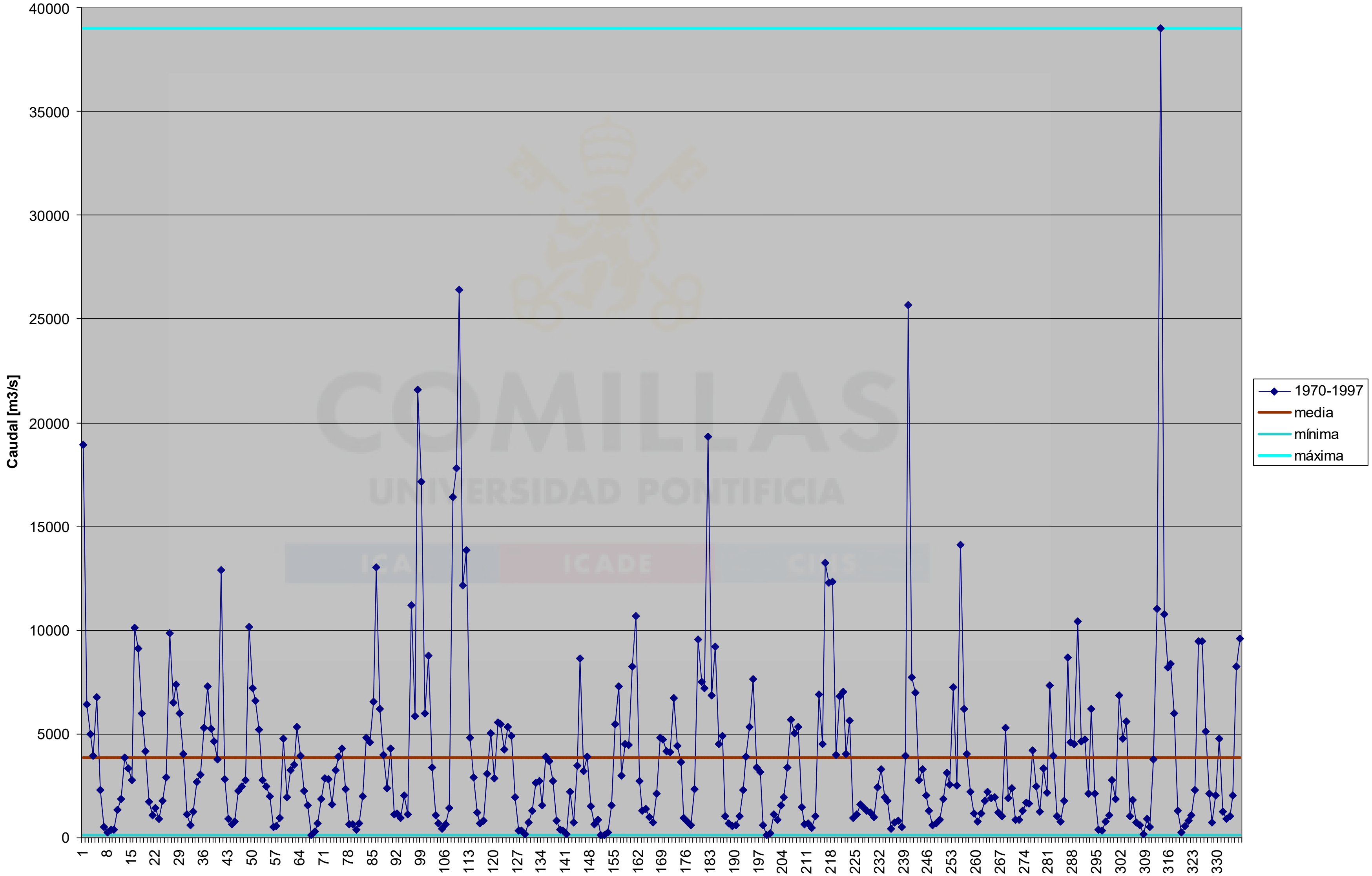


Output: stochastic reservoir levels

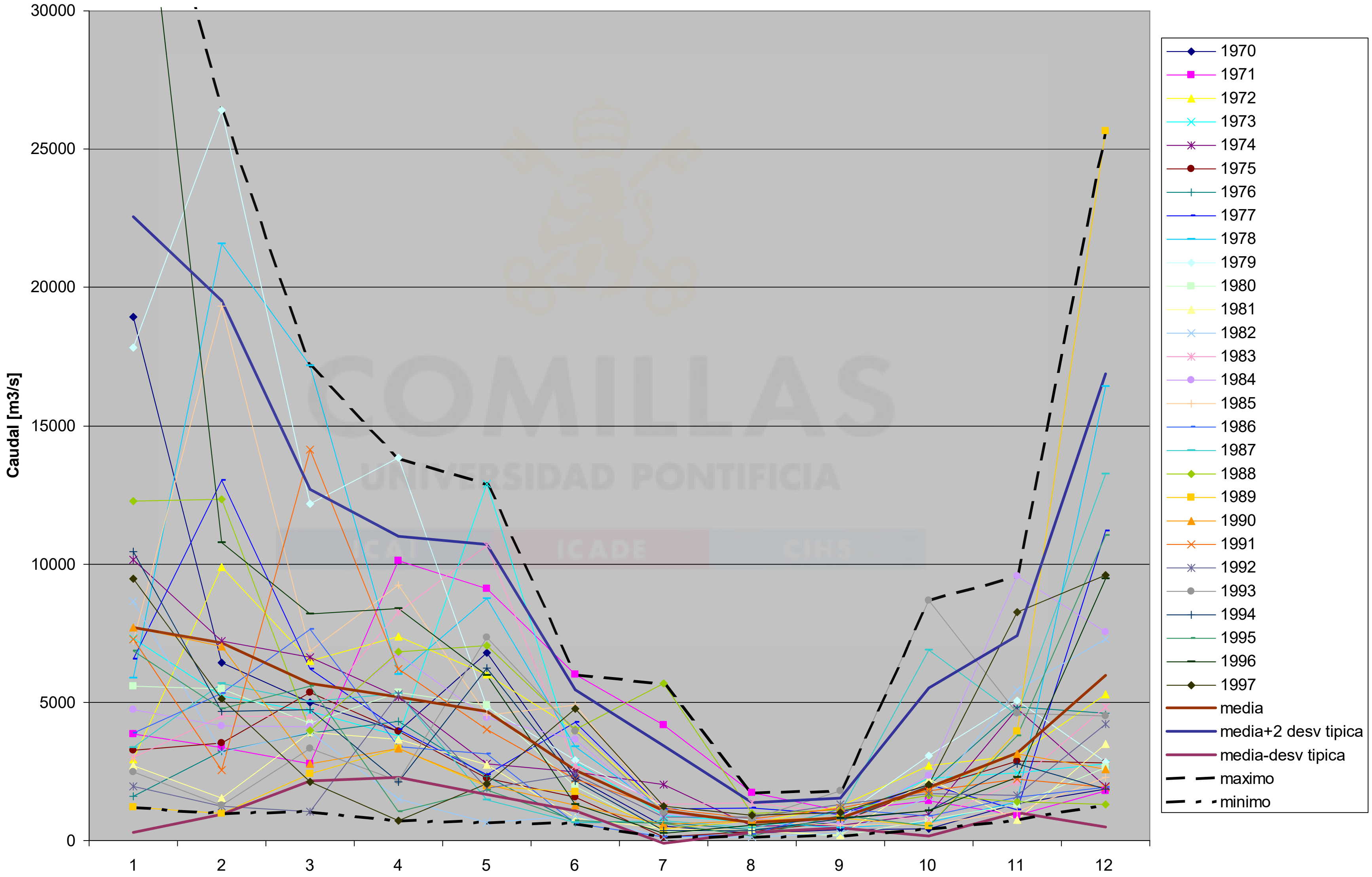


Máximo y mínimo estadístico: media de los valores máximos y mínimos de los últimos 20 años.

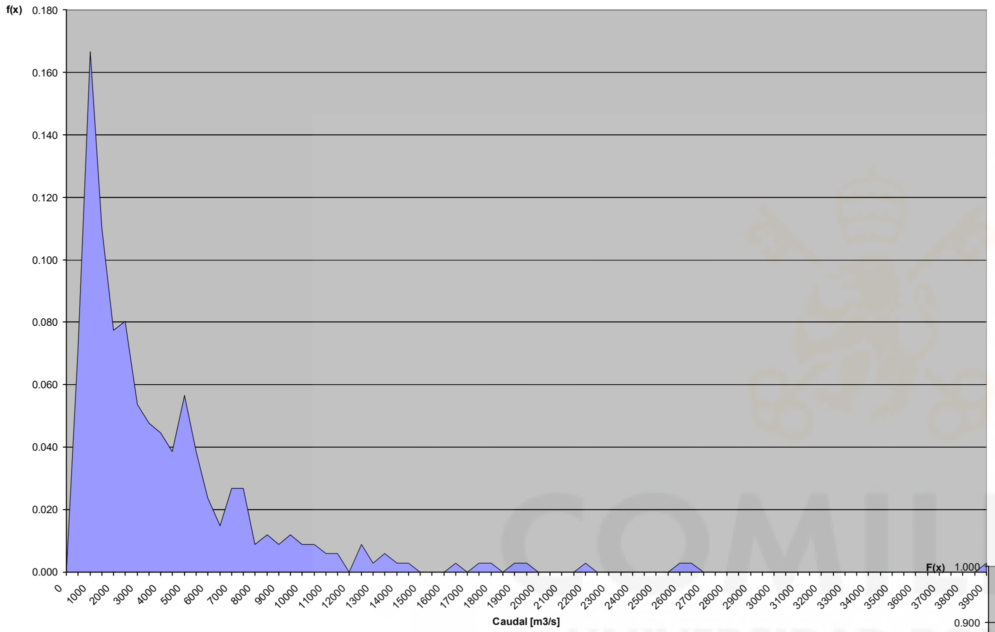
Natural hydro inflows: (monthly) historical series

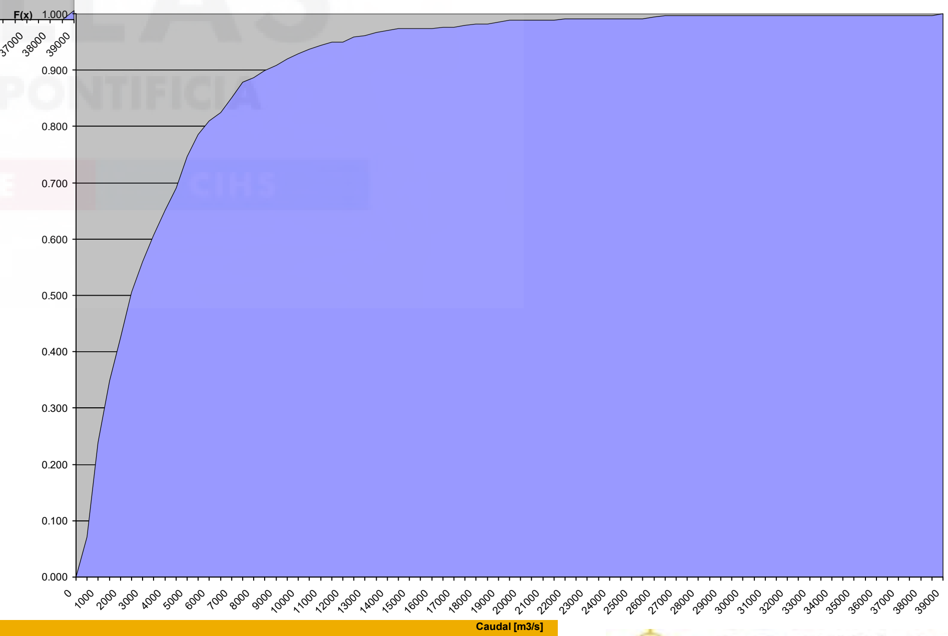


Natural hydro inflows: (monthly) historical series



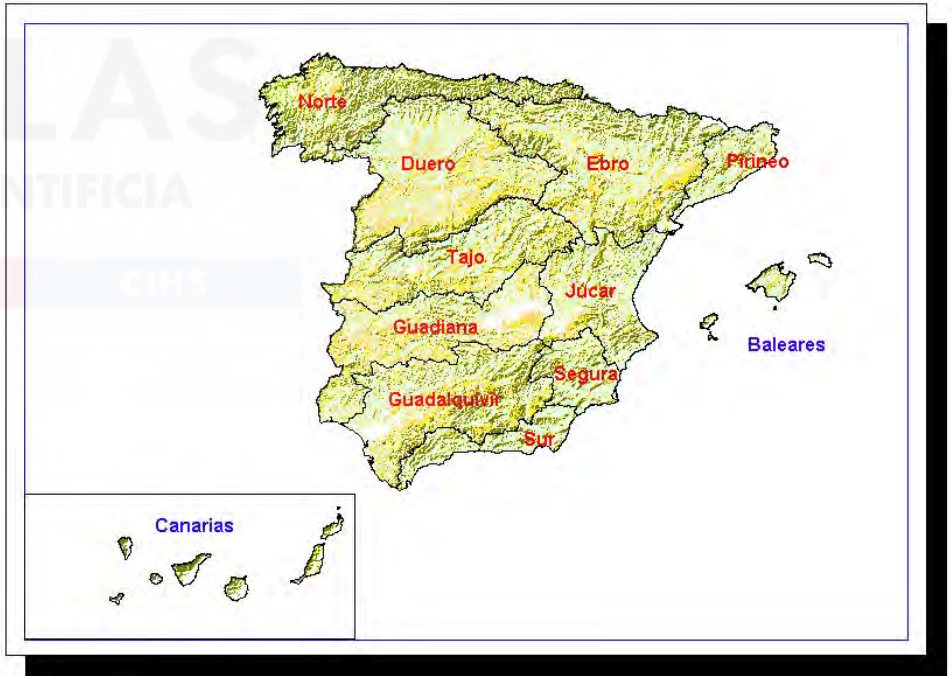
Probability density function (pdf) $f(x)$
Cumulative distribution function (cdf) $F(x)$



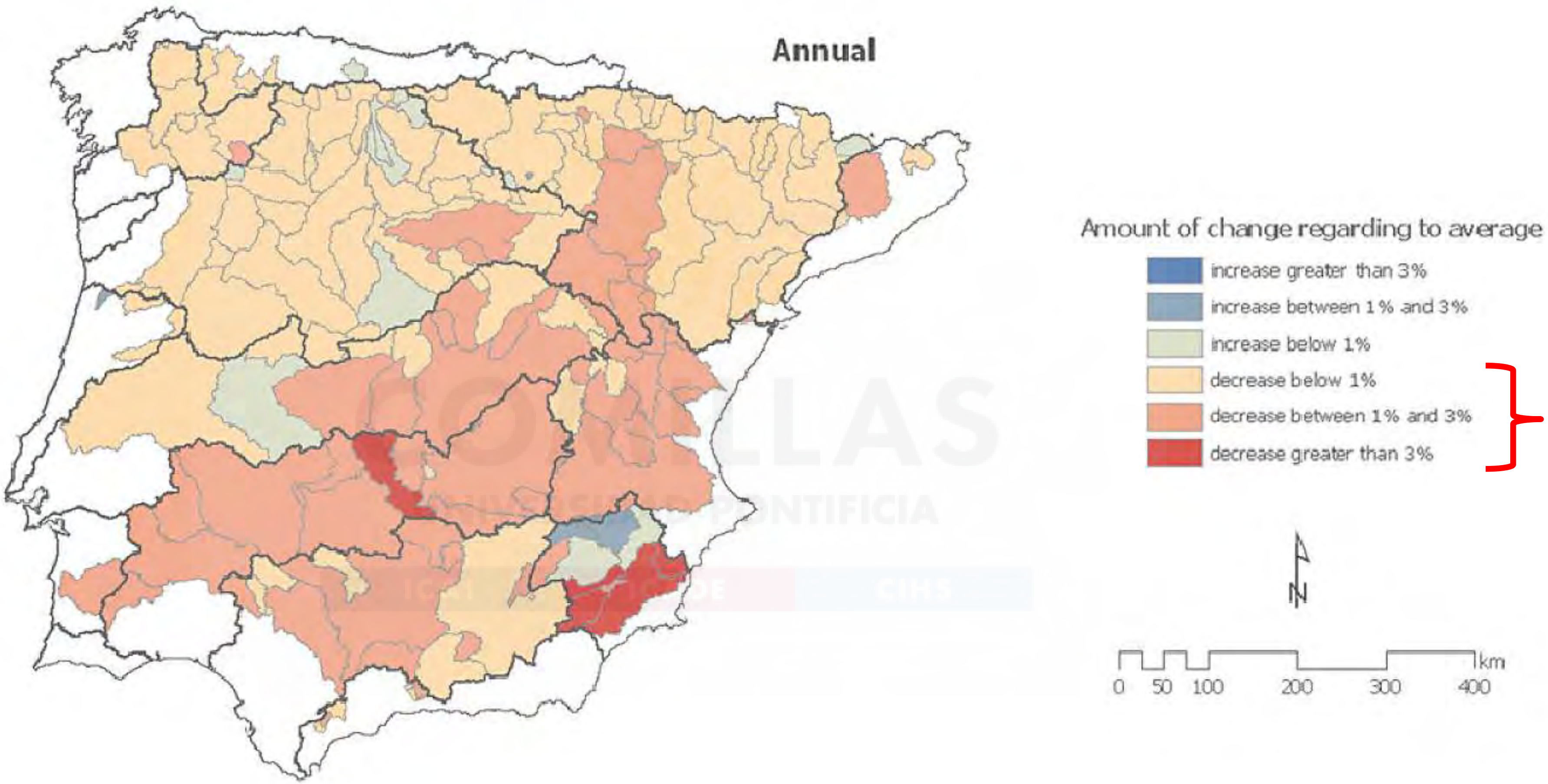


Water inflows

- Several measurement points in main different river basins
- Partial spatial correlation among them
- Temporal correlation in each one
- No establish method for obtaining a unique multivariate probability tree

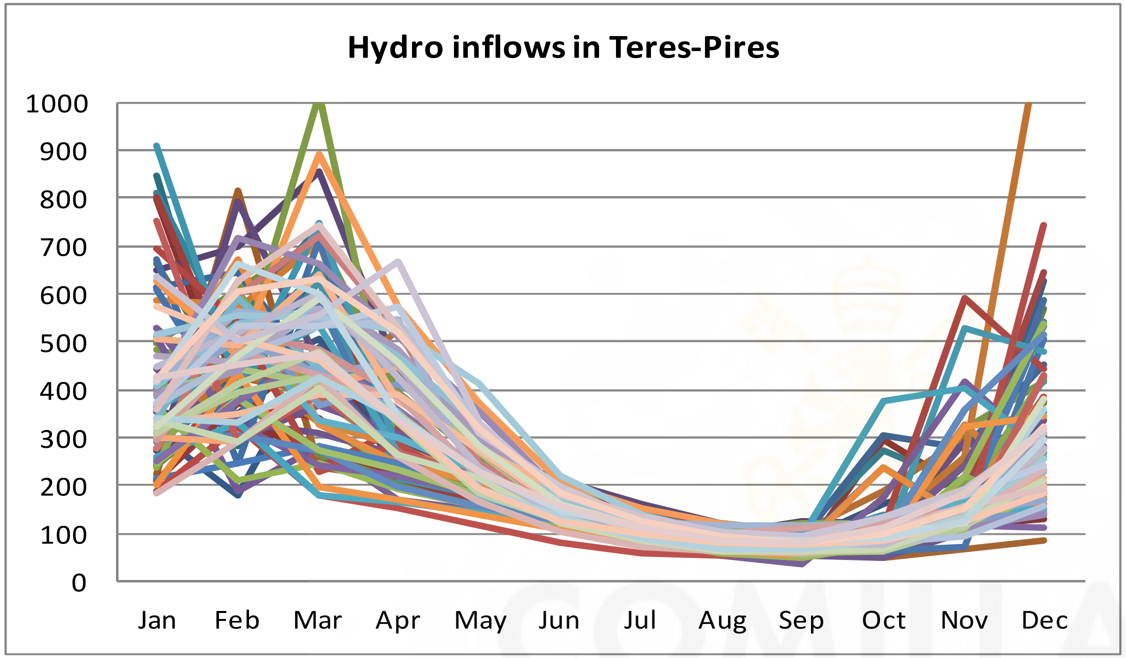


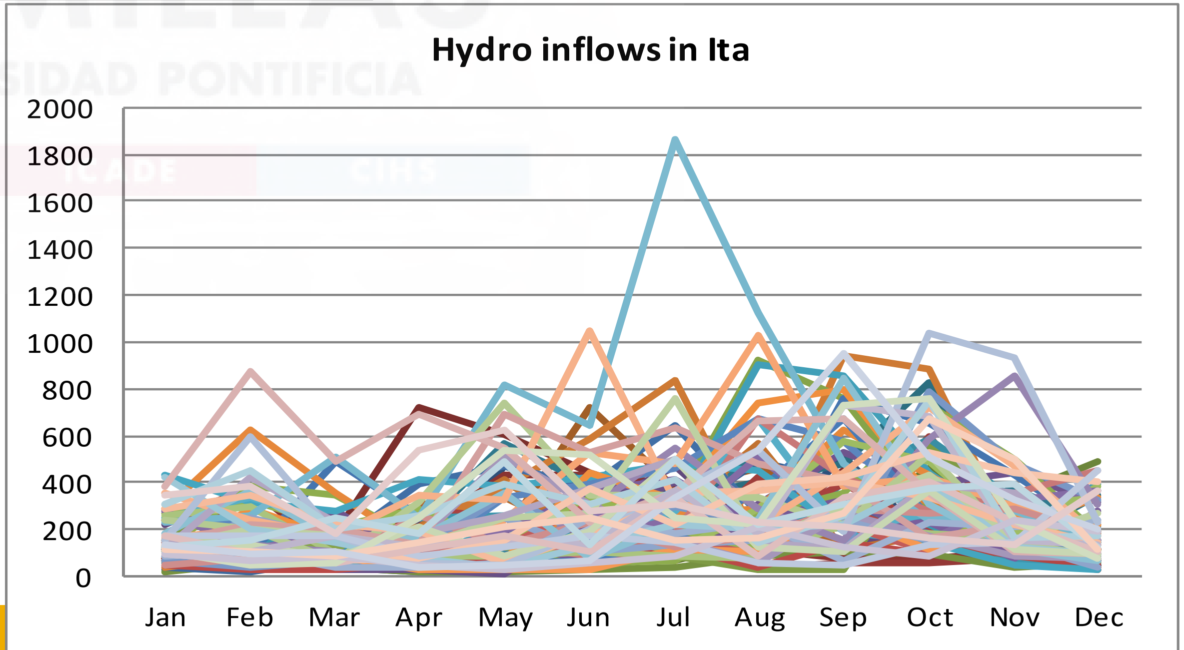
Inflows decrease in Iberian rivers in the last 60 years



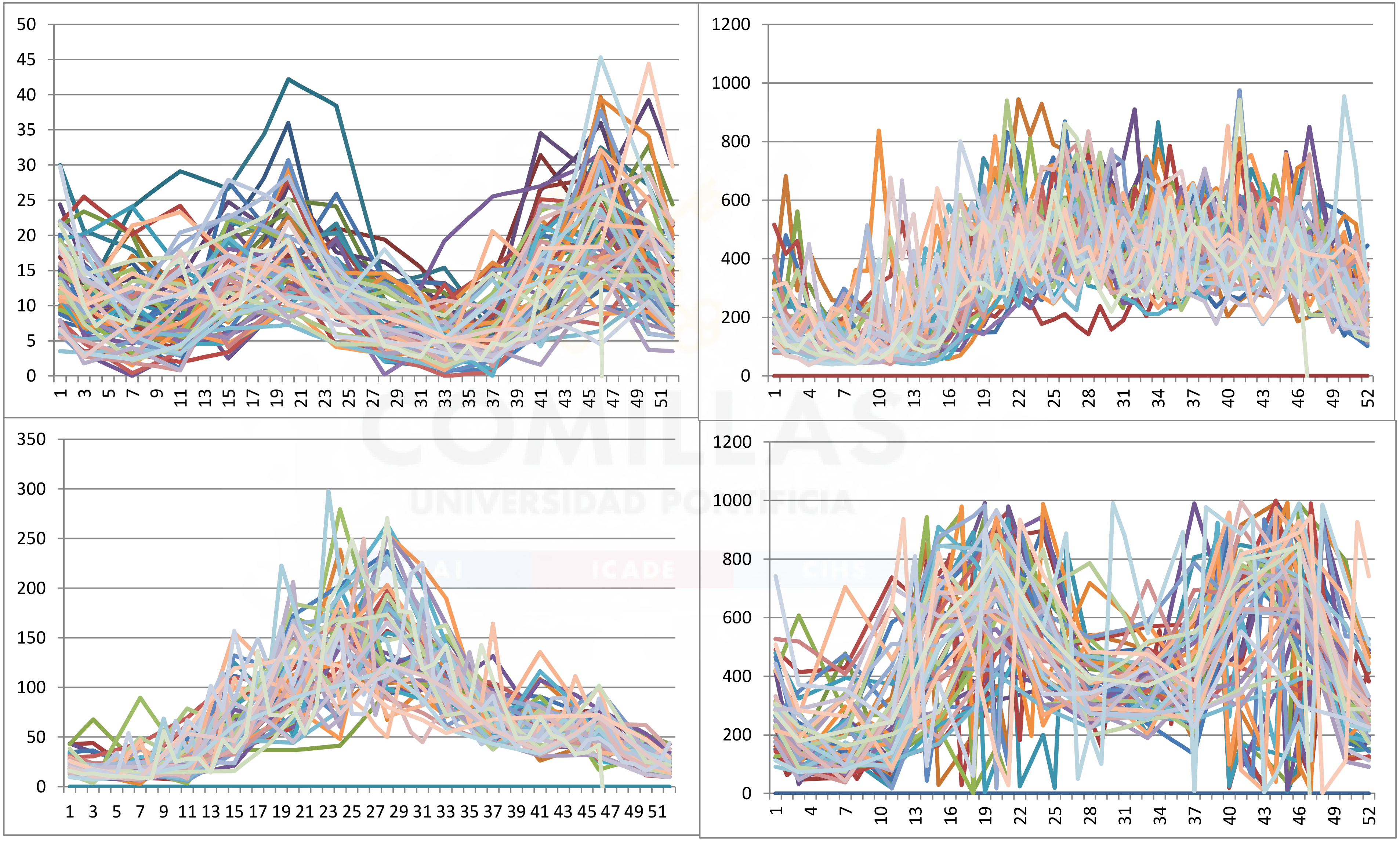
J. Lorenzo-Lacruz, S.M. Vicente-Serrano, J.I. López-Moreno, E. Morán-Tejeda, J. Zabalza. **Recent trends in Iberian streamflows (1945–2005)**. *Journal of Hydrology*.

How uncertain are hydro inflows in Brazil?





How uncertain are hydro inflows in Colombia?



Alternatives to model stochastic parameters

- Discrete probability function (i.e., scenario tree)
- Continuous or historic probability function that generates the tree by sampling (simulating) in each time period

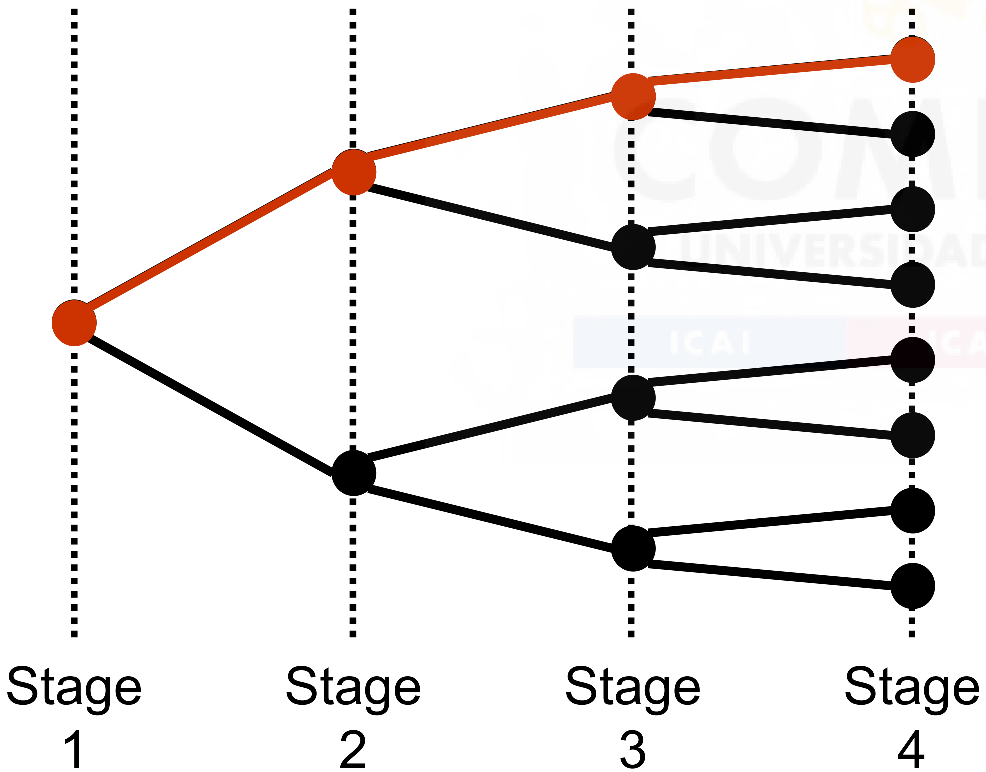
Scenario or probability tree

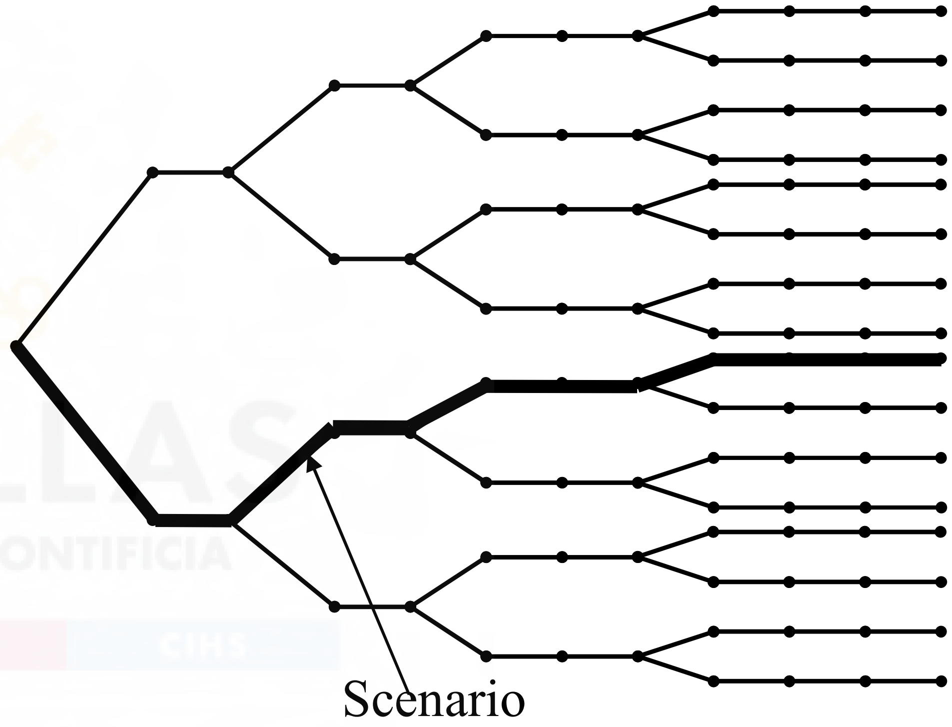
- Tree: represents how the stochasticity is revealed over time, i.e., the different states of the random parameters and simultaneously the non anticipative decisions over time.

  Correlation among parameters should be taken into account
- Scenario: any path going from the root to the leaves
- The scenarios that share the information until a certain period do the same into the tree (non anticipative decisions)

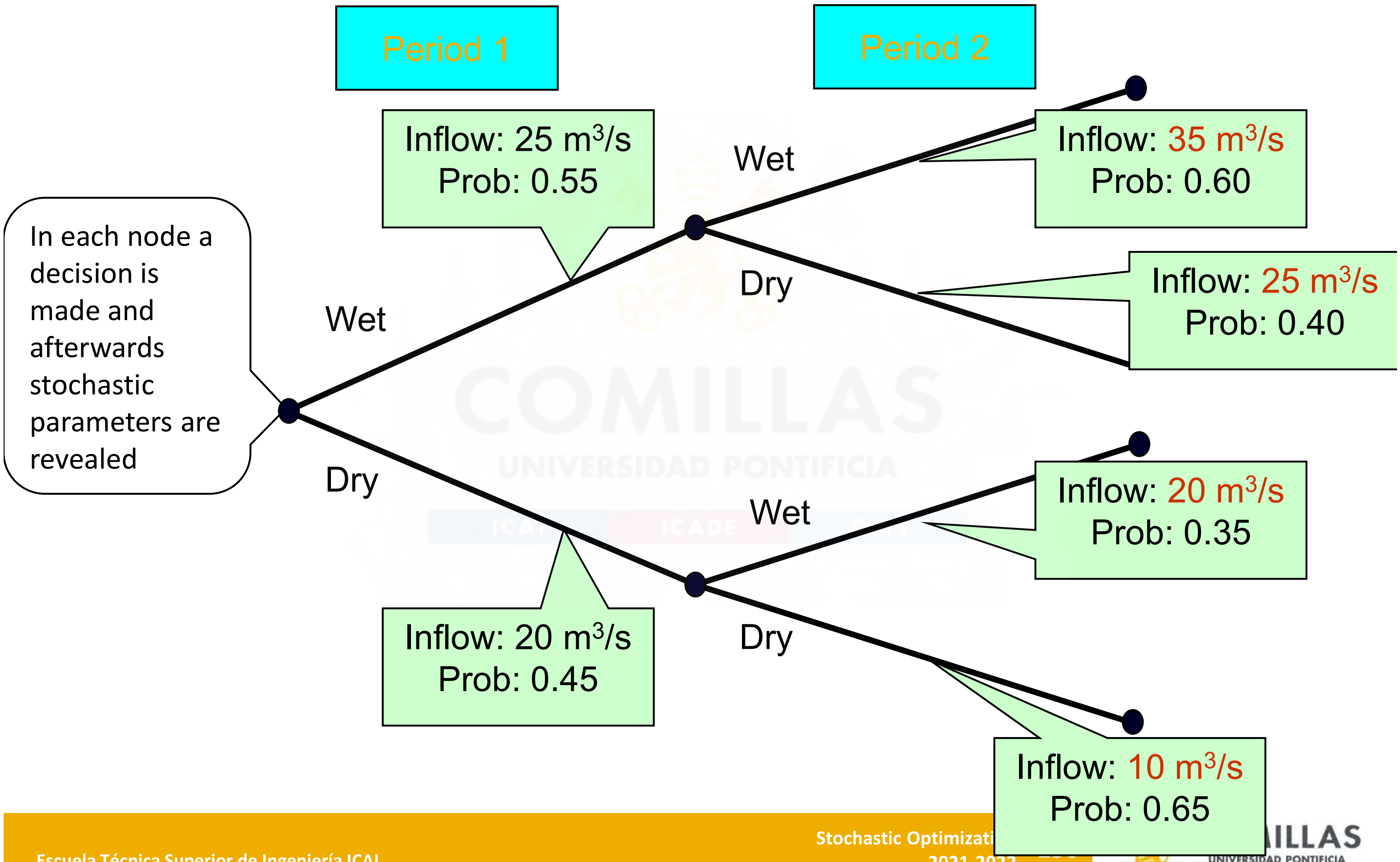
Scenario tree

- **Nodes**: where decisions are taken.
- **Scenarios**: realizations of the random process.

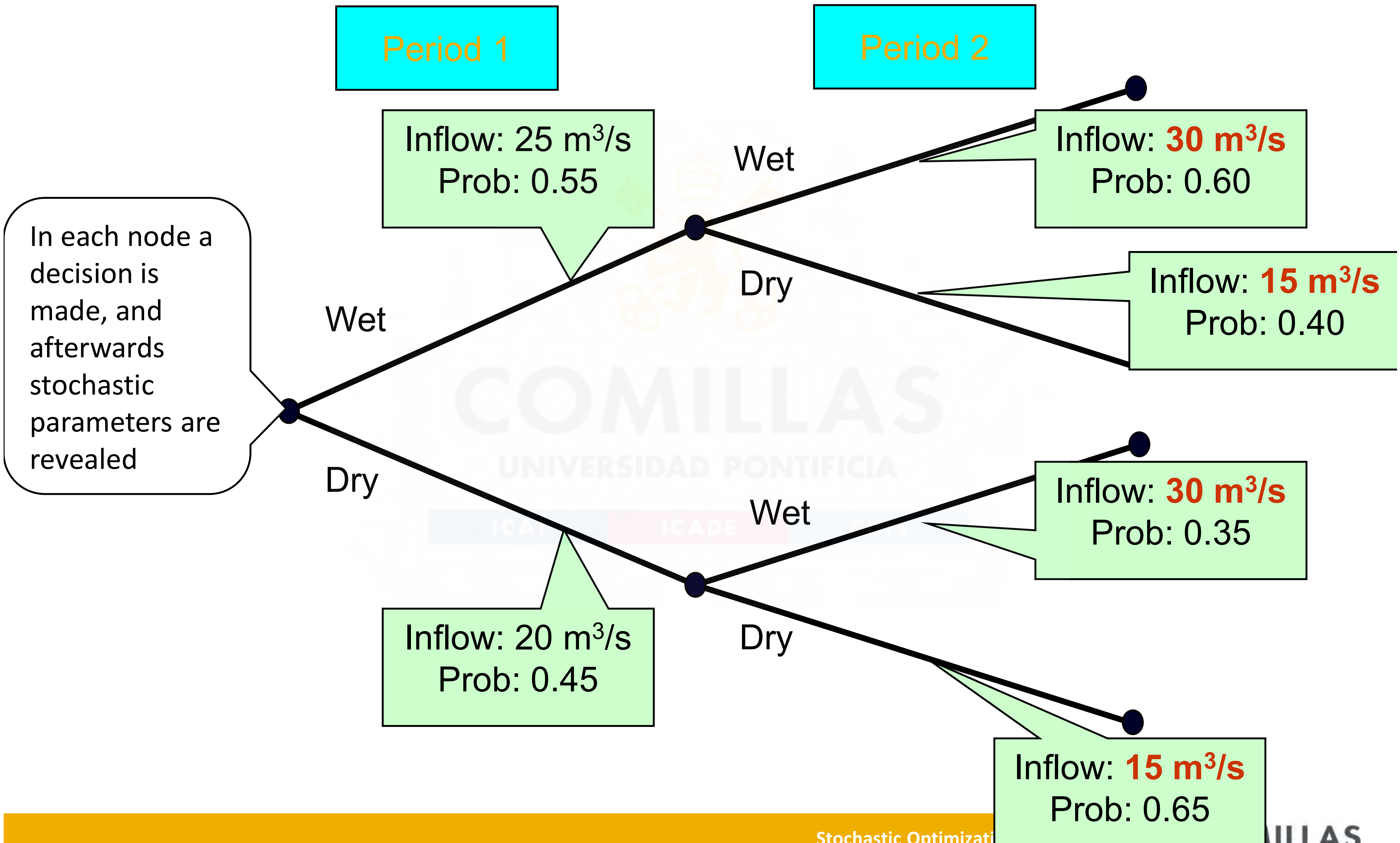




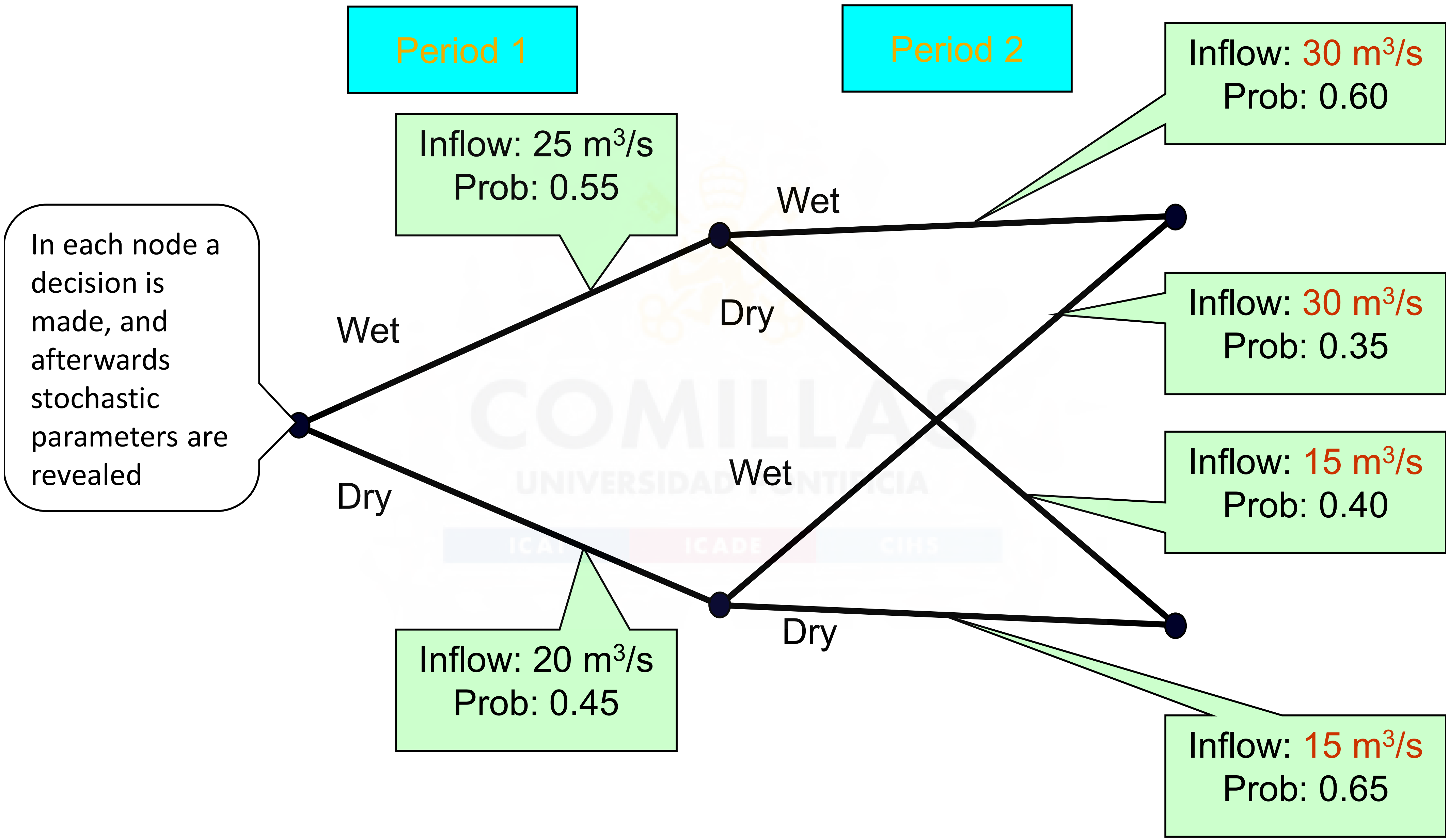
Scenario tree example



Recombining scenario tree example



Recombining scenario tree example



Recombining scenario tree

- The inflows depend on the scenarios in each period.
  - In the previous tree in period 2 there are four scenarios, 30, 25, 20 y 10 m^3/s.
  - In the previous recombining tree, in period 2 there are only two scenarios, 30 y 15 m^3/s.

Efect of the uncertainty representation

- Tree based in
  - Historical series (usually in a reduced number)
  - Synthetic series
- Tree generation based on
  - Results of the stochastic optimization in the first stage
  - Statistical properties (moments, distances) of the original series and the scenario tree
- Tree options
  - Recombining
  - Not recombining

Scenario tree trade-off

- Big scenario tree and simplified optimization model
  - Where do we branch the tree?

- Small scenario tree and realistic optimization model

Where do we branch?

- Where there are high variability of stochastic values
  - Winter and spring in hydro inflows
- Short-term future will affect more that long-term future
  - If the scope of the model is from January to December branching in winter and spring will more relevant than branching in autumn

Scenario tree generation (i)

- Univariate series (one inflow)
  - Distance from the cluster centroid to each series from a period to the last one

- Multivariate series (several inflows)
  - Distance from the multidimensional cluster centroid to each series of each variable from a period to the last one

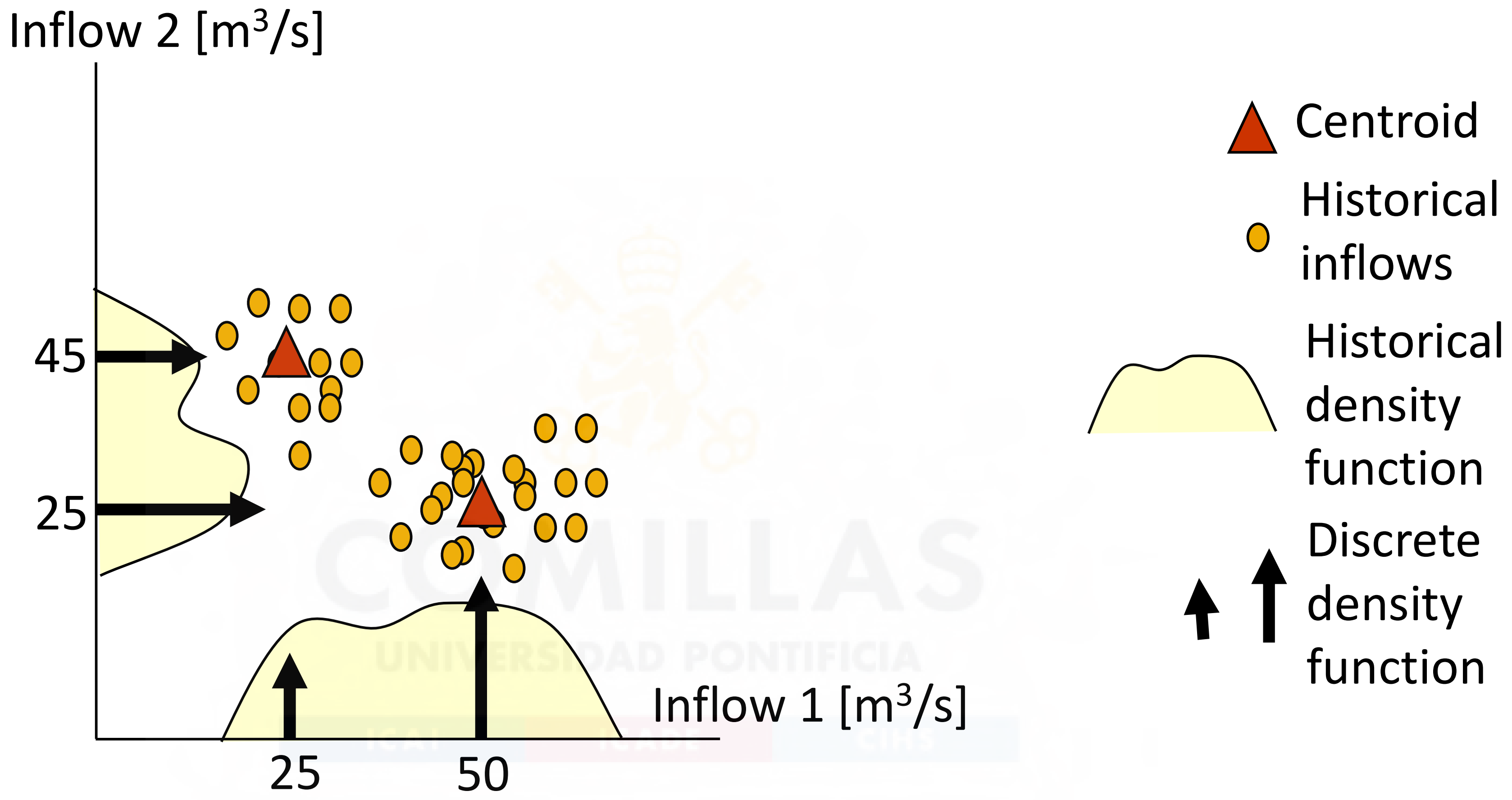
Scenario tree generation (ii)

- There is no established method to obtain a unique scenario tree
- A multivariate scenario tree is obtained by neural gas clustering technique that simultaneously takes into account the main stochastic series and their spatial and temporal dependencies.
- Contamination: very extreme scenarios can be artificially introduced with a very low probability
- Number of scenarios generated enough for observing parameter variability

Common approach for tree generation

- Process divided into two phases:
  - Generation of a scenario tree.
    Neural gas method.
  - Reduction of a scenario tree.
    Using probabilistic distances.

Clustering in two dimensions



Centroids have the minimum distance to their corresponding points

Their probability is proportional to the number of points included in the centroid

Scenario tree generation

- Idea
  - Minimize the distance of the scenario tree to the original series
  - Predefined maximum tree structure (2x2x2x2x1x1x1x1x1x1x1, for example)
  - Extension of clustering technique to consider many inflows and many periods

- J.M. Latorre, S. Cerisola, A. Ramos *Clustering Algorithms for Scenario Tree Generation. Application to Natural Hydro Inflows* European Journal of Operational Research 181 (3): 1339-1353 Sep 2007

Neural gas algorithm (i)

- Soft competitive learning method
  - All the scenarios/centroids are adapted for each new series introduced
  - Decreasing adapting rate
- Iterative adaptation of the scenario/centroid as a function of how close is to a new series randomly chosen
- Modifications to this method
  - Initialization: considers the tree structure of the centroids
  - Adaptation: the modification of each node is the average of the corresponding for belonging to each scenario

Neural gas algorithm (ii)

1. Initialize the tree $\{\omega^k\}$ with randomly chosen series.
2. Choose randomly a new series ω.
3. Compute the distance of the scenario tree to the series:

$$d^k = \left\|\omega - \omega^k\right\| \text{ for } k = 1,2,\ldots,K$$

4. Sort by increasing order these distances and store in o^k the order of each scenario in this sequence
5. Compute the modification of each node:

$$\Delta\omega_t^n = \varepsilon(j)\frac{\sum\limits_{k=1,\ldots,K/\omega_t^n\in\omega^k} h_\lambda(o^k)\cdot(\omega-\omega^k)}{\sum\limits_{k=1,\ldots,K/\omega_t^n\in\omega^k} 1}$$

j iteration
$\varepsilon(j)$ (adaptation) exponential
$h(o)$ (neighborhood) exponential

6. If maximum number of iterations has not been reached go to 2.

Natural inflows (I)

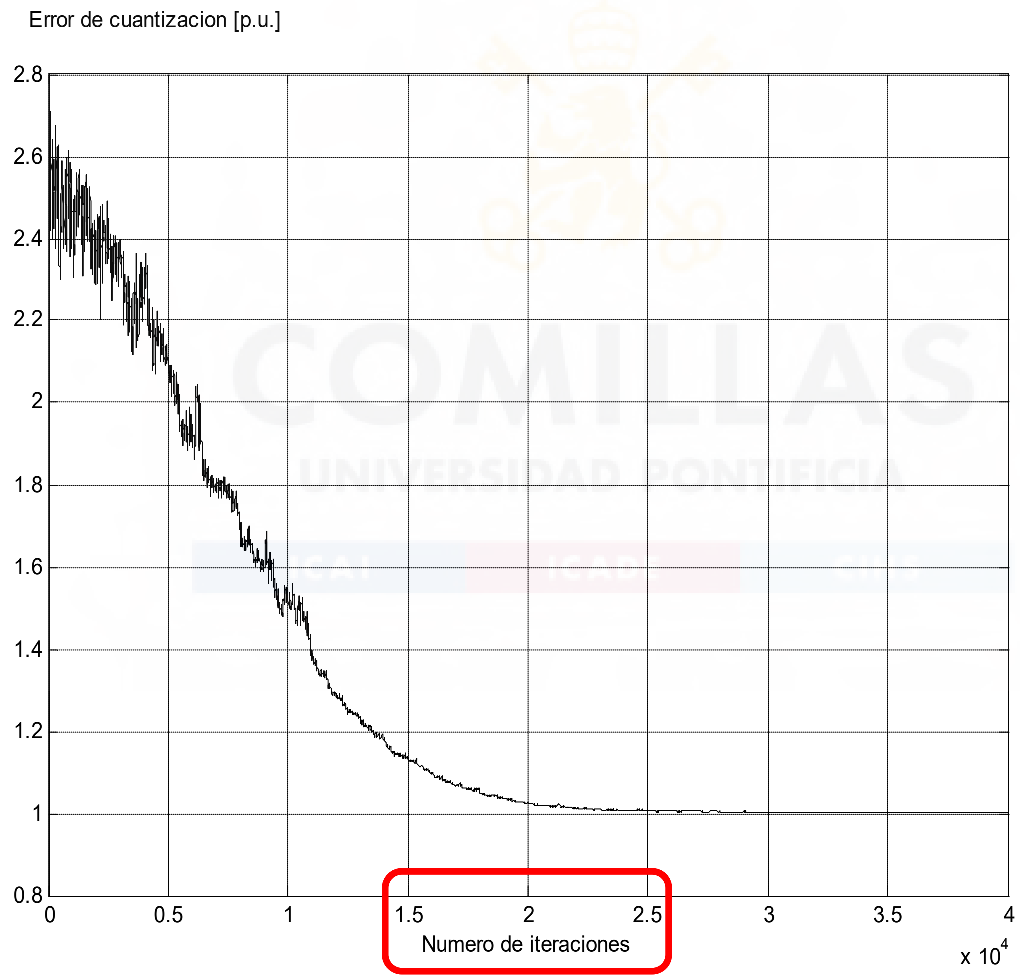
- Data series for a almost 30 years.
- Weekly data in m^3/s.
- Corresponding to 8 measurement points in 3 basins.
- Organized in natural hydrological years, from October to September.

Maximum structure of the tree:

- 16 scenarios
- Branches in stages 5, 9, 13 and 17

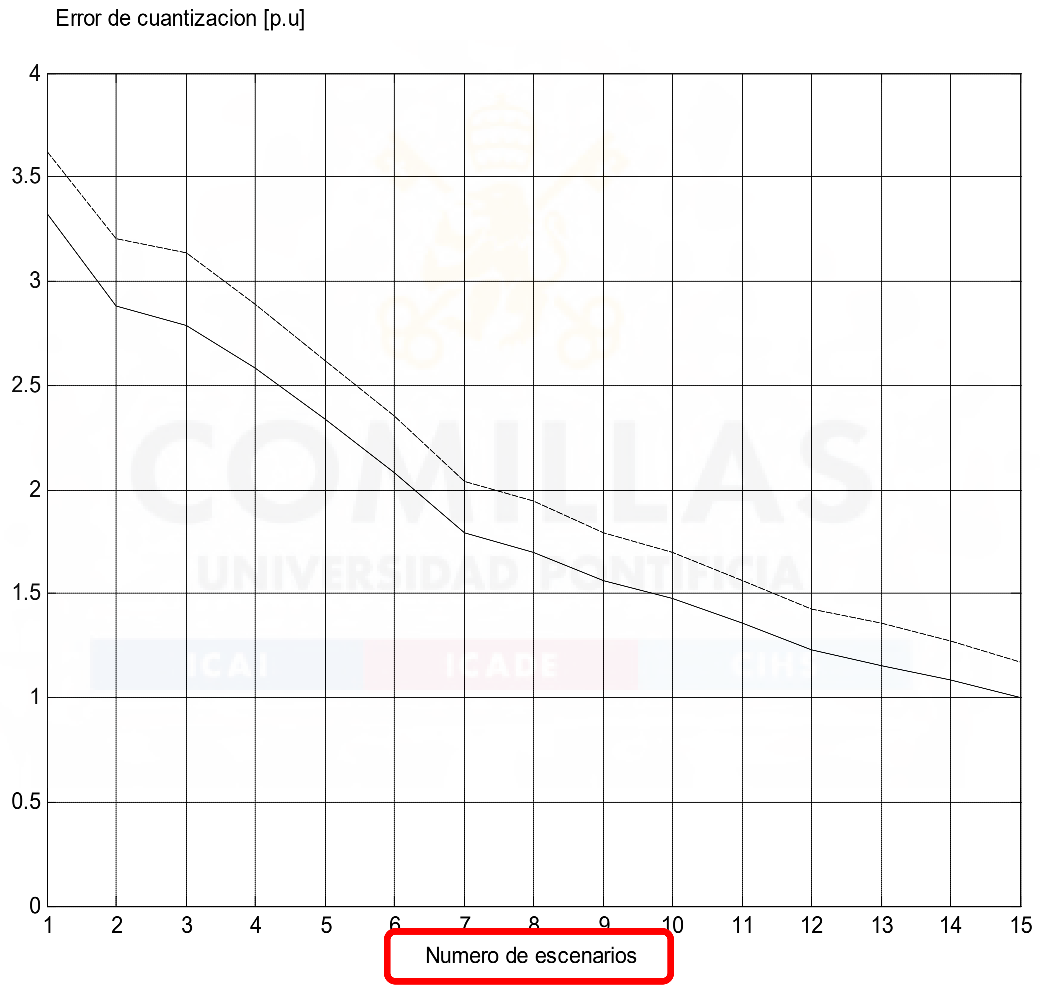
Natural inflows (II)

- Quantization error: distance from the series to the branches of the tree they belong to



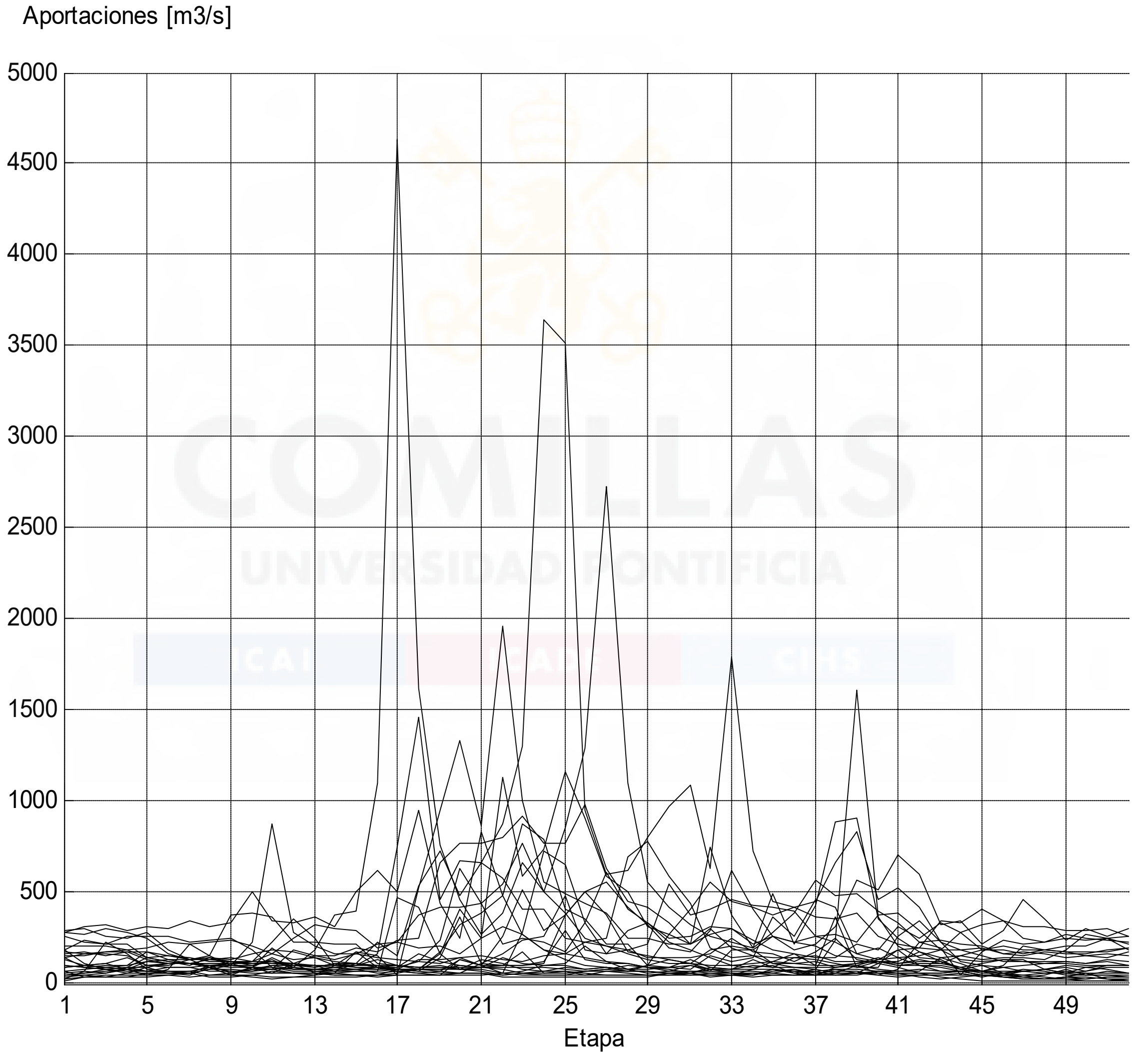
Natural inflows (III)

- Relative quantization error with number of scenarios



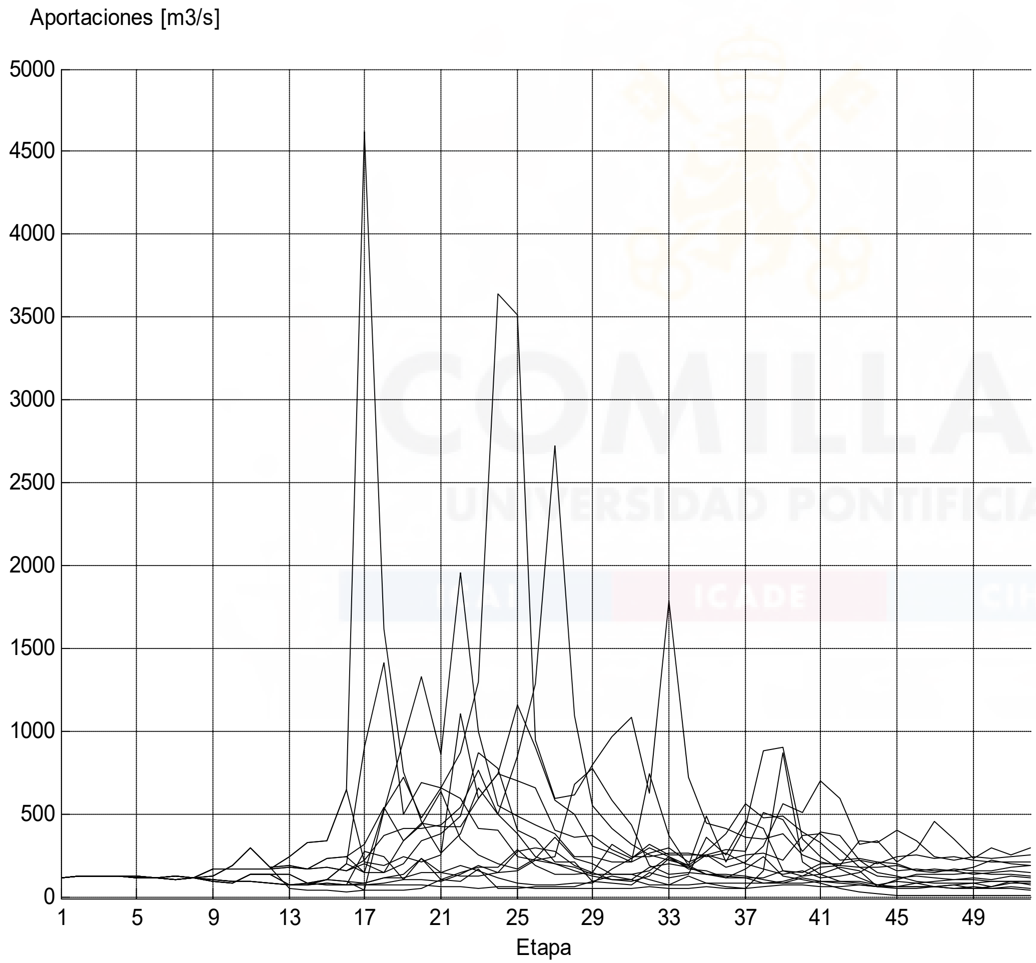
Natural inflows (V)

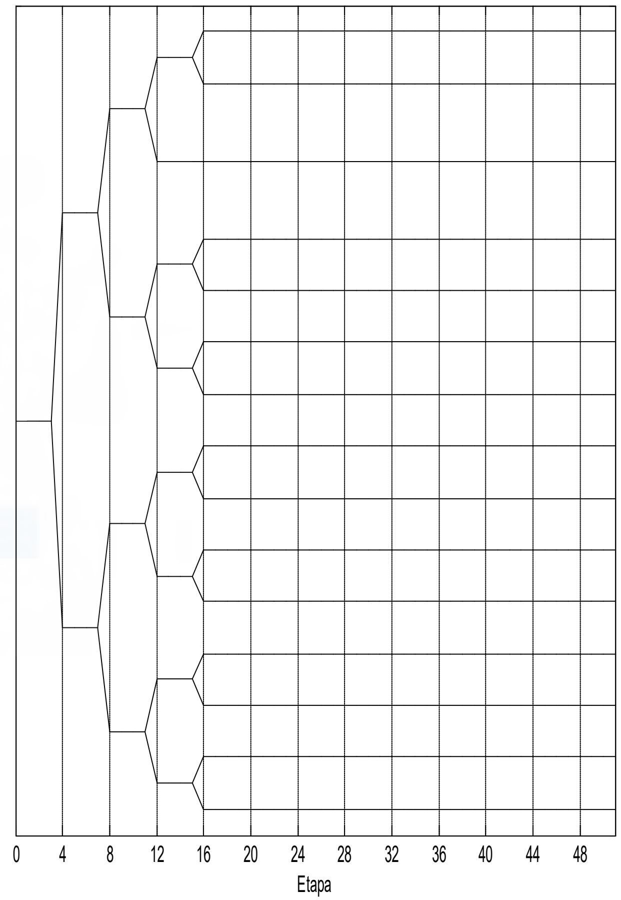
- Data series for one hydro inflow:



Natural inflows (VI)

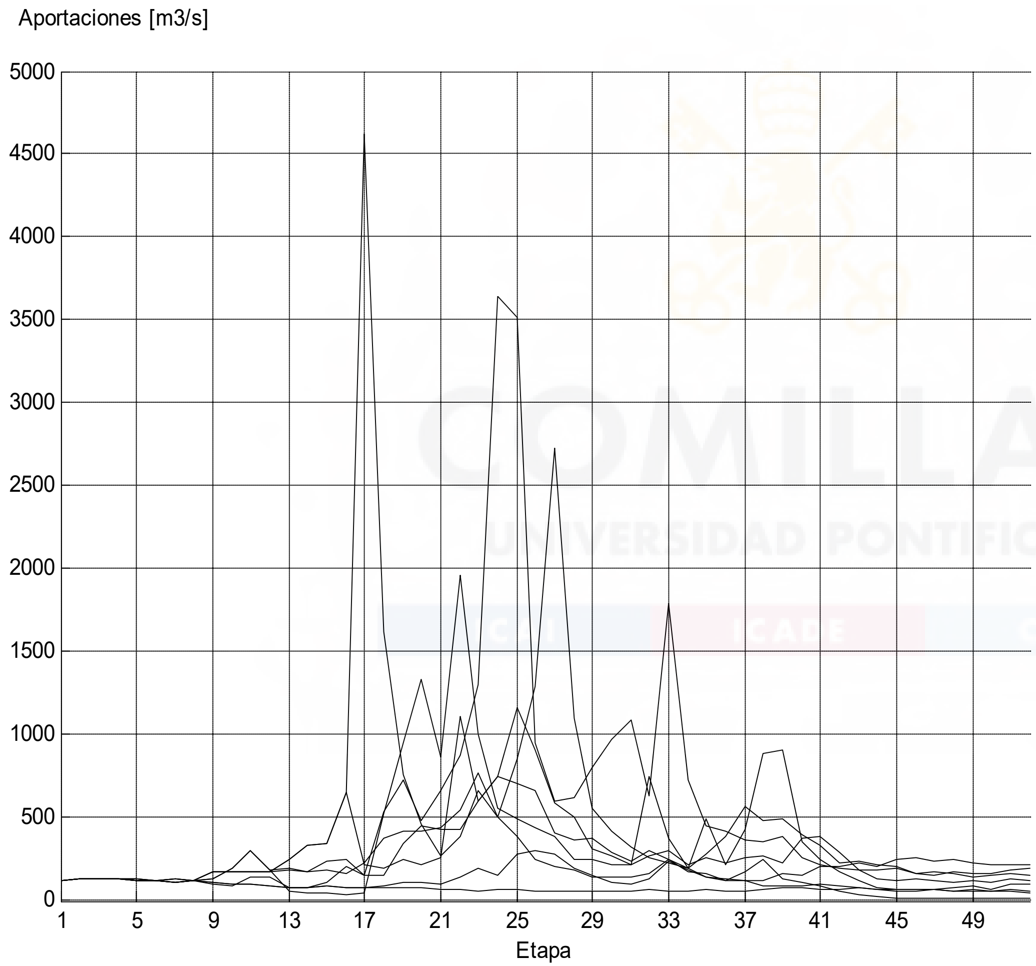
- Initial scenario tree for one hydro inflow:

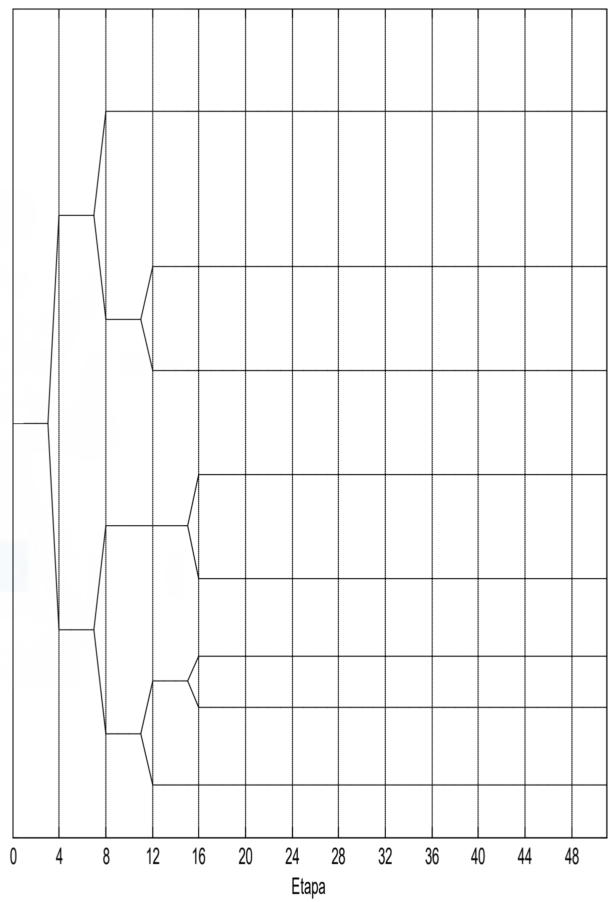




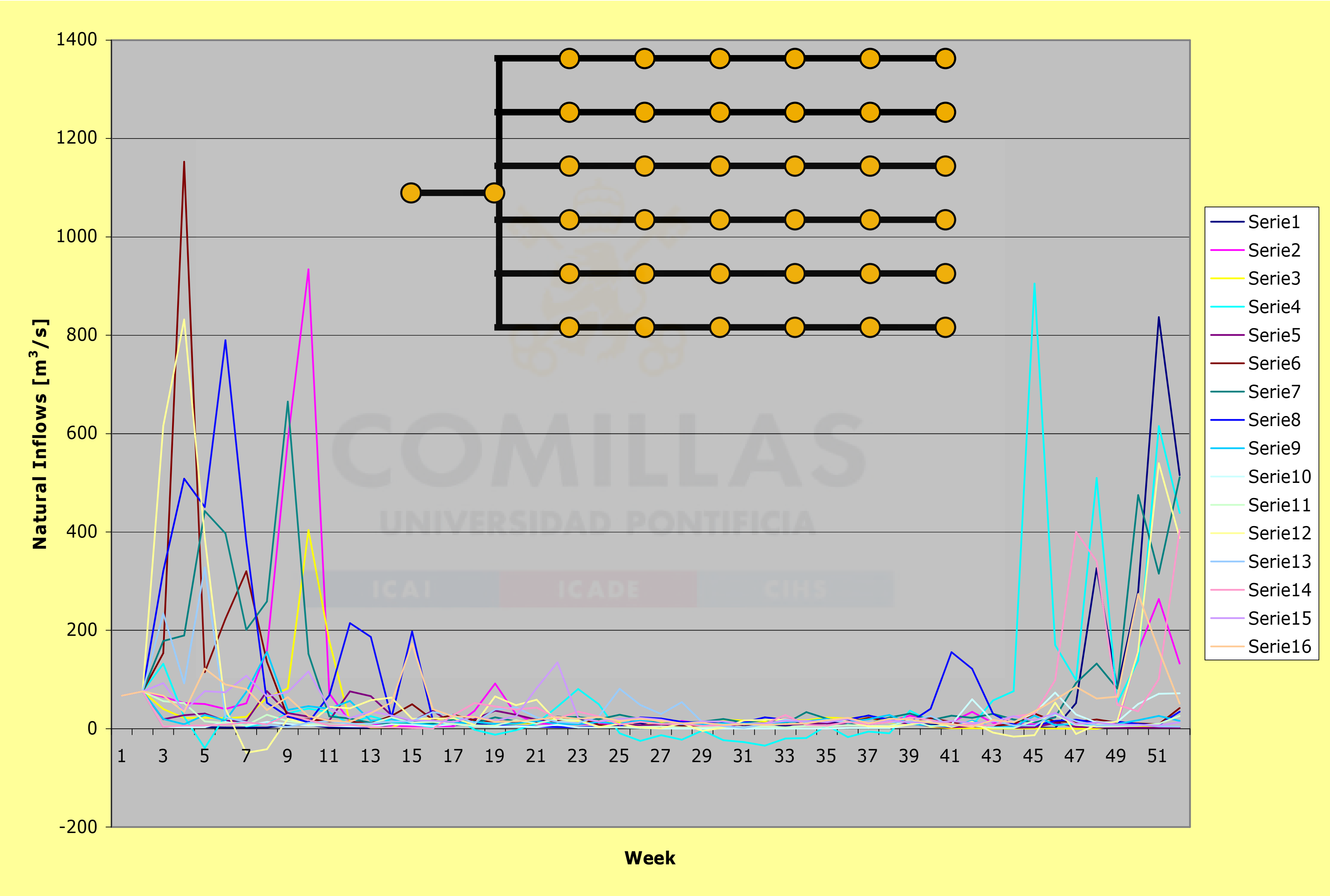
Natural inflows (VII)

- Reduced scenario tree for one hydro inflow:

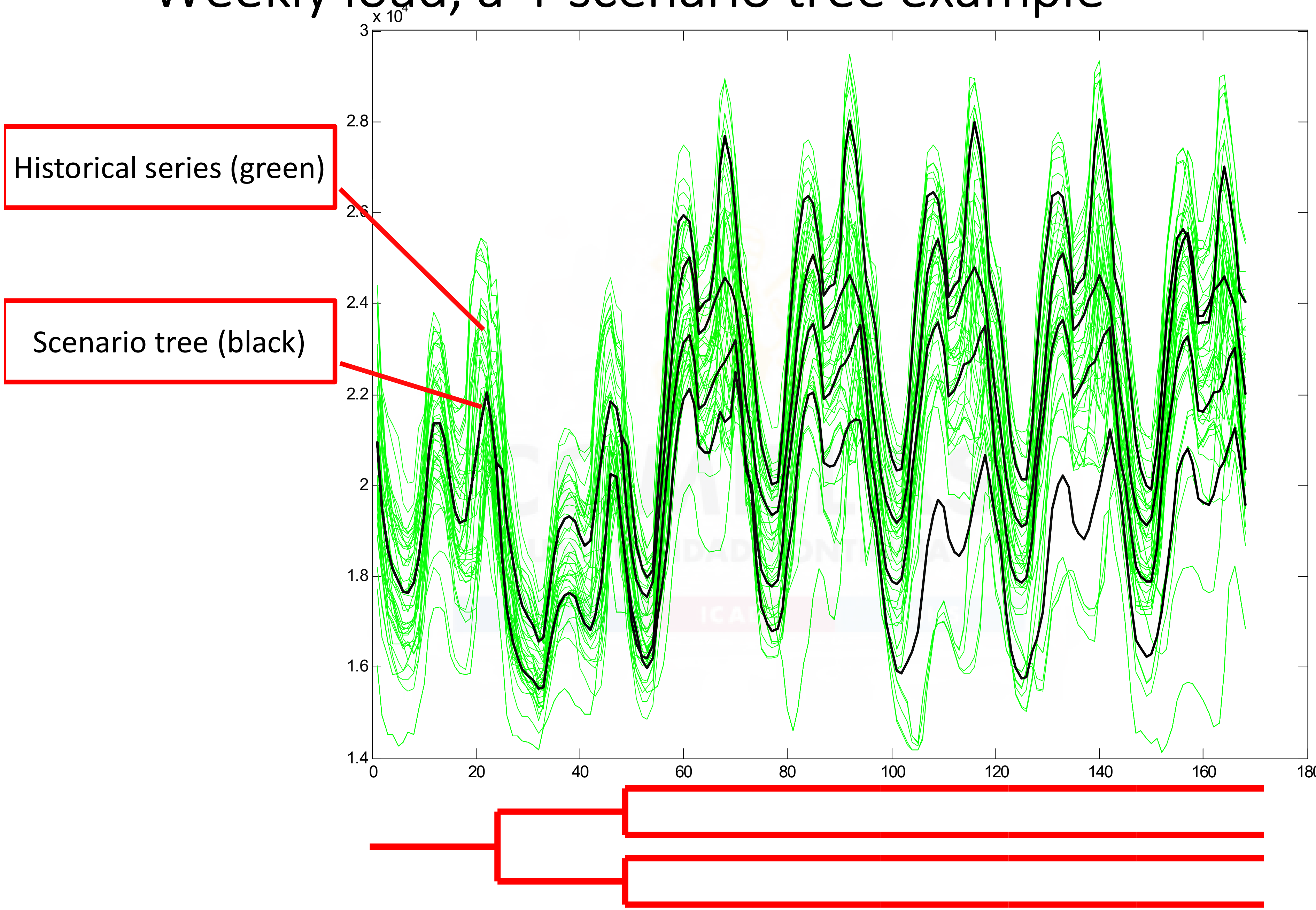




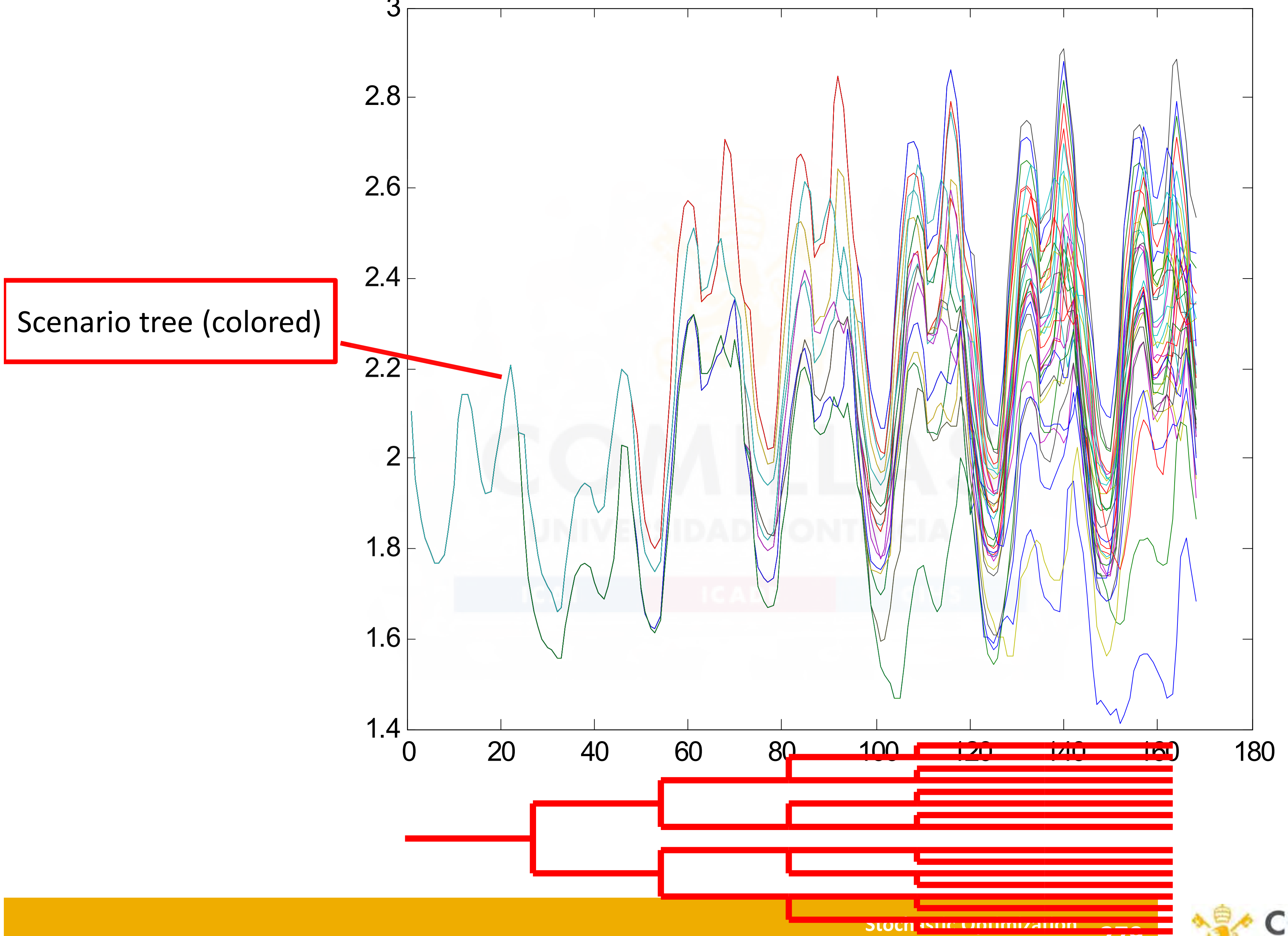
Natural inflows: scenario tree



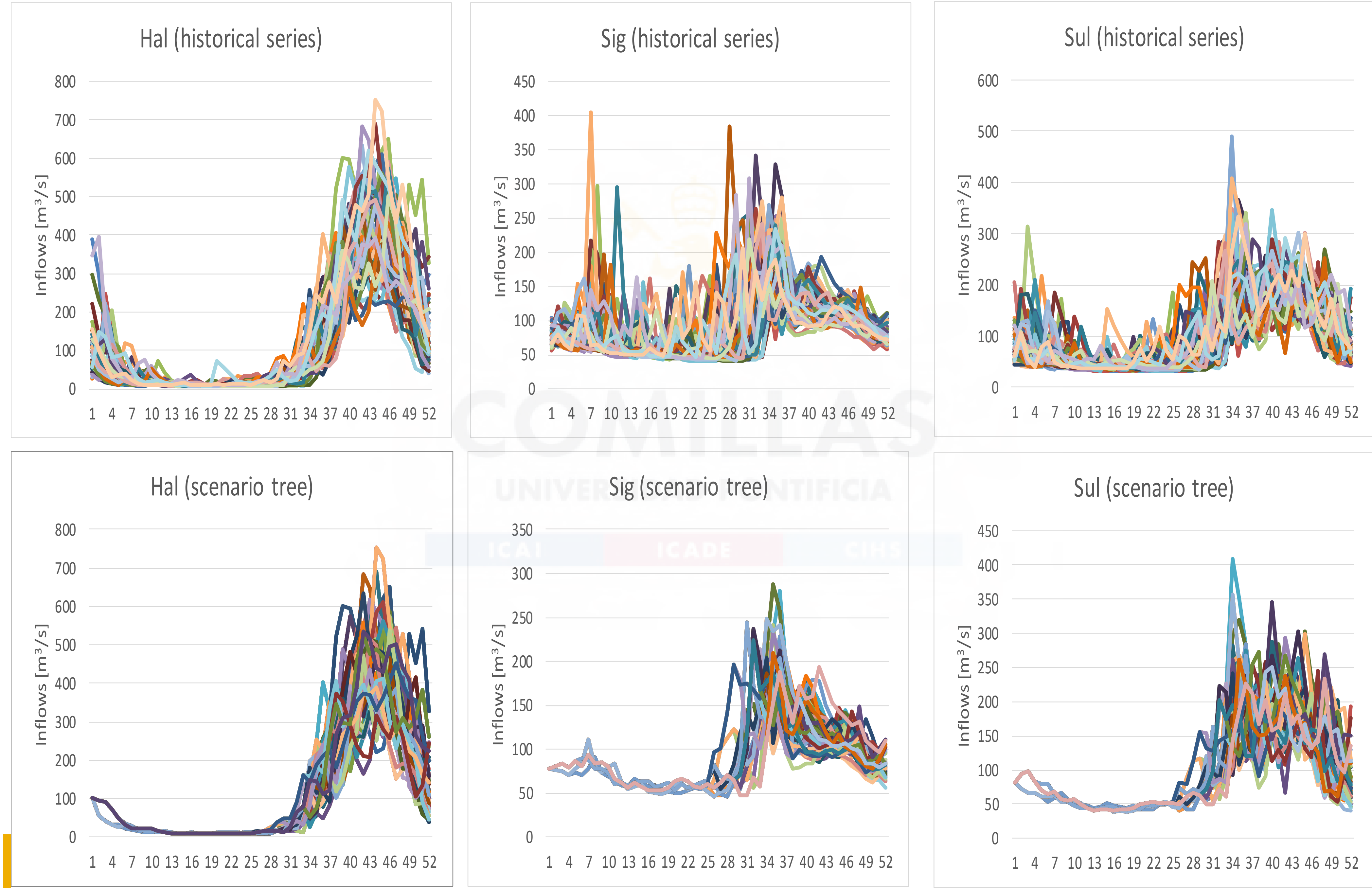
Weekly load, a 4-scenario tree example



Weekly load: a 32-scenario tree example



Scenario tree for hydro inflows in Iceland



10

Decomposition in two-stage and multistage stochastic programming

Two-stage stochastic linear programming PLE-2

- O.F. minimizes first-stage costs and expected value of second-stage costs

$$
\begin{aligned}
&\min_{x_1, x_2^\omega} c_1^T x_1 + \sum_{\omega \in \Omega} p^\omega c_2^{\omega T} x_2^\omega \\
&A_1 x_1 \qquad\qquad\qquad = b_1 \\
&B_1^\omega x_1 \quad + A_2^\omega x_2^\omega \quad = b_2^\omega \\
&x_1, \qquad\quad x_2^\omega \qquad \geq 0
\end{aligned}
$$

- If A_2^ω doesn't depend on ω it is called fixed resource
- Structure of the constraint matrix

A_1			
B_1^1	A_2^1		
B_1^2		A_2^2	
B_1^3			A_2^3

Deterministic equivalent problem (DEP)

- State space is small
- Formulation of the deterministic equivalent problem

$$
\begin{aligned}
\min_{x_1, x_2^{\omega_1}, x_2^{\omega_2}, x_2^{\omega_3}} \quad & c_1^T x_1 + p^{\omega_1} c_2^{\omega_1 T} x_2^{\omega_1} + p^{\omega_2} c_2^{\omega_2 T} x_2^{\omega_2} + p^{\omega_3} c_2^{\omega_3 T} x_2^{\omega_3} \\
& A_1 x_1 = b_1 \\
& B_1^{\omega_1} x_1 + A_2^{\omega_1} x_2^{\omega_1} = b_2^{\omega_1} \\
& B_1^{\omega_2} x_1 + A_2^{\omega_2} x_2^{\omega_2} = b_2^{\omega_2} \\
& B_1^{\omega_3} x_1 + A_2^{\omega_3} x_2^{\omega_3} = b_2^{\omega_3} \\
& x_1, \; x_2^{\omega_1}, \; x_2^{\omega_2}, \; x_2^{\omega_3}, \geq 0
\end{aligned}
$$

- In Benders decomposition subproblem results separable and has the same structure in the constraints

$$
c_2 = \begin{pmatrix} p^{\omega_1} c_2^{\omega_1} \\ p^{\omega_2} c_2^{\omega_2} \\ p^{\omega_3} c_2^{\omega_3} \end{pmatrix} \quad
B_1 = \begin{pmatrix} B_1^{\omega_1} \\ B_1^{\omega_2} \\ B_1^{\omega_3} \end{pmatrix} \quad
A_2 = \begin{pmatrix} A_2^{\omega_1} & & \\ & A_2^{\omega_2} & \\ & & A_2^{\omega_3} \end{pmatrix} \quad
b_2 = \begin{pmatrix} b_2^{\omega_1} \\ b_2^{\omega_2} \\ b_2^{\omega_3} \end{pmatrix} \quad
x_2 = \begin{pmatrix} x_2^{\omega_1} \\ x_2^{\omega_2} \\ x_2^{\omega_3} \end{pmatrix}
$$

Decomposition in PLE-2

- Bd Relaxed Master monocut and multicut

$$\begin{aligned} &\min_{x_1,\theta_2} c_1^T x_1 + \theta_2 \\ &A_1 x_1 = b_1 \\ &\sum_{\omega\in\Omega} p^\omega \pi_2^{\omega l T} B_1^\omega x_1 + \theta_2 \ge \sum_{\omega\in\Omega} p^\omega \pi_2^{\omega l T} b_2^\omega \quad l = 1,\dots,j \\ &x_1 \ge 0 \end{aligned}$$

$$\begin{aligned} &\min_{x_1,\theta_2^\omega} c_1^T x_1 + \sum_{\omega\in\Omega} p^\omega \theta_2^\omega \\ &A_1 x_1 = b_1 \\ &\pi_2^{\omega l T} B_1^\omega x_1 + \theta_2^\omega \ge \pi_2^{\omega l T} b_2^\omega \quad \omega\in\Omega \quad l = 1,\dots,j \\ &x_1 \ge 0 \end{aligned}$$

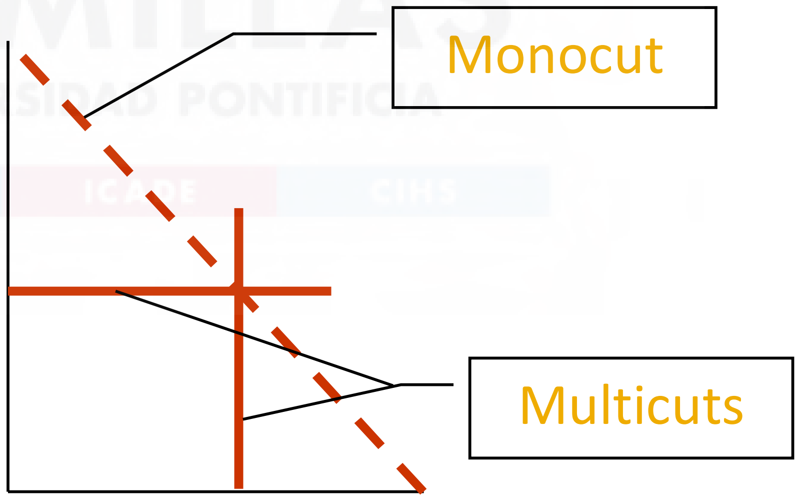
  - Multicut approximates independently each scenario
  - Monocut approximates the weighted sum of scenarios
- Bd Subproblem

$$\begin{aligned} &\min_{x_2^\omega} c_2^{\omega T} x_2^\omega \\ &A_2^\omega x_2^\omega = b_2^\omega - B_1^\omega x_1^l \quad : \pi_2^\omega \\ &x_2^\omega \ge 0 \end{aligned}$$

One subproblem for each scenario

Monocut vs. multicut

- Master problem
  - Monocut $(m_1 + j) \times (n_1 + 1)$
  - Multicut $(m_1 + j\Omega) \times (n_1 + \Omega)$
- Multicut convenient when m_2 is large and Ω no much larger than n_1. Requires less Benders iterations but more cumbersome



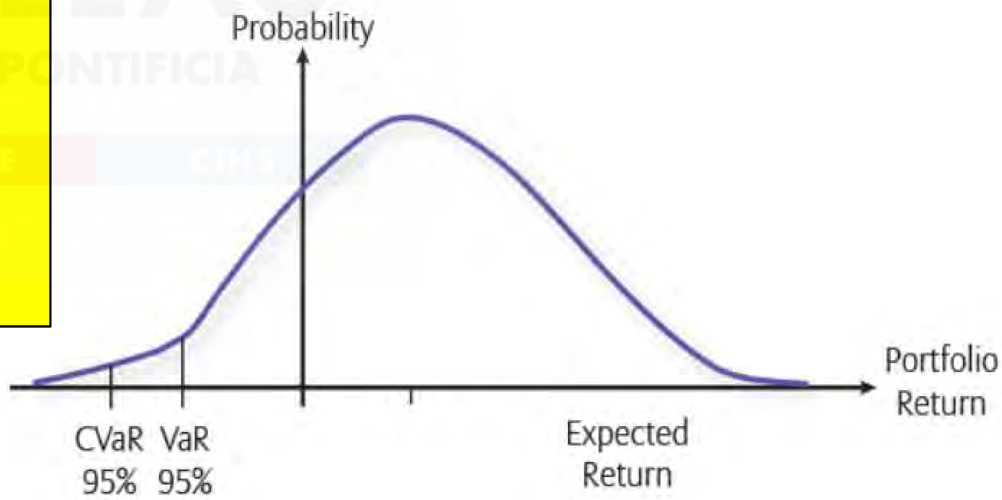
Risk measures.
Conditional Value at Risk ($CVaR$). Value at Risk (VaR)

$$CVaR = \max_{VaR, y^{\omega}} \left(VaR - \frac{1}{\beta}\sum_{\omega} p^{\omega} y^{\omega}\right)$$
$$y^{\omega} \geq VaR - x^{\omega}, \forall \omega \quad : q^{\omega}$$
$$y^{\omega} \geq 0, \forall \omega$$

and its dual problem

$$CVaR = \min_{q^{\omega}} \sum_{\omega} x^{\omega} q^{\omega}$$
$$\sum_{\omega} q^{\omega} = 1 \quad : VaR$$
$$q^{\omega} \leq \frac{p^{\omega}}{\beta}, \forall \omega \quad : -y^{\omega}$$
$$q^{\omega} \geq 0$$

Variables
y^{ω} profit value below VaR
q^{ω} modified probability of scenario ω
Parameters
x^{ω} objective function (profit)
p^{ω} probability of each scenario ω
β probability (5 %)



Risk-constrained profit-based operation model

$$\max_{x^\omega, y^\omega, CVaR, VaR} (1-\mu) \sum_\omega p^\omega x^\omega + \mu \, CVaR$$
$$CVaR = VaR - \frac{1}{\beta} \sum_\omega p^\omega y^\omega$$
$$y^\omega \geq VaR - x^\omega, \forall \omega$$
$$y^\omega \geq 0, \forall \omega$$
$$x^\omega \in K$$

K model constraints
μ risk weight factor

Complicating constraints because link all the scenarios

x^ω can be expressed as a linear combination of the vertices

$$K = \left\{ x^\omega = \sum_l x^{l\omega} \lambda^l \mid \sum_l \lambda^l = 1, \lambda^l \geq 0, \forall l \right\}$$

Risk-constrained profit-based operation model

- DW Complete master problem

$$\max_{\lambda^l, y^\omega, VaR} (1-\mu)\sum_\omega p^\omega \sum_l x^{l\omega}\lambda^l + \mu\left(VaR - \frac{1}{\beta}\sum_\omega p^\omega y^\omega\right)$$

$$y^\omega \geq VaR - \sum_l x^{l\omega}\lambda^l, \forall\omega \quad : \pi^\omega$$

$$\sum_l \lambda^l = 1 \quad : \theta$$

$$\lambda^l \geq 0, \forall l$$

$$y^\omega \geq 0, \forall\omega$$

π^ω, θ dual variables

$$\max \left((1-\mu)\sum_\omega p^\omega x^{l\omega} \quad -\frac{\mu}{\beta}p^\omega \quad \mu \right) \begin{pmatrix} \lambda^l \\ y^\omega \\ VaR \end{pmatrix}$$

$$\begin{pmatrix} x^{l\omega} & 1 & -1 \\ 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} \lambda^l \\ y^\omega \\ VaR \end{pmatrix} \begin{matrix} \geq \\ = \end{matrix} \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

$$\lambda^l \geq 0, \forall l$$

$$y^\omega \geq 0, \forall\omega$$

Risk-constrained profit-based operation model

- Taking the dual

$$\min \theta$$

$$\begin{pmatrix} x^{l\omega} & 1 \\ 1 & 0 \\ -1 & 0 \end{pmatrix} \begin{pmatrix} \pi^{\omega} \\ \theta \end{pmatrix} \begin{matrix} \geq \\ \geq \\ = \end{matrix} \begin{pmatrix} (1-\mu)\sum_{\omega} p^{\omega} x^{l\omega} \\ -\frac{\mu}{\beta} p^{\omega} \\ \mu \end{pmatrix}$$

$$\pi^{\omega} \leq 0, \forall \omega$$

$$\min_{\pi^{\omega},\theta} \theta$$

$$\theta \geq (1-\mu)\sum_{\omega} p^{\omega} x^{l\omega} - \sum_{\omega} x^{l\omega} \pi^{\omega}$$

$$\pi^{\omega} \geq -\frac{\mu}{\beta} p^{\omega}$$

$$-\sum_{\omega} \pi^{\omega} = \mu$$

$$\pi^{\omega} \leq 0, \forall \omega$$

Risk-constrained profit-based operation model

- LR Relaxed master problem

If we define $q^{\omega} = -\frac{\pi^{\omega}}{\mu}$

$$\min_{q^{\omega},\theta} \theta$$
$$\theta \geq (1-\mu)\sum_{\omega} p^{\omega} x^{j\omega} + \mu \sum_{\omega} x^{j\omega} q^{\omega}, \forall j$$
$$0 \leq q^{\omega} \leq \frac{p^{\omega}}{\beta}, \forall \omega$$
$$\sum_{\omega} q^{\omega} = 1$$

- LR subproblem

$$\max_{x^{\omega}} (1-\mu)\sum_{\omega} p^{\omega} x^{\omega} + \mu \sum_{\omega} q^{\omega} x^{\omega}$$
$$x^{\omega} \in K$$

Multistage stochastic linear programming PLE-P

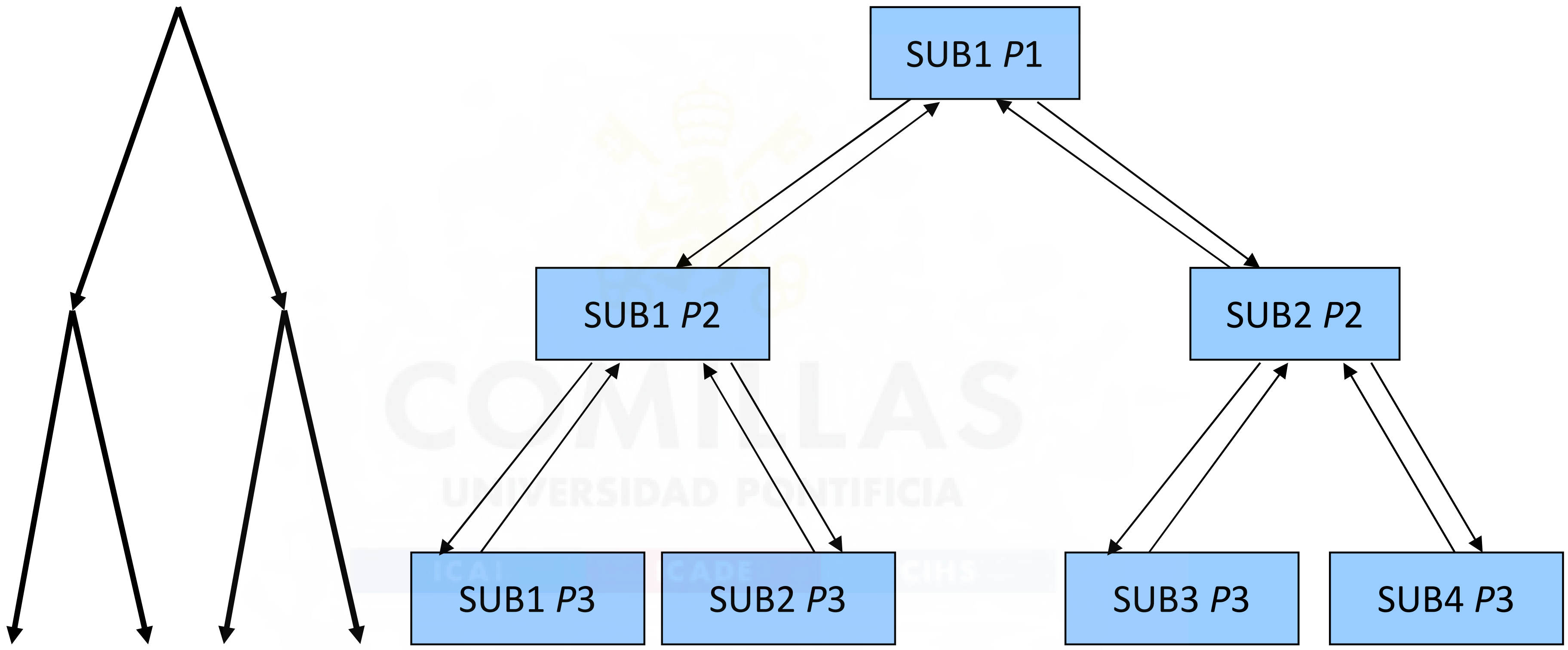
- O.F. minimizes expected costs of all the stages

$$
\begin{aligned}
&\min_{x_p^{\omega_p}} \sum_{p=1}^{P} \sum_{\omega_p \in \Omega_p} p_p^{\omega_p} c_p^{\omega_p T} x_p^{\omega_p} \\
&B_{p-1}^{\omega_p} x_{p-1}^{\omega_{p-1}} + A_p^{\omega_p} x_p^{\omega_p} = b_p^{\omega_p} \qquad p = 1, \dots, P \\
&x_p^{\omega_p} \geq 0 \\
&B_0^{\omega_1} \equiv 0
\end{aligned}
$$

- Probabilities $p_p^{\omega_p}$ are conditional
- Constraint matrix

A_1						
B_1^1	A_2^1					
B_1^2		A_2^2				
	B_2^1		A_3^1			
	B_2^2			A_3^2		
		B_2^3			A_3^3	
		B_2^4				A_3^4

Multistage stochastic problem. Nested Benders decomposition



Decomposition of PLE-P

- NBd Relaxed Master monocut

$$\min_{x_p^{\omega_p},\theta_{p+1}^{\omega_p}} c_p^{\omega_p T} x_p^{\omega_p} + \theta_{p+1}^{\omega_p}$$

$$A_p^{\omega_p} x_p^{\omega_p} = b_p^{\omega_p} - B_{p-1}^{\omega_p} x_{p-1}^{a(\omega_p)l} \quad : \pi_p^{\omega_p}$$

$$\sum_{k\in d(\omega_p)} p^{\omega_p} \pi_{p+1}^{klT} B_p^k x_p^{\omega_p} + \theta_{p+1}^{\omega_p} \geq q_p^{\omega_p} = \sum_{k\in d(\omega_p)} p^{\omega_p} \left(\pi_{p+1}^{klT} b_{p+1}^k + \eta_{p+1}^{klT} q_{p+1}^k \right) \quad : \eta_p^{\omega_p}$$

$$x_p^{\omega_p} \geq 0$$

where $a(\omega_p)$ is the ancestor and $d(\omega_p)$ is the descendant of a given subproblem

Stochastic multistage decomposition

Step 0 Set $I_t^{\xi_t} = J_t^{\xi_t} = 0$. Set $\theta_t^{\xi_t} \equiv 0$ at the initial iteration

Step 1 Forward pass:

Repeat for $t = 1, \dots, T$

Repeat for each node ξ_t of stage t

Solve $(RP_t^{\xi_t})$

If feasible: obtain solution $x_t^{\xi_t}$

If $t = 1$ obtain lower bound $\underline{z} = v(RP_1^{\xi_1})$

If infeasible: stop forward pass, set $T' = t$ and go to Step 4

Step 2 Upper bound computation:

Evaluate objective function of the complete problem with the primal solutions so far obtained. $\overline{z} = v(P)$

Step 3 (stopping rule)

If $\overline{z} - \underline{z} < tol$ stop, $x_t^{\xi_t}$ is optimal solution, else go to Step 4

Step 4 Backward pass

Repeat for $t = T', \dots, 1$

Repeat for each node ξ_t of stage t

Solve $(RP_t^{\xi_t})$

If feasible: obtain objective $\theta_t^{\xi_t, i} = v(RP_t^{\xi_t})$ and dual values $\pi_t^{\xi_t, i}$

Augment $I_t^{\xi_t} = I_t^{\xi_t} + 1$

If infeasible: obtain sum of infeasibilities $\tilde{\theta}_t^{\xi_t, j}$ and dual values $\tilde{\pi}_t^{\xi_t, j}$

Augment $J_t^{\xi_t} = J_t^{\xi_t} + 1$

Go to step 1

Hydrothermal scheduling

- S. Cerisola, A. Ramos *Benders' decomposition for Mixed-Integer Hydrothermal Problems by Lagrangean Relaxation* 14th Power Systems Computation Conference (PSCC '02) Session 05, Paper 6, Pages 1-8. Seville, Spain June 2002
- S. Cerisola, A. Ramos *Node Aggregation in Stochastic Nested Benders' decomposition Applied to Hydrothermal Coordination* 6th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS) RUM-102. Madeira, Portugal September 2000
- J.C. Enamorado, A. Ramos, T. Gómez *Multi-area Decentralized Optimal Hydro-Thermal Coordination by the Dantzig-Wolfe Decomposition Method* 2000 IEEE Power Engineering Society Summer Meeting 4: 2027-2032. Seattle, WA, USA July 2000 10.1109/PESS.2000.866958
- J.M. Latorre, S. Cerisola, A. Ramos, R. Palacios *Analysis of Stochastic Problem Decomposition Algorithms in Computational Grids* Annals of Operations Research 166 (1): 355-373 Feb 2009 10.1007/s10479-008-0476-1
- A. Ramos, L. Muñoz, F. Martínez-Córcoles, V. Martín-Corrochano *A medium term bulk production cost model based on decomposition techniques* 1995 IEEE PowerTech Symposium on Electric Power Engineering 5: 110-116 Stockholm, Sweden June 1995166 (1): 355-373 Feb 2009 10.1007/s10479-008-0476-1

11

Improvements in decomposition techniques

Areas for improvement

- On the problem solution:
  - Master problem solution time (solution time can be large because of size, integer variables or many cuts)
  - Subproblem solution time (solution time can be large because of many scenarios or many constraints)
- On the decomposition algorithm
  - Decrease the number of iterations
- Practical benefit achieved must be assessed on a case-by-case basis
  - Most of them involve trade-offs that must be assessed individually

S. Lumbreras, A. Ramos *How to Solve the Transmission Expansion Planning (TEP) Problem Faster: Acceleration Techniques Applied to Benders Decomposition* IET Generation, Transmission & Distribution 10: 2351-2359, Jul 2016 10.1049/iet-gtd.2015.1075

Optimization technique for any problem

- Optimization method used for the problems

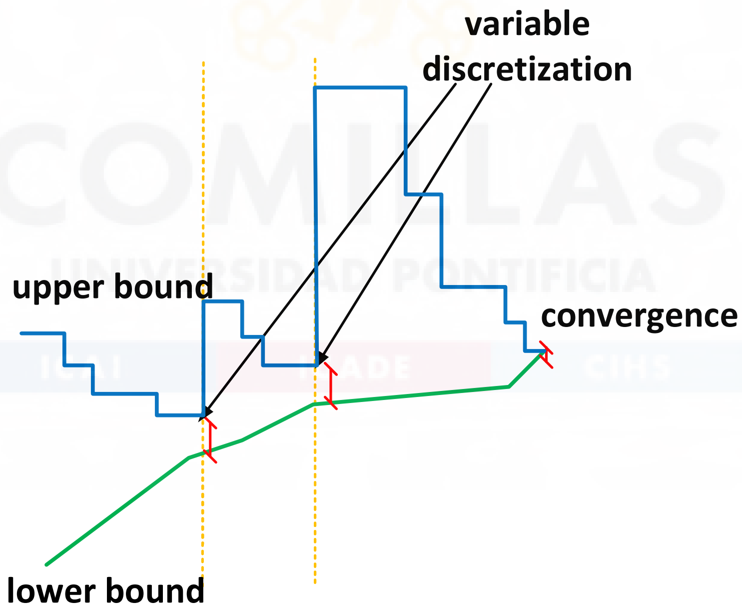
  Problems solved many times with modifications.

  In Benders decomposition the master problem adds constraints and the subproblem changes the constraint RHS.

  – Simplex dual method is the initial candidate. Try primal simplex method or interior point method.
  – Use the best solver (CPLEX, GUROBI, XPRESS)
  – Solver tuning: use of non default values of solver parameters
  – Warm start: use of a previous basis (controlled by option BRATIO in GAMS or persistent solver in Pyomo) for the same GAMS model
  – Use of initial point taken from the deterministic equivalent problem for a scenario.

Optimization technique for the MIP master problem (i)

- Master problem relaxations
  - Binary variables complicate the resolution, so solving the LP relaxation is much quicker
  - It yields valid Benders cuts
  - Progressive discretization of variables to improve convergence

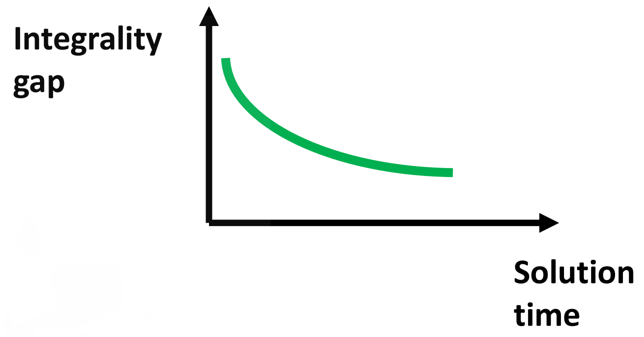


S. Lumbreras, A. Ramos and S. Cerisola *A Progressive Contingency Incorporation Approach for Stochastic Optimization Problems* IEEE Transactions on Power Systems 28 (2): 1452-1460, May 2013 10.1109/TPWRS.2012.2225077

Optimization technique for the master problem (ii)

- Sub-optimal MIP master problem solution
  - Early terminations of the master problem might improve convergence
  - Using any feasible solution
  - Rounding linearized solutions

  (feasibility must be checked)



- Advanced start: provide with initial promising solutions that might generate preliminary cuts for the first iterations prior to initiating a formal Benders algorithm
- Box-step method: limit the difference between consecutive proposals
- Introduction of additional constraints:
  - Ideally, they are constraints that should be met at the optimum, so they only eliminate not useful zones of the feasible region
    - From expert opinion
    - Data mining

Optimization technique for the MIP master problem (iii)

- Use of a more suitable technique
  - E.g. Constraint programming and logic-based methods
    - In many cases, complex problems include many logical constraints that make use of auxiliary binary variables
    - This greatly complicates the problem
    - There are techniques that have been specially developed for these problems, where the logical constraints are included explicitly (e.g. LOGMIP)
  - Alternative strategies to find master proposals
    - Metaheuristic techniques can be applied to find near-optimal solutions
- Local branching: divide the feasible region of the master problem into several sub-areas and searches in a neighborhood
    - If a better solution is found, the upper bound will be improved
    - If not, the explored solutions will result in an improved lower bound (more cuts)

Alternative optimization technique for subproblem

- Bunching: if the second-stage scenarios are similar we can solve only one scenario and calculate the others using the calculated sensitivities
- GAMS GUSS (Gather-Update-Solve-Scatter)
  - Use of sensitivity analysis for solving many similar problems
- Application of specific solution algorithms or even a series of increasingly accurate versions (problem formulation) of the subproblem (if they have increasing values of the o.f.) (e.g., lossless power flow for the first iterations and then with losses)
- Sub-optimal subproblem solutions (Zakeri's cuts)
  - Any unfeasible solution in the subproblem will give a valid cut (can use IPM)

Reduction of master Benders cuts (i)

- Extracting non-dominated cuts / Pareto-optimal cuts
  - A cut or constraint dominates another if any evaluation of first stage decisions is larger than or equal to the previous one
- Removing inactive cuts
  - Dynamically defining the master problem so that only the cuts that are likely to be active constraints are considered

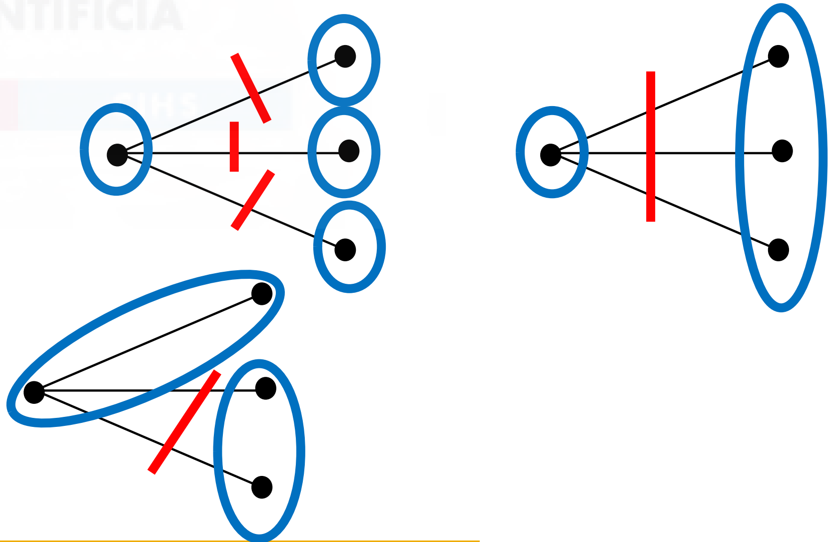
Generation of master Benders cuts (ii)

- Generating covering cuts
  - Generating cuts so that they carry the maximum amount of information possible
  - They include the maximum possible number of 1^{st} stage variables
- Minimal Infeasible Subsystems (MIS) can be used to modify the way feasibility cuts are calculated
  - Instead of minimizing the sum of infeasibilities the problem minimizes the number of equations that are infeasible
  - This enables faster convergence in some cases
  - Conversely, if most of the solutions are infeasible, it is possible to keep a maximum feasible set to derive optimality cuts to better guide the search

Generation of master Benders cuts (ii)

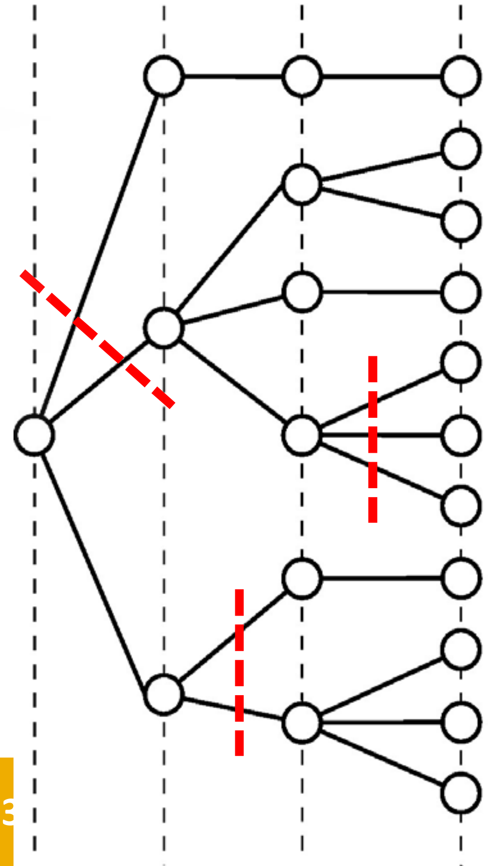
- Type of cut formulation
  - Linear or nonlinear type (linearization around a point)

- Partitioning the scenario tree for subproblem solution
  – Cut aggregation: monocut or multicut
    - More cuts ⇒ more information to the master ⇒ less iterations. More variables ⇒ more constraints in the master
  – Divide the scenario tree for reducing total solution time, dealing with:
    - Decide size of subproblems
      - 3 small subproblems vs. 1 large subproblem vs. 2 medium subproblem
    - Number of cuts
    - Number of Benders iterations



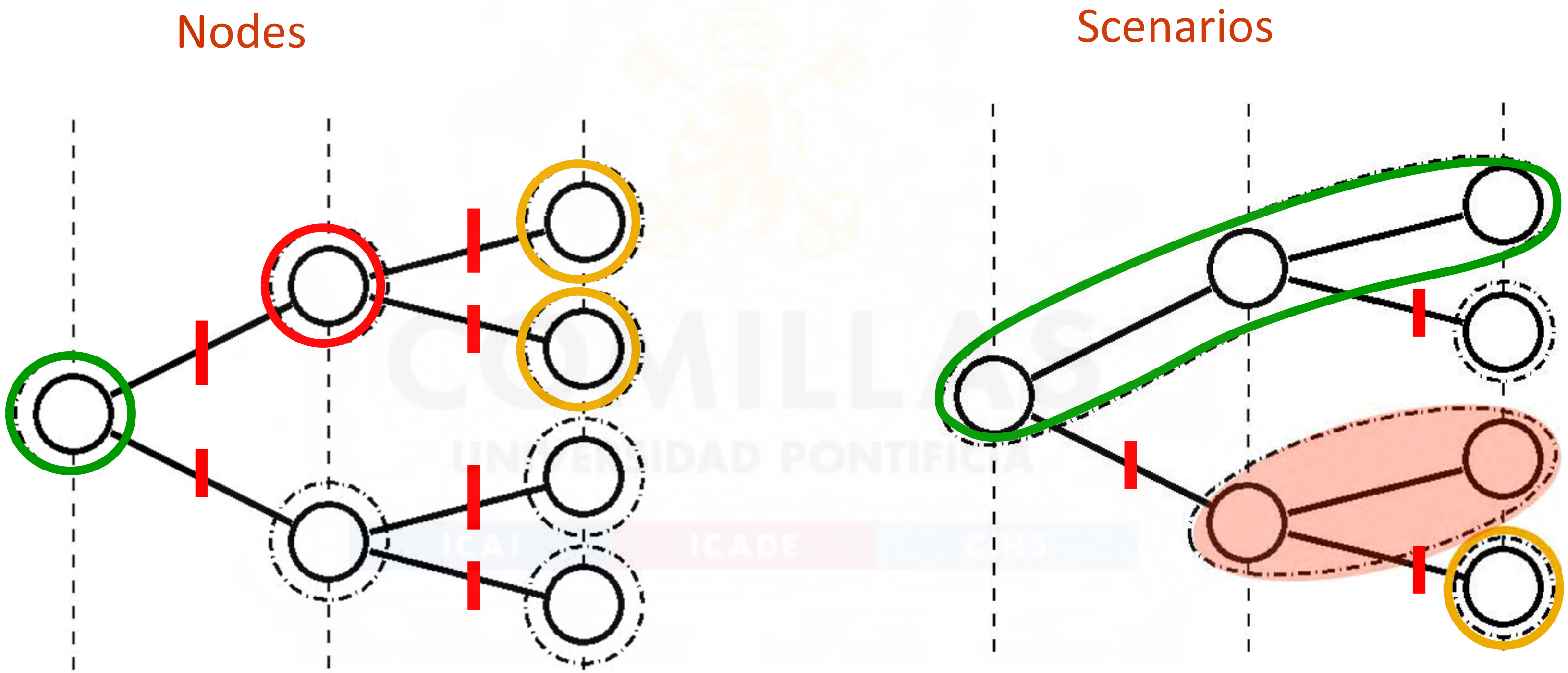
Scenario tree for subproblem solution (i)

- Tree partition or node aggregation (multicoordination)
  - Decompose as less as possible
    - Subtrees defined by the tree breaks
  - Advantage: reduction in decomposition algorithm iterations
  - Disadvantage: potential increase in problem solution time (interior point method)
  - Methods
    - By nodes
    - By scenarios
    - By subtrees
    - By complete scenarios
    - By graph partition



Scenario tree for subproblem solution (ii)

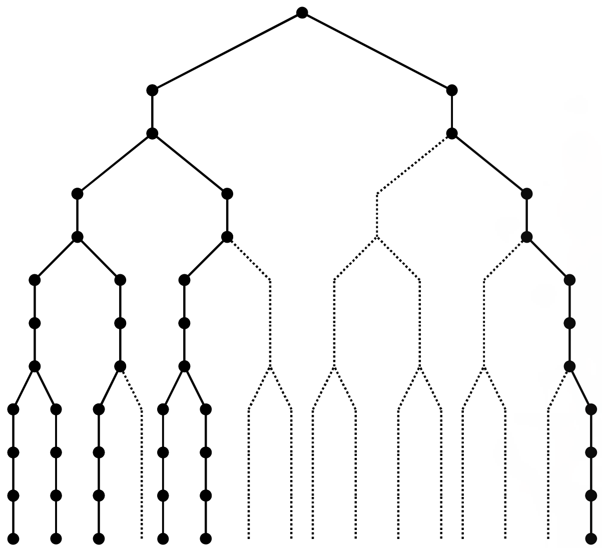
- **Node** and **scenario** partition



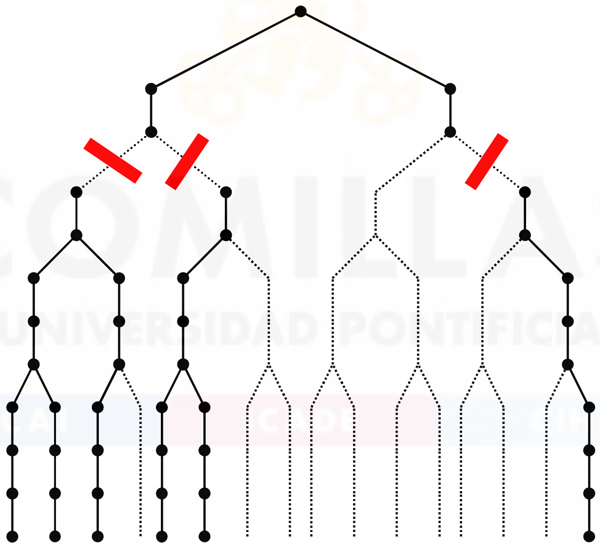
Scenario tree for subproblem solution (iii)

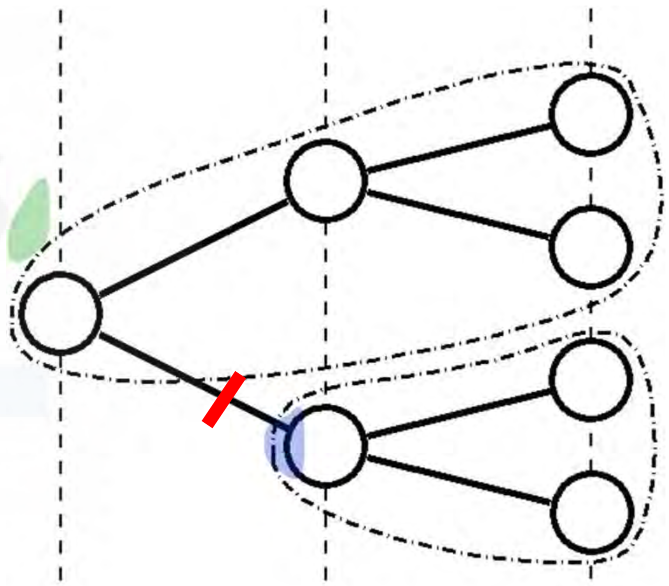
- Subtree partition
  - Ascendant node aggregation (from the leaves to the root) is the one with the best performance

Original tree (asymmetric)



Subtrees from leaves to root





S. Cerisola, A. Ramos *Node Aggregation in Stochastic Nested Benders decomposition Applied to Hydrothermal Coordination* 6th International Conference on Probabilistic Methods Applied to Power Systems (PMAPS) RUM-102. Madeira, Portugal September 2000

Scenario tree for subproblem solution for two-stage problem (iv)

- Subtree partition
  - Two types of scenarios: wind scenarios and system state scenarios. Second-stage scenarios can be arranged differently
  - In general, the most efficient arrangement cannot be known beforehand (tradeoff between the accuracy of the cuts and solution time for the master problem)

(a) Benders' scenario decomposition by both wind scenarios and system states

(b) Benders' scenario decomposition by system states

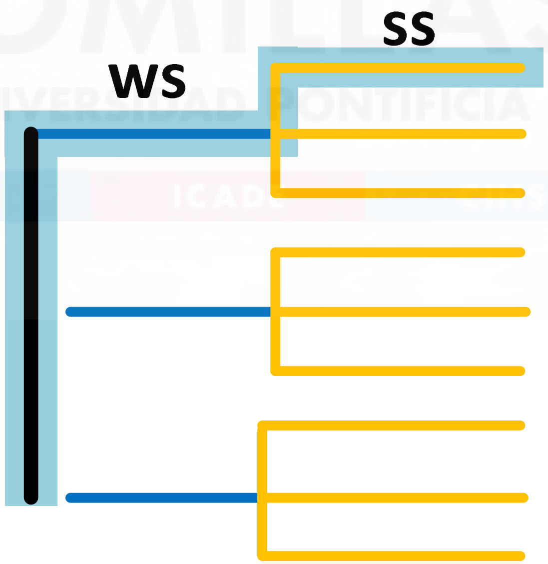
(c) Benders' scenario decomposition by wind scenarios

ws: wind scenario

ss: system state

Scenario tree for subproblem solution for two-stage problem (v)

- Scenario aggregation: the second-stage scenario corresponding to the scenarios with the highest impact on the final design are added to the master problem
  - The master problem proposes solutions that are closer to the optimum
  - Convergence speed can be increased
  - In this case, the most probable states are added to the master
    - Base case system state (no failures)
    - High wind



ws: wind scenario
ss: system state

Scenario tree traversing strategies

- Ways to traverse through the tree from the root to the leaves. The "best" tree traversing strategy will properly balance the quality of the cuts (and hence the lower bound) with the computational effort required to generate them
  - Fast-pass: from 1 to P and from P-1 to 1
  - Shuffle: solves the stage with largest error between lower and upper bounds. It is centered in final stages, never goes backward until the error of a stage is bounded (*fast-forward*)
  - Cautious: goes forward when the error in a stage is small enough. It is centered in initial stages, never goes forward until the error of a stage is bounded (*fast-backward*)

D.P. Morton *An enhanced decomposition algorithm for multistage stochastic hydroelectric scheduling* Annals of Operations Research 64, 211–235. 1996

Benders decomposition in grid computing

- Distributed computing
  - J.M. Latorre, S. Cerisola, A. Ramos, R. Palacios *Analysis of Stochastic Problem Decomposition Algorithms in Computational Grids* Annals of Operations Research 166 (1): 355-373 Feb 2009
  - J.M. Latorre *Resolución distribuida de problemas de Optimización estocástica. Aplicación al problema de coordinación hidrotérmica* Doctoral thesis. Universidad Pontificia Comillas. November 2007

- GAMS grid
  - Use of multiple cores of a computer

Benders decomposition algorithm extensions

- To integer subproblems:
  - Decomposition for multistage problems with integer variables in each stage. Convex hull calculation
    - S. Cerisola, J.M. Latorre, A. Ramos *Stochastic Dual Dynamic Programming Applied to Nonconvex Hydrothermal Models* European Journal of Operational Research 218 (2012) 687–697 10.1016/j.ejor.2011.11.040
    - S. Cerisola, A. Baillo, J.M. Fernandez-Lopez, A. Ramos, R. Gollmer *Stochastic Power Generation Unit Commitment in Electricity Markets: A Novel Formulation and A Comparison of Solution Methods* Operations Research 57 (1): 32-46 Jan-Feb 2009
    - S. Cerisola *Benders decomposition for mixed integer problems. Application to a medium-term hydrothermal coordination problem* Doctoral thesis. Universidad Pontificia Comillas. April 2004
    - S. Cerisola, A. Ramos *Benders' decomposition for Mixed-Integer Hydrothermal Problems by Lagrangean Relaxation* 14th Power Systems Computation Conference (PSCC '02) Session 05, Paper 6, Pages 1-8. Seville, Spain June 2002

Further improvements

- Decrease number of subproblem evaluations
  - Simulation in stochastic optimization
- Approximate recourse function by an analytic expression to avoid subproblem evaluation
  - Multivariate nonlinear regression

12

Simulation in stochastic optimization

Why do we need simulation?

- It is used when the number of values of random parameters is too high
- Computation of expectation in the recourse function (multicut) or expectation in the cut terms (monocut)
- Equivalent to integrate or sample in the random parameter hyperspace with known probability density function. A sample is a combination of random parameter values
- Each sample is computationally cumbersome (solving an LP problem)

Types of sampling

- External sampling
  - Samples are taken to reduce the scenario tree size and then we solve the stochastic optimization problem
    - In nested Benders decomposition we take samples in the forward pass (when making proposals to the descendants)

- Internal sampling
  - Samples are taken while we solve the stochastic optimization problem
    - In a two-stage planning problem, the expected value for the second state is substituted by the sample mean of the second stage

Monte Carlo simulation

- If parameters are independent: the joint probability density function if the product of all the probability density functions
- Computation of sample mean, standard error, confidence interval.
- Stop sampling when confidence interval of the sample mean of the second stage objective function lower than a certain threshold.
- Quadratic behavior (multiply by 4 the number of samples to half the confidence interval)
- Events with low probability but large values of the objective function cause large variances. Therefore, many samples are needed

Stages in general Monte Carlo simulation

1. Random variable generation
2. Simulation or parameter sampling
3. (Variance Reduction Techniques)
4. Results collection
5. Stop the Monte Carlo sampling process

Monte Carlo simulation. Assumptions

- Expected cuts are no longer supporting hyperplanes of the convex recourse function, they may intersect
- Assumptions in the sampling process
  - Error in Benders cuts are in the RHS, not in the slopes, and their variance is the same for the objective function
  - Master problem has the same basis independently of the cut RHS
  - Benders cuts of different iterations are statistically independent

Monte Carlo simulation. Simple convergence criterion

- Lower and upper bounds are random parameters
  - Lower upper bound is the upper bound of the smaller mean of all the iterations
  - Upper lower bound is the last lower bound
  - Variance of each bound associated with the objective function of the subproblems
- Convergence criterion for several samples: confidence interval of the difference of bounds contains the 0.
- Confidence interval of the optimal solution defined by the lower limit of interval of the lower bound and the upper limit of interval of the upper bound. It must be smaller than a threshold. If not increase the number of samples

Variance reduction techniques VRT (i)

- Reduce the size of the mean confidence interval without perturbing its value for the same number of samples or, alternatively, achieve the desired precision with lower sampling effort
- Usually, it is impossible to know beforehand which if going to be the variance reduction o even if it is going to be reduced. You must experiment considering the system under analysis.
- The use of VRT can be understood to take advantage from the system information.
- They imply a computational cost to do some preliminary computation or complementary computation during the simulation process

Variance reduction techniques VRT (ii)

- Common random numbers or correlated sampling or comparative simulation or matched pairs
  - Samples are done for different system configurations with the same set of random numbers being used, each number for the same random variable in the different samples

- Antithetic variables
  - The basic idea in to introduce a negative correlation between two consecutive samples
  - It consists in the use of complementary random numbers in two consecutive samples

Variance reduction techniques VRT (iii)

- Control variable
  - The basic idea is to use the results of a simpler model to predict or explain part of the variance of the mean to estimate.
  - A preliminary computation is needed to estimate the mean of the control variable.
  - This computation has to be very quick with respect to the one of the variable to estimate.

Variance reduction techniques VRT (iv)

- Importance sampling
  - Replaces the original random variable by another one with the same mean and lower variance.
  - The sampling probability density function is modified to center it in the interesting zone.
  - Avoids sampling frequent values but not interesting

- Stratified sampling
  - Intuitive idea similar to previous one but in discrete version.
  - Take more samples of the random variables in the interesting zones.
  - Variance is reduced by concentrating the sampling effort in the relevant strata

13

1. General overview
2. Applications in electric systems
3. Two-stage and multistage programming
4. Decomposition techniques
5. Benders decomposition
6. Nested Benders decomposition
7. Dantzig-Wolfe decomposition
8. Lagrangian relaxation
9. Scenario tree
10. Decomposition in two-stage and multistage stochastic programming
11. Improvements in decomposition techniques
12. Simulation in stochastic optimization
13. **Stochastic dual dynamic programming**

Stochastic dual dynamic programming

Stochastic Dual Dynamic Programming SDDP

- Nested Benders decomposition with
  - Forward pass: scenario sampling instead of solving all the scenarios (sample average approximation, SAA)
  - Backward pass: solution of all the nodes of the recombining tree. It approximates for each scenario the recourse function for the sampled values obtained in the forward iteration.
- Therefore, we have stochastic convergence
  - Lower bound is deterministic while upper bound is stochastic
  - Stopping criterion: lower bound enters the confidence interval of the upper bound
    - If the stopping criterion is 1% for a confidence interval with a 95% of confidence level. The algorithm stops with we are sure with a confidence level of 95% that the relative difference between upper and lower bounds is lower than 1%

M. V. F. Pereira, L. M. V. G. Pinto "Multi-stage stochastic optimization applied to energy planning" Mathematical Programming May 1991, 52 (1), 359-375 10.1007/BF01582895

A. Shapiro "Analysis of stochastic dual dynamic programming method" European Journal of Operational Research 209 (2011) 63–72 10.1016/j.ejor.2010.08.007

A. Philpott and Z. Guan, "On the convergence of stochastic dual dynamic programming and related methods," Operations Research Letters, vol. 36, no. 4, pp. 450–455, 2008 10.1016/j.orl.2008.01.013

Stochastic Dual Dynamic Progamming SDDP

Step 0 Set $I_t^{\xi_t} = J_t^{\xi_t} = 0$. Set $\theta_t^{\xi_t} \equiv 0$ at the initial iteration

Step 1 Simulate N scenarios $(h_t^{\xi_t})^n$, $n:1,\dots,N$, $t=1,\dots,T$

Forward pass:

Repeat for $n:1,\dots,N$

Repeat for $t=1,\dots,T$

Solve $(RP_t^{\xi_t})$ with r. hand side value $(h_t^{\xi_t})^n$ and obtain solution $(x_t^{\xi_t})^n$

If $t=1$ obtain lower bound $\underline{z} = v(RP_1^{\xi_1})$

If infeasible: stop forward pass for simulation n

Step 2 Upper bound computation:

Evaluate objective function of the complete (deterministic) problem for each of the primal solutions so far obtained. $\overline{z} = \frac{1}{N}\sum_{t=1}^{T} c_t (x_t^{\xi_t})^n$

Step 3 (stopping rule)

If $\overline{z} - \underline{z} < tol$ stop, $x_1^{\xi_1}$ is optimal solution, else go to Step 4

Step 4 Backward pass

Repeat for $t = T,\dots,1$

Repeat for each node ξ_t of stage t

Repeat for each proposal obtained in forward pass, modifying the right hand side value of subproblem $(RP_t^{\xi_t})$

Solve $(RP_t^{\xi_t})$

If feasible: obtain objective $\theta_t^{\xi_t,i} = v(RP_t^{\xi_t})$ and dual values $\pi_t^{\xi_t,i}$

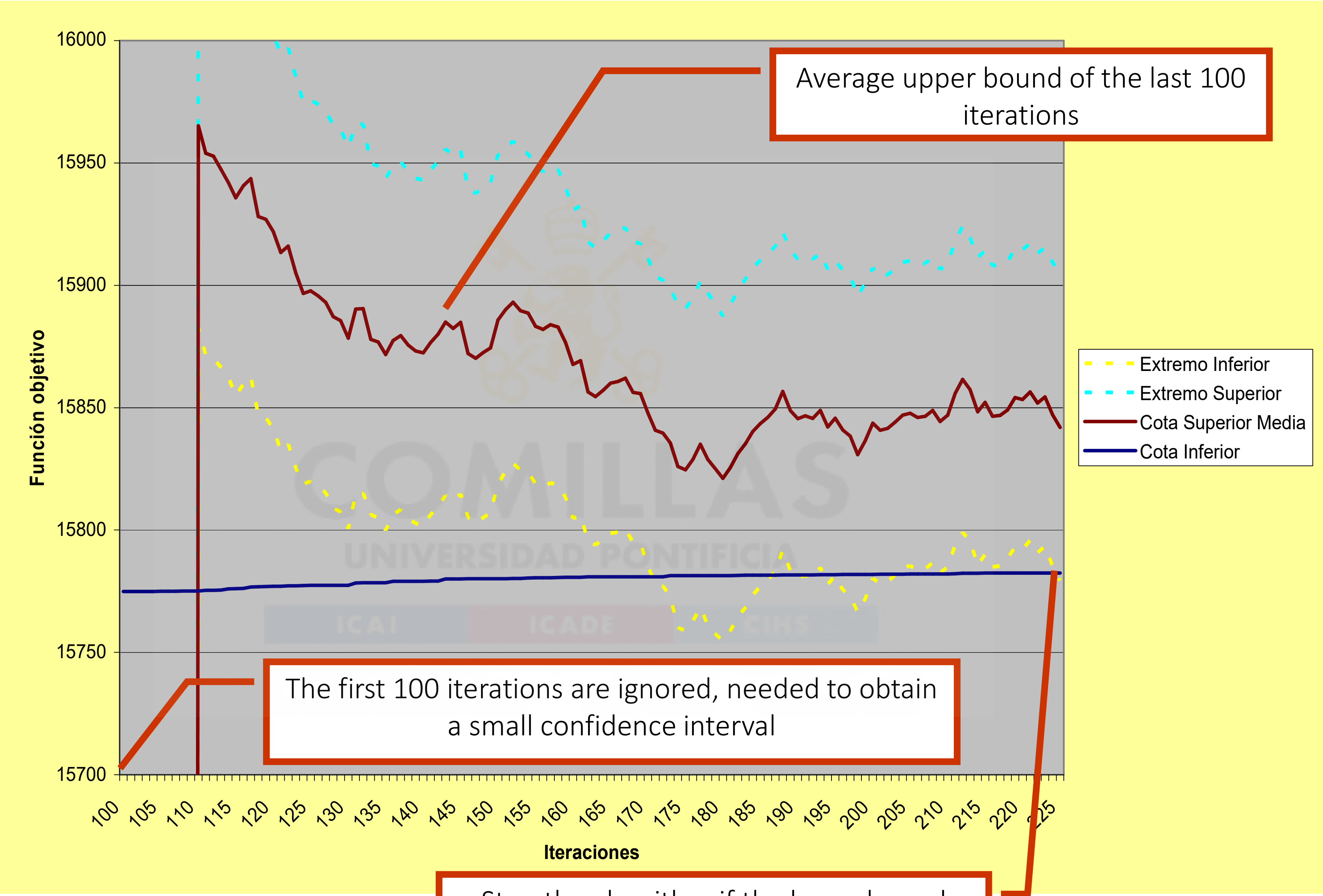
Augment $I_t^{\xi_t} = I_t^{\xi_t} + 1$

If infeasible: obtain sum of infeasibilities $\tilde{\theta}_t^{\xi_t,j}$ and dual values $\tilde{\pi}_t^{\xi_t,j}$

Augment $J_t^{\xi_t} = J_t^{\xi_t} + 1$

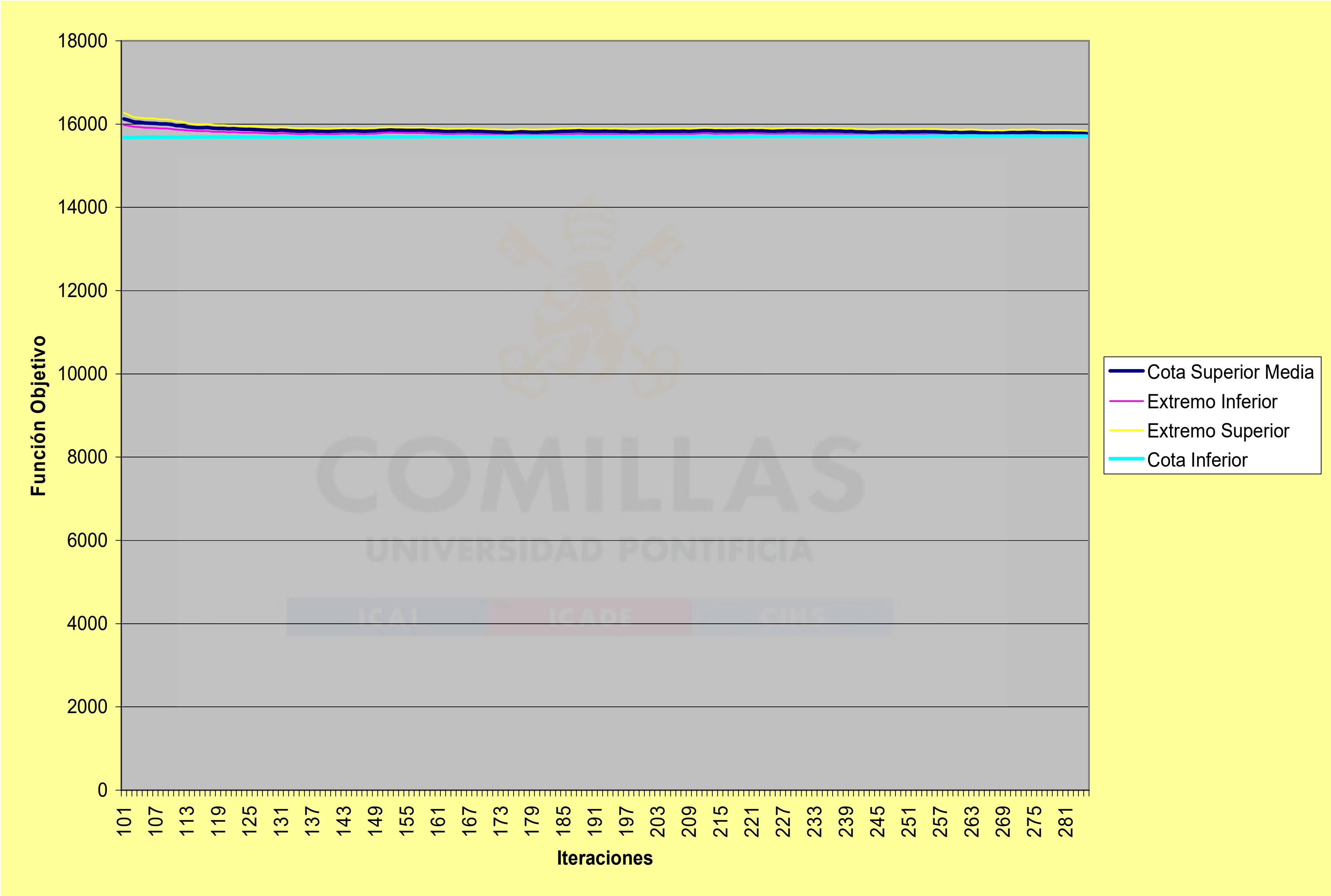
Go to step 1

Stochastic convergence in SDDP (i). Case 1

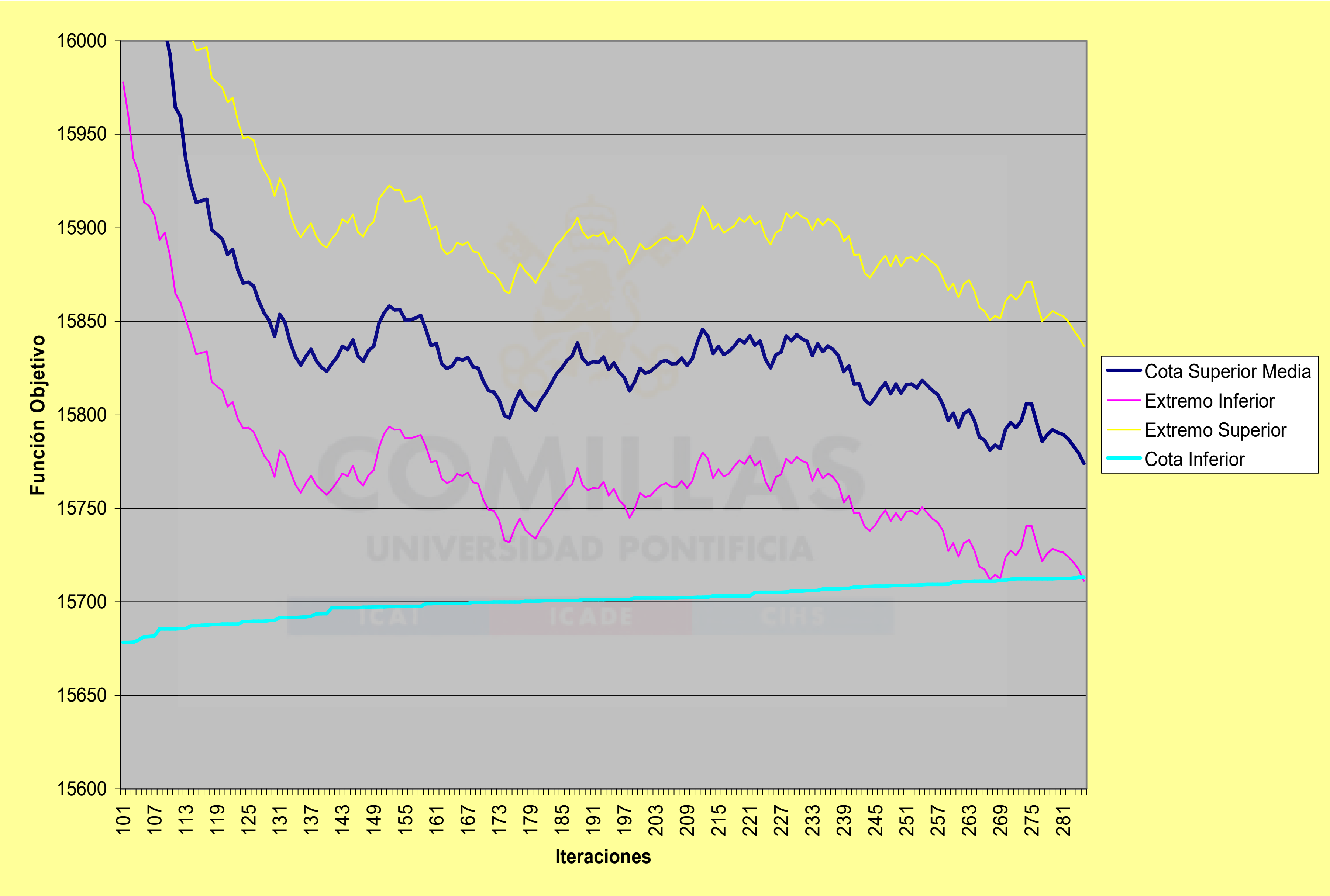


Stop the algorithm if the lower bound enters the confidence interval of the upper bound for a confidence level of 5 %

Stochastic convergence in SDDP (ii)



Stochastic convergence in SDDP (ii). Case 2



Comparison of decomposition methods

Benders	SDDP
Suitable for stochastic problems with tree structure	Suitable for stochastic problems where stochasticity is introduced independently by scenarios
Scenario tree with no fixed structure (symmetrical or nonsymmetrical)	Recombining scenario tree (dependence of one scenario with respect to other is modeled by transition probabilities)
Solution of the deterministic equivalent problem	Impossible to solve the deterministic equivalent problem
Multicut	Multicut
Flexible node aggregation (tree partition)	Rigid node aggregation (tree partition) conditioned by the branching periods
In forward pass all the scenarios are solved	In forward pass only one scenario is solved
Deterministic stopping criterion	Stochastic stopping criterion
Exponential time increase with number of scenarios	Linear time increase with number of scenarios

Prof. Andres Ramos
https://www.iit.comillas.edu/aramos/
Andres.Ramos@comillas.edu
arght@mit.edu