How the European day-ahead electricity market works

ELEC0018-1 - Marché de l'énergie - Pr. D. Ernst

Bertrand Cornélusse, Ph.D.

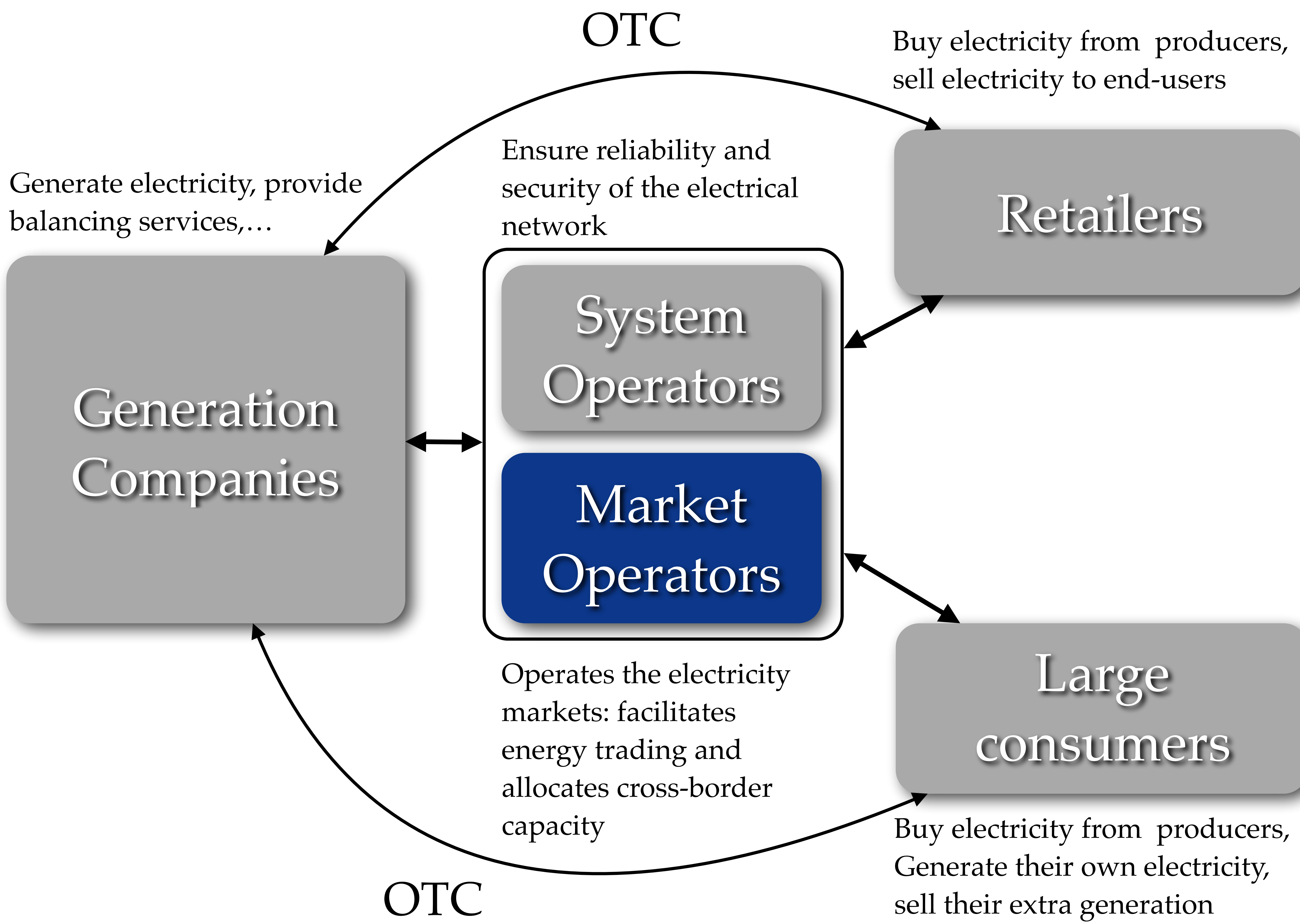
bertrand.cornelusse@ulg.ac.be

March 2017

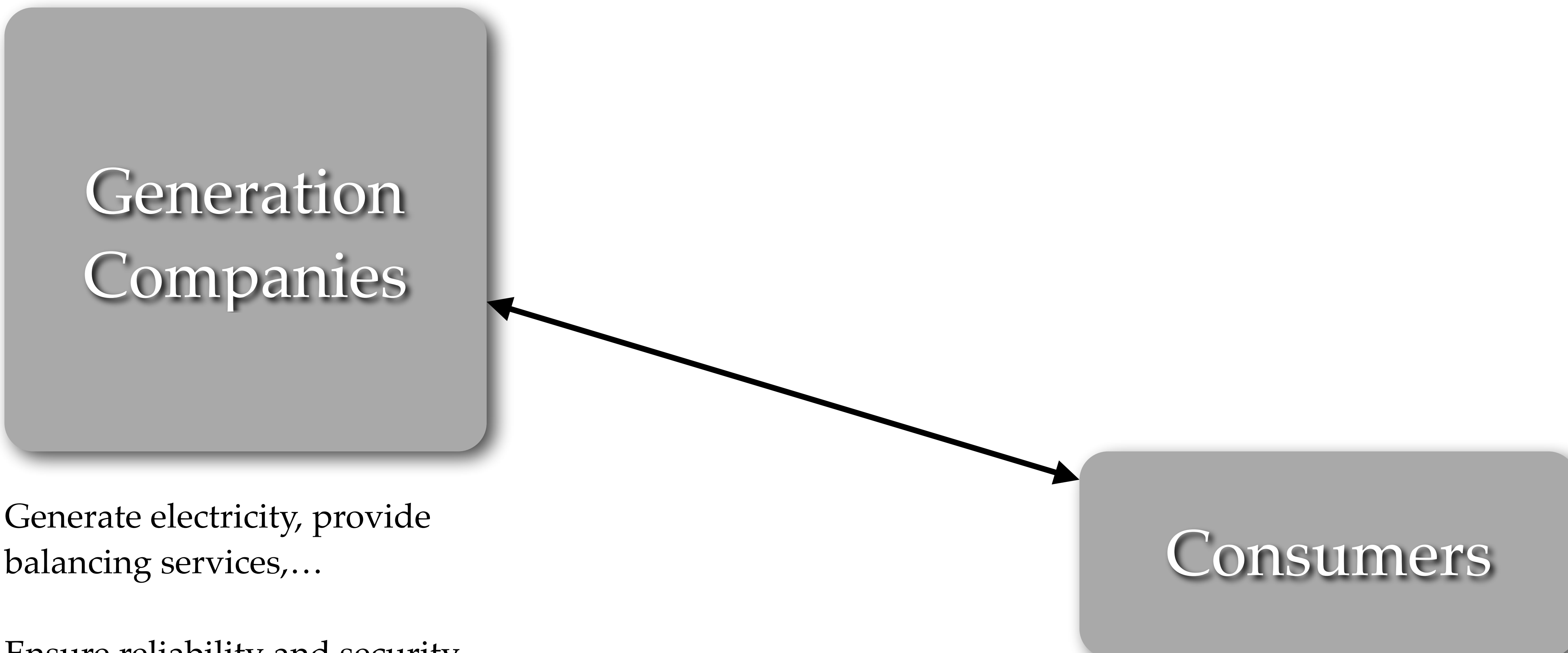
Starting question

How is electrical energy traded in Europe?

Organization of the electrical power system (simplified)



Previous organization (simplified)



Generate electricity, provide balancing services,…

Ensure reliability and security of the electrical network

Sell electricity to end-users

Why an Electricity Market?

Break monopolies and open electricity generation and retail to competition.

Facilitate exchanges between countries. This does not only require creating a markets in each country, but also coupling those markets.

Give incentives for building capacity / consuming electricity where appropriate. Market results provide time series of prices per area and period of the day, volumes that are exchanged, etc. This is part of necessary information to determine the appropriate capacity investments (at least in principle…).

What is an Electricity Market?

A centralized platform where participants can exchange electricity transparently according to the price they are will to pay or receive, and according to the capacity of the electrical network.

Fixed gate auction

- Participants submit sell or buy orders for several areas, several hours,
- the submissions are closed at a pre-specified time (closure)
- the market is cleared. Uniform clearing price: market prices are (well, ***should be …)*** sufficient to determine whether orders are accepted or rejected
- Example: day-ahead. This is the topic of this lecture.

Continuous time auction

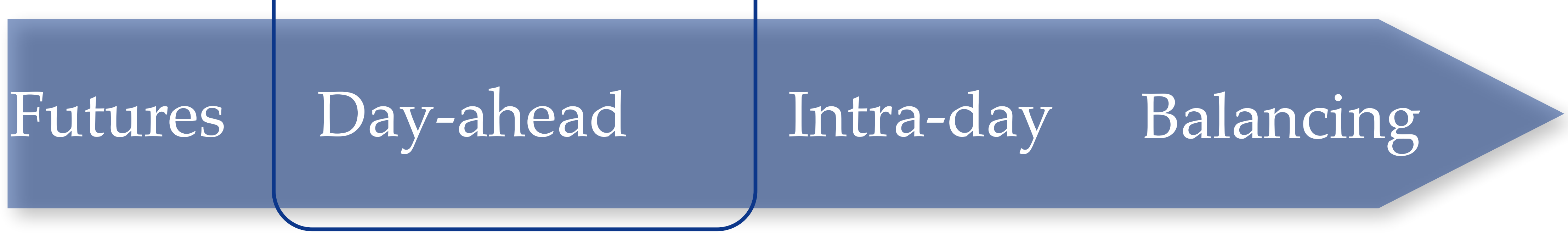
- Participants continuously submit orders. Orders are stored,
- Each time a deal is feasible, it is executed,
- Example: intra-day.

Markets by time horizon and activity

Towards real-time

Markets by time horizon and activity

- Generation companies / retailers submit supply / demand orders
- TSOs allocate cross-border capacity
- Cleared once per day around 1PM



Towards real-time

Day-ahead market operation and coupling

Each zone (or **bidding area**) has its own **Power eXchange (PX)** which collects participants orders.

France and Germany : EPEX spot, Belgium and the Netherlands: APX-Endex, etc.

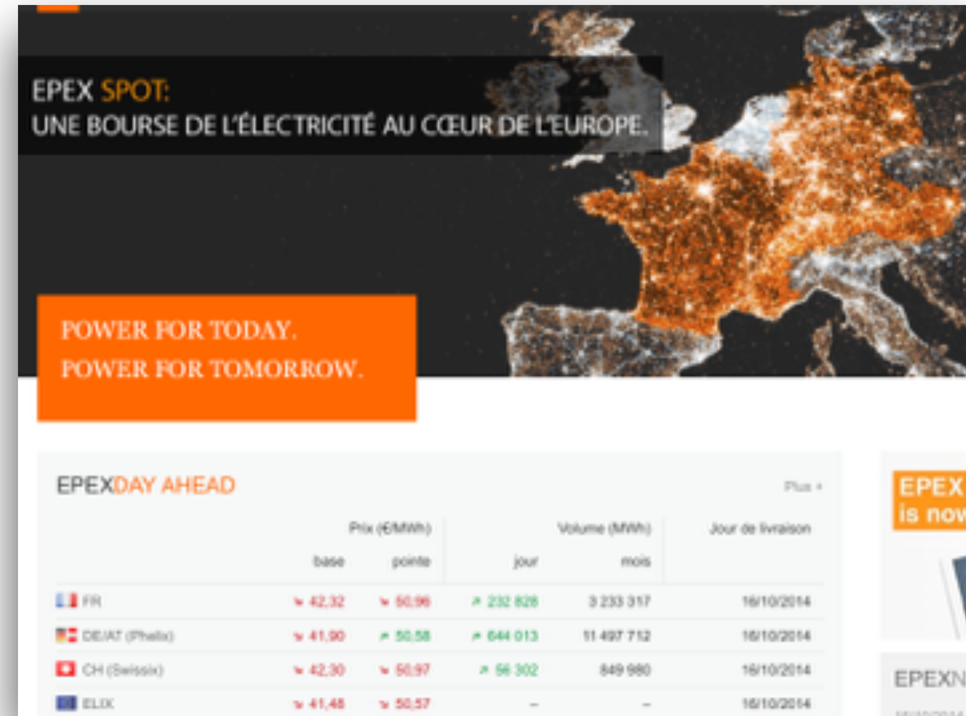
Market coupling

- Perform clearing once per day for all coupled zones
  - hence orders can be matched between markets
  - and cross-border capacity is thus implicitly allocated.
- A price difference between countries must be explained by the congestion of some transmission lines.

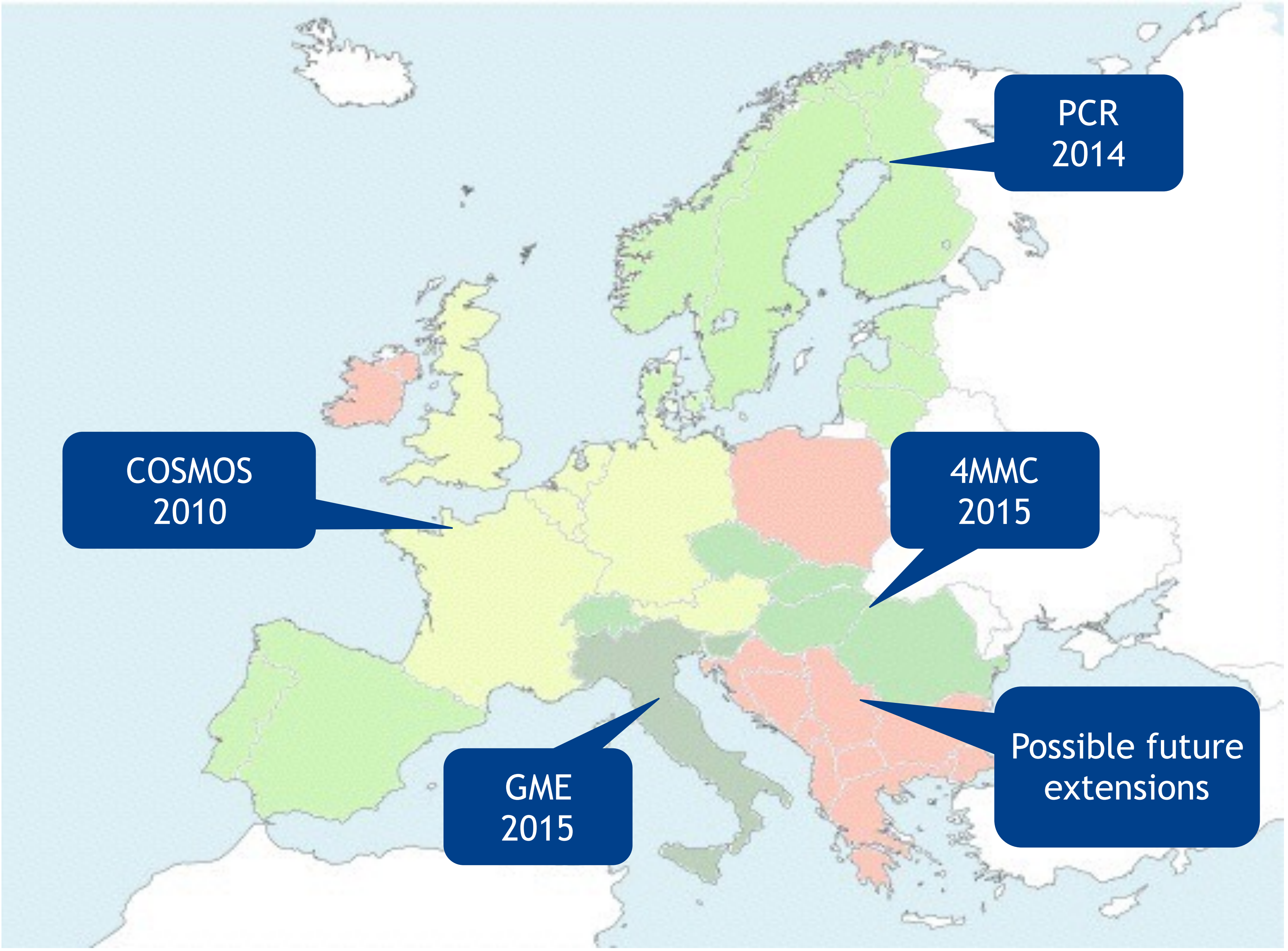
Note: before market coupling, cross-border (i.e. inter-market) capacities were allocated through explicit auctions before deals were actually performed.

Detail: in general, each PX is responsible for the allocation to its participants (**portfolio allocation**). It has its own tie rules to lift indeterminacies.

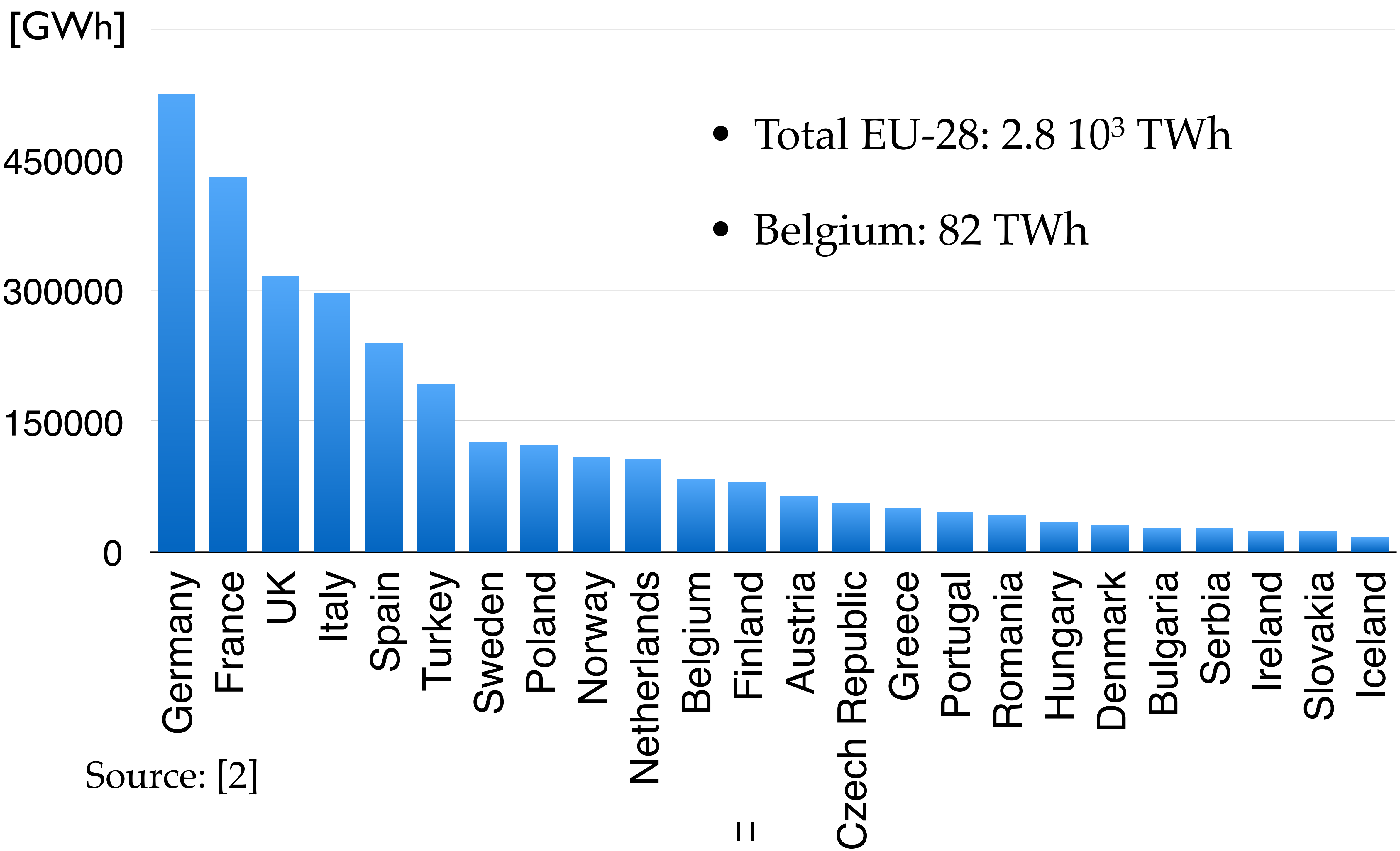
Market operators / Power exchanges



History of the coupling project



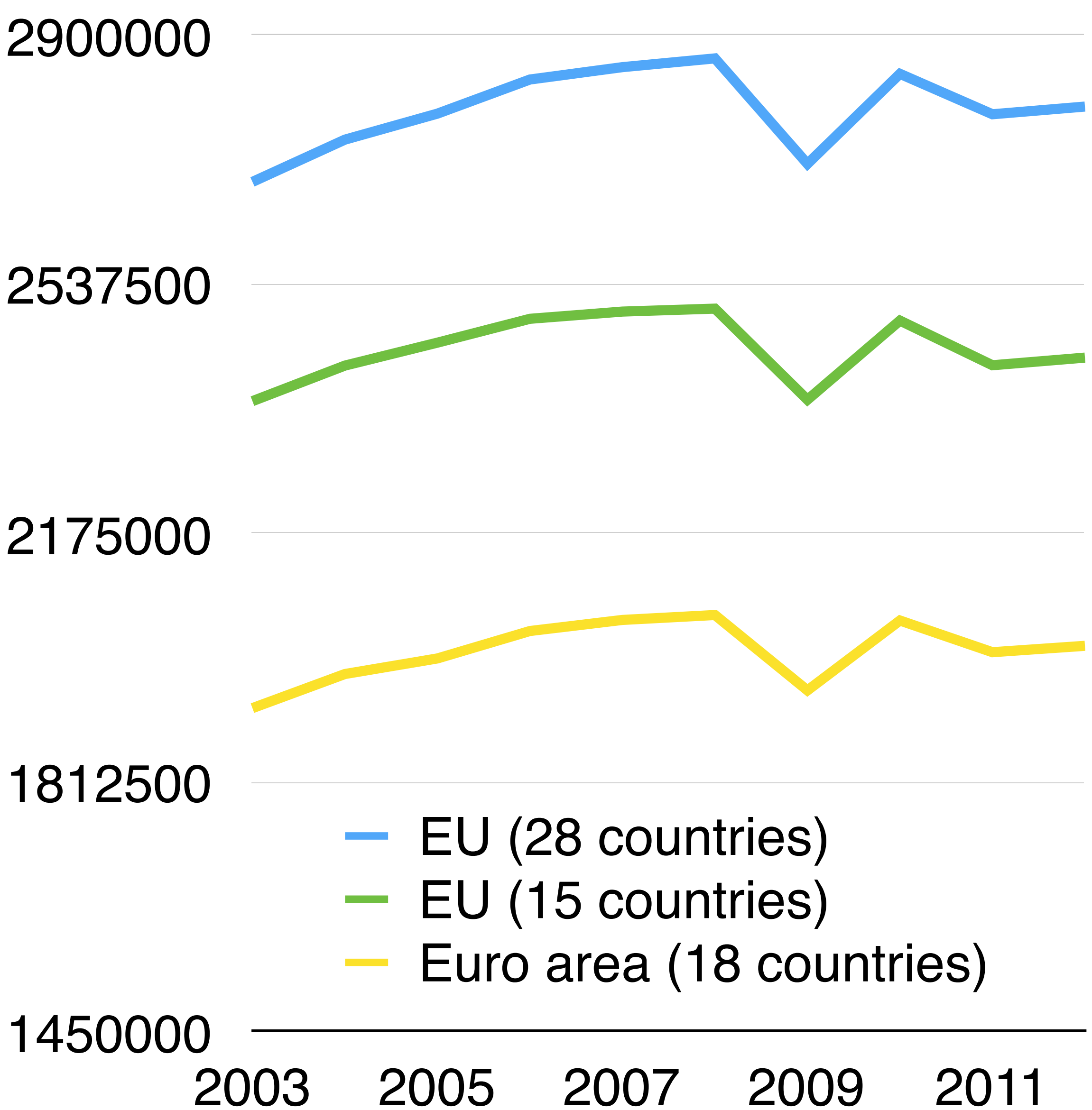
Quantitative insight: Yearly consumption (2012)

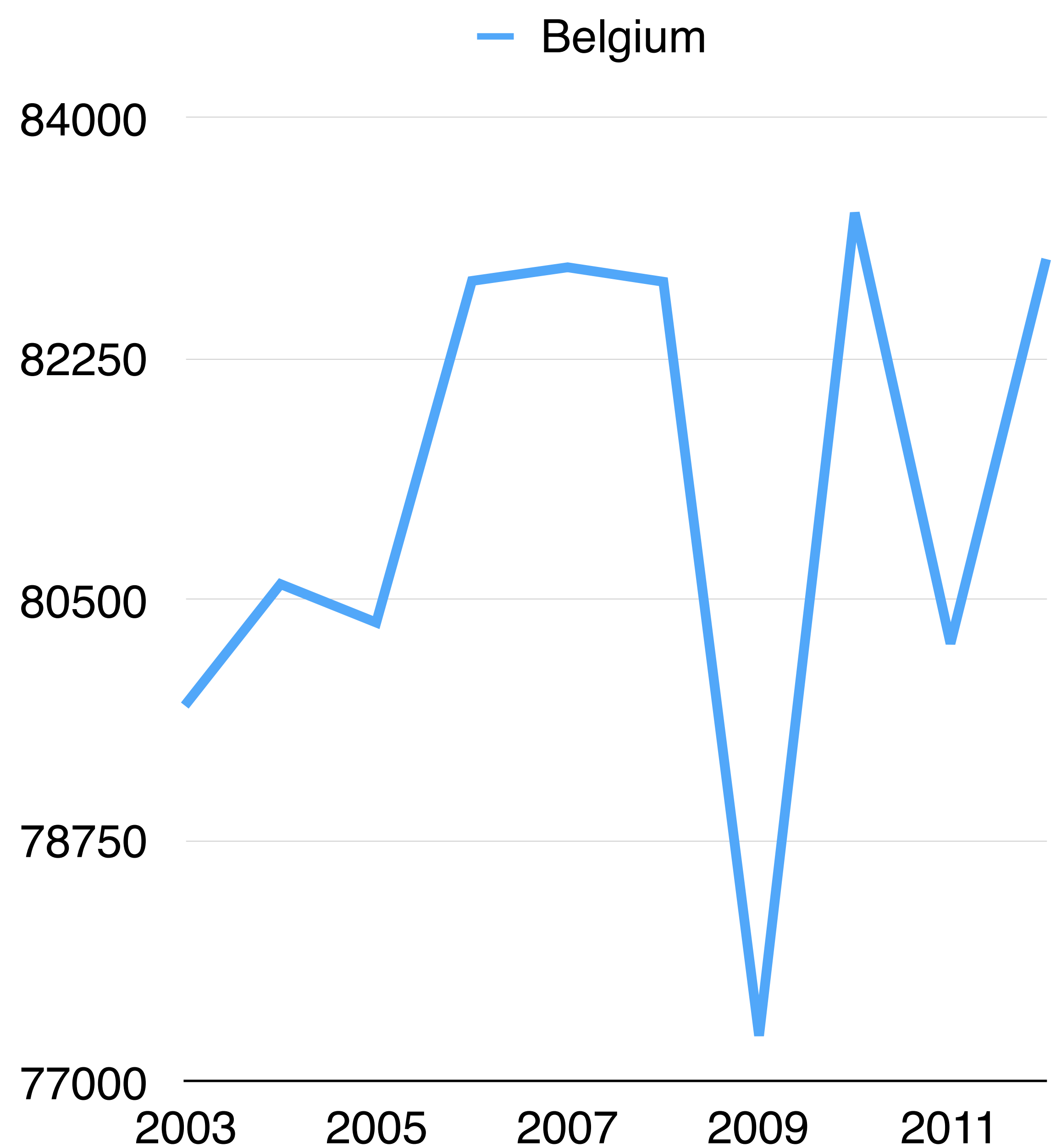


- Total EU-28: 2.8 10^3 TWh
- Belgium: 82 TWh

Source: [2]

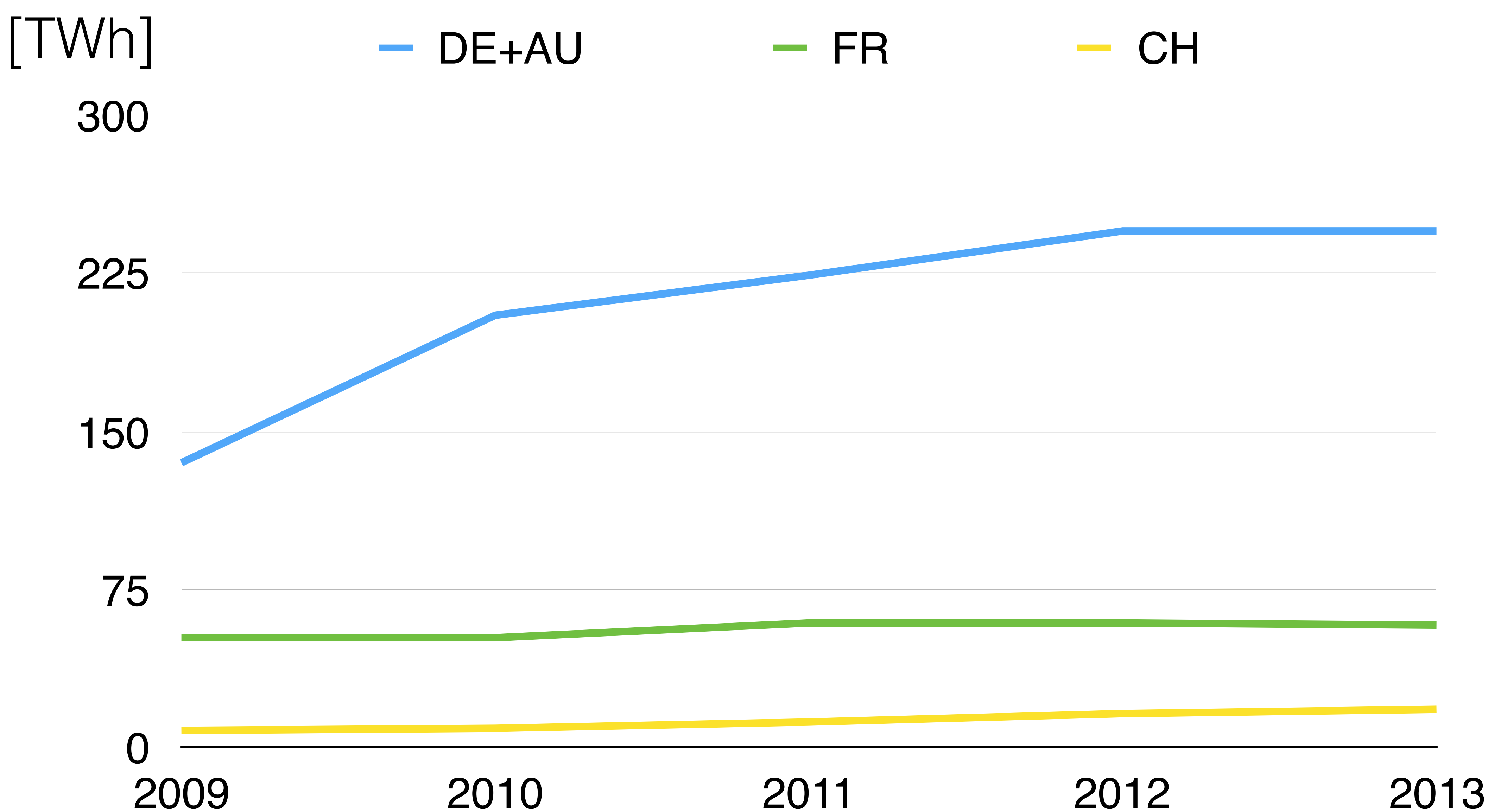
Evolution of consumption [GWh]





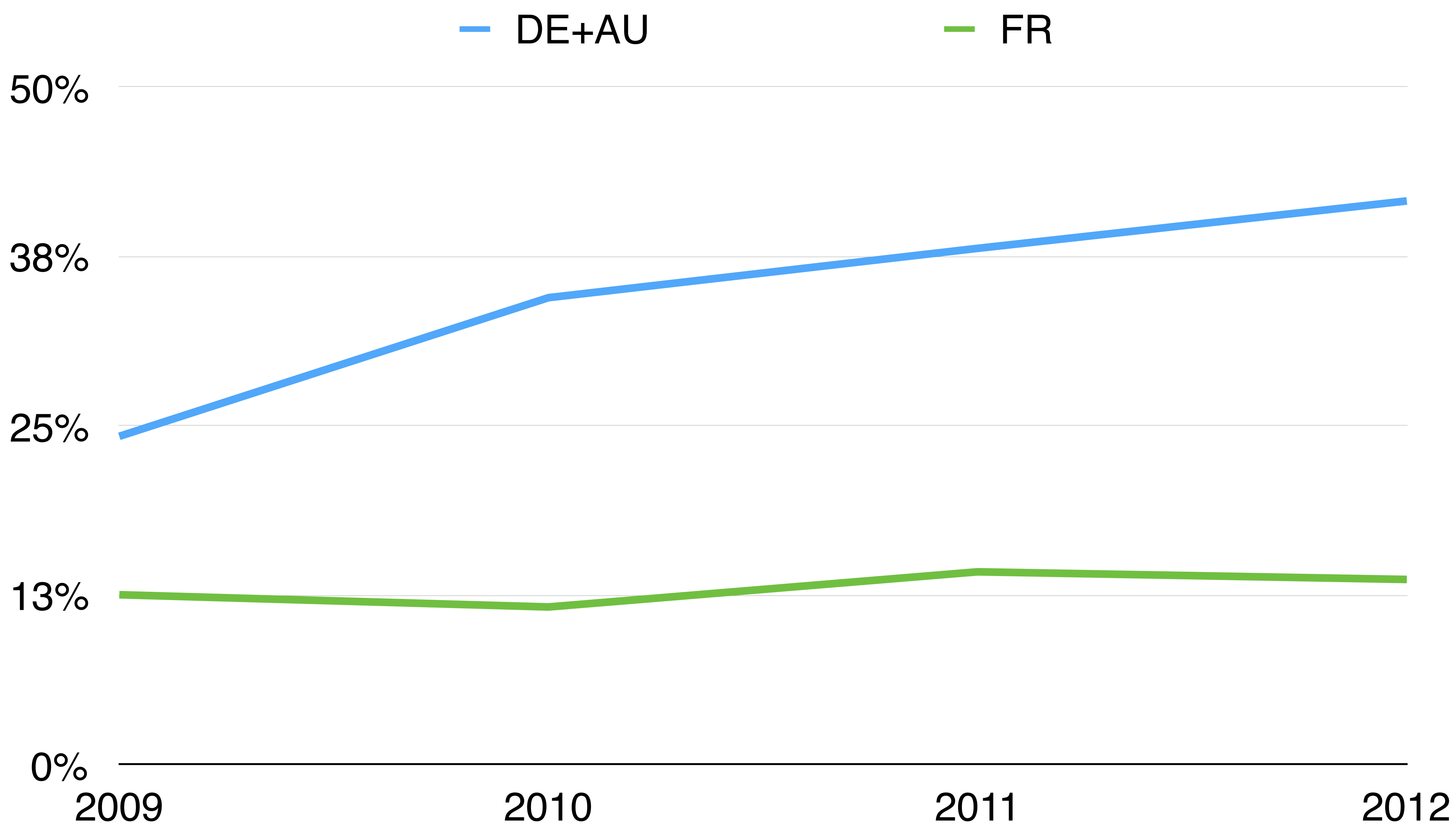
Source: [2]

Energy traded on the day-ahead market (DAM)



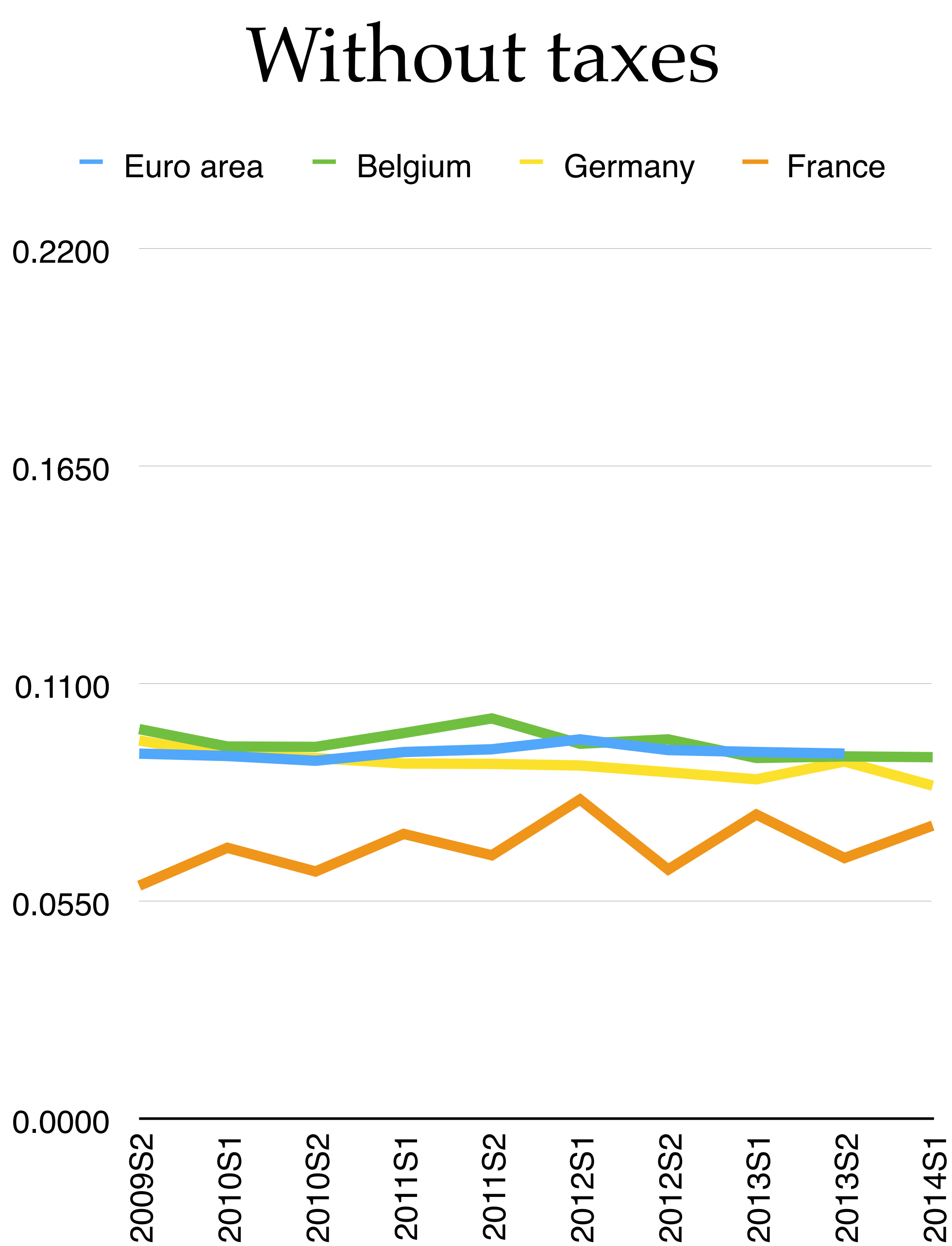
Source: [1]

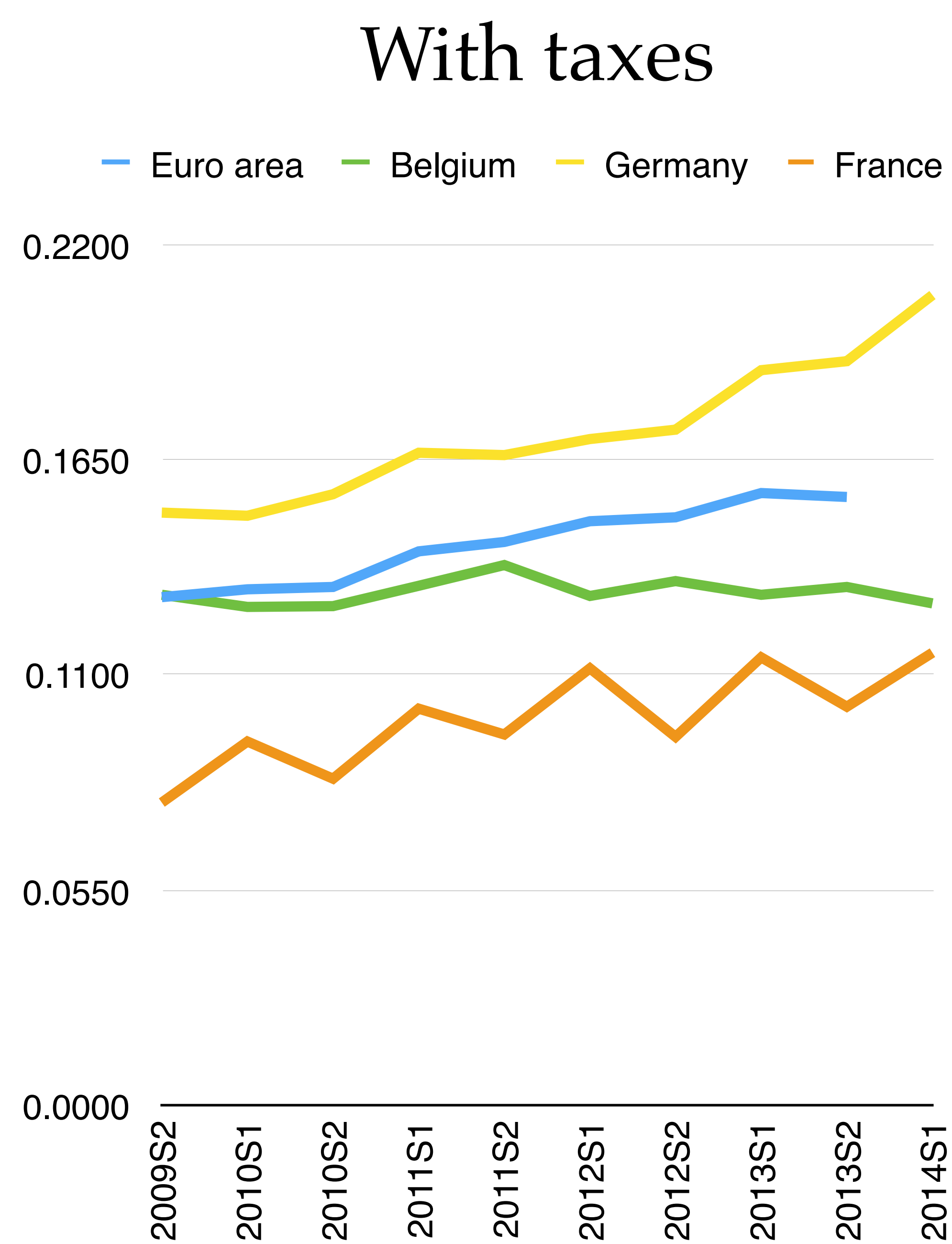
Traded volume in DAM/ total consumption



Source: [1,2]

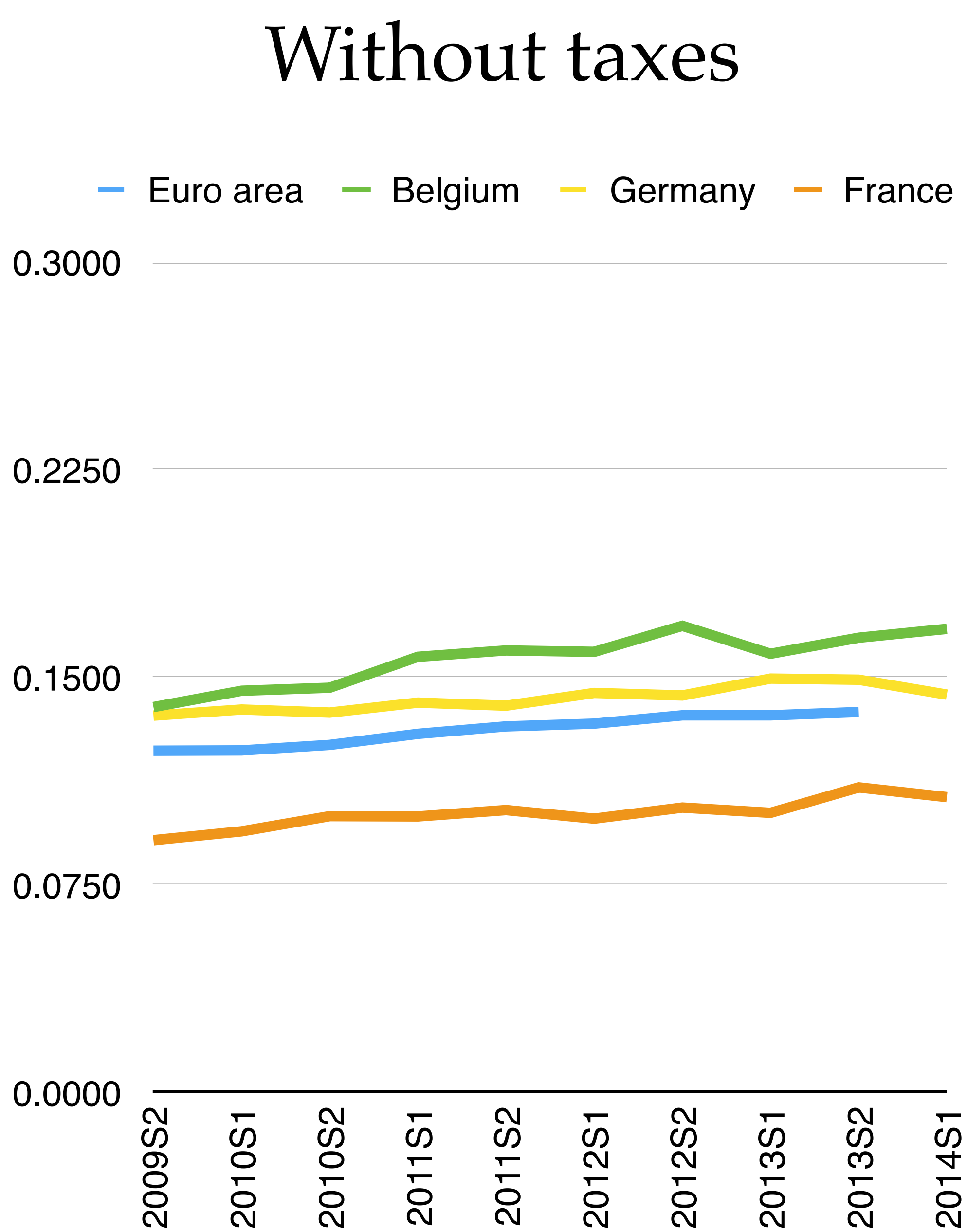
500MWh < Consumption < 2000 MWh

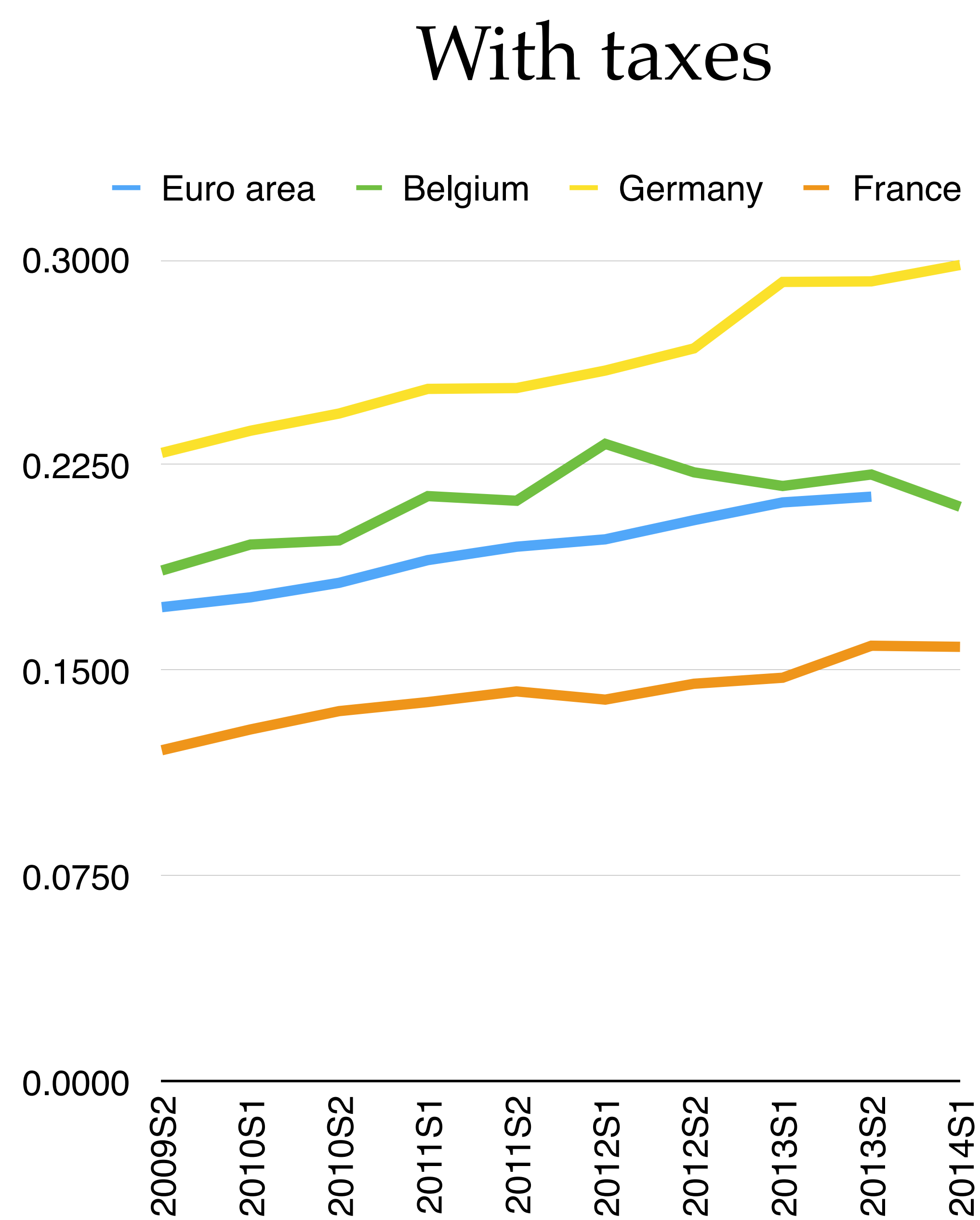




Source: [2]

Electricity prices for household consumers [Euro/kWh]





Source: [2]

Outline of the lecture

1. Definitions and market rules
2. Selected topics in Mathematical Programming
3. Formalization of the day-ahead market coupling problem
4. A few words about the solution method implemented in EUPHEMIA

Outline of the lecture

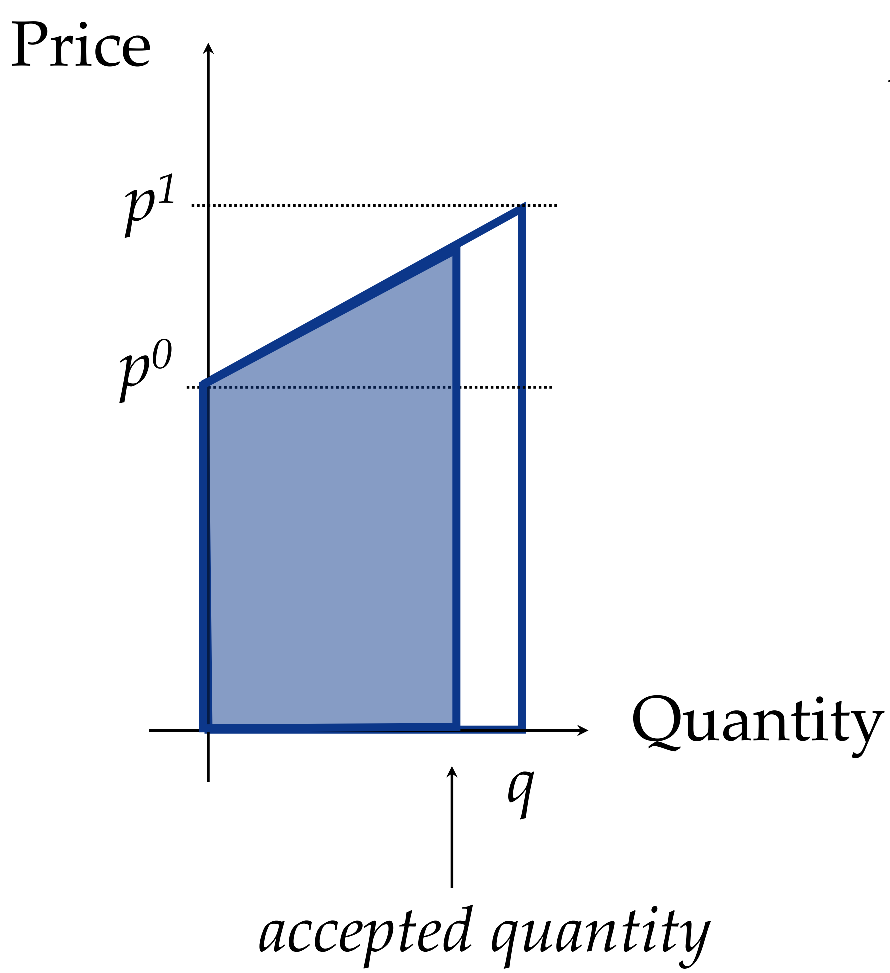
1. **Definitions and market rules**

2. Selected topics in Mathematical Programming

3. Formalization of the day-ahead market coupling problem

4. A few words about the solution method implemented in EUPHEMIA

Orders: expressing the willingness to buy or sell

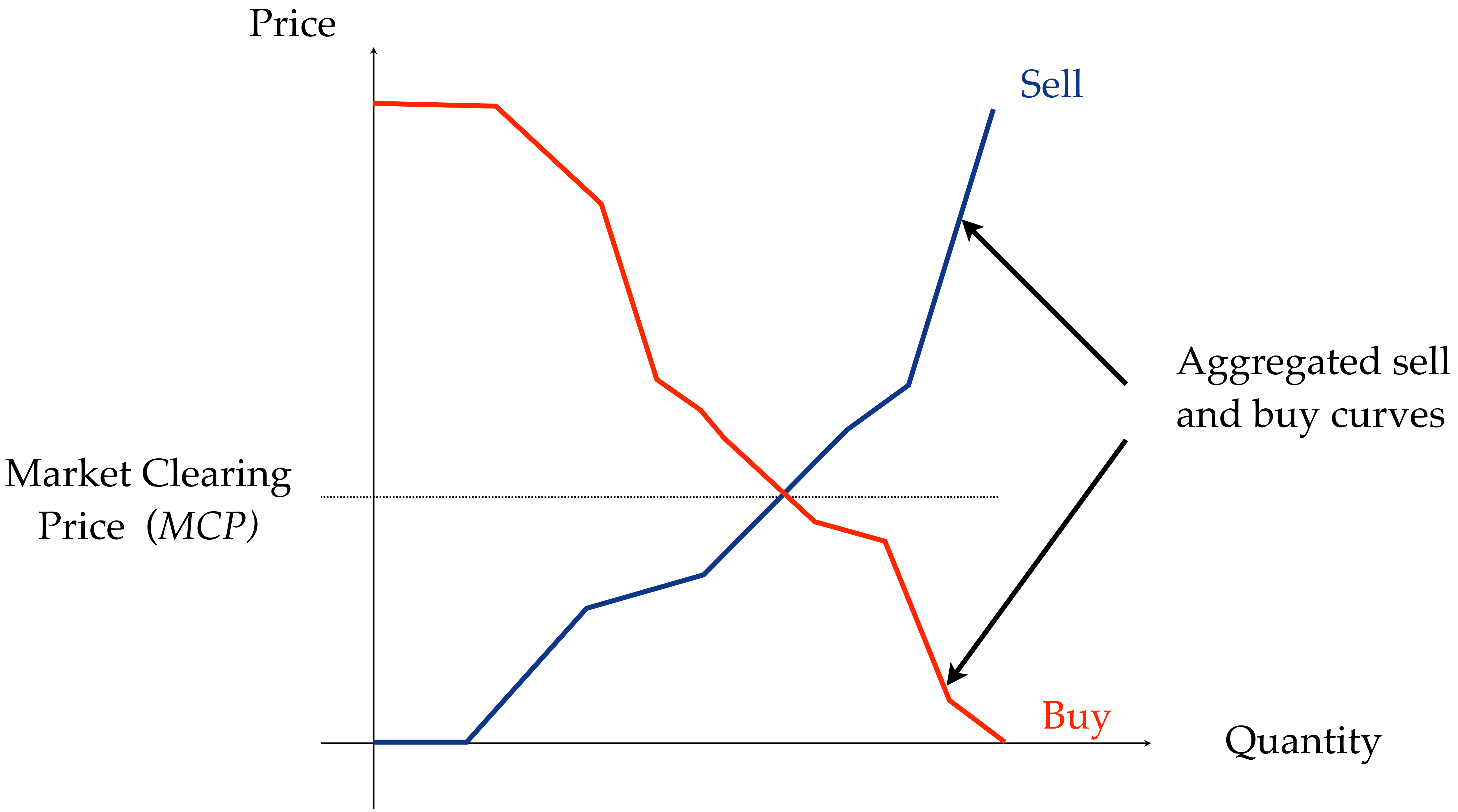


Later on, we will denote the fraction of q accepted by $x \in [0, 1]$

For now, assume

- participants submit orders that can be matched to any proportion
- p^0: price at which the order starts to be accepted
- p^1: price at which the order is totally accepted
- only one period
- the *quantity* is an energy amount, expressed in MWh (MegaWatt X hour)

A single period, single location, day-ahead market



"Clearing" the market amounts to determining which orders should be accepted and at which price

- But what are the properties of the computed prices? Ideally, they should be such that all orders that are
    - **in-the-money** are fully accepted
    - **out-of-the-money** are fully rejected
    - **at-the-money** are accepted
        - at a proportion $\dfrac{MCP - p_0}{p_1 - p_0}$ if $p_0 \neq p_1$
        - at any proportion if $p_0 = p_1$

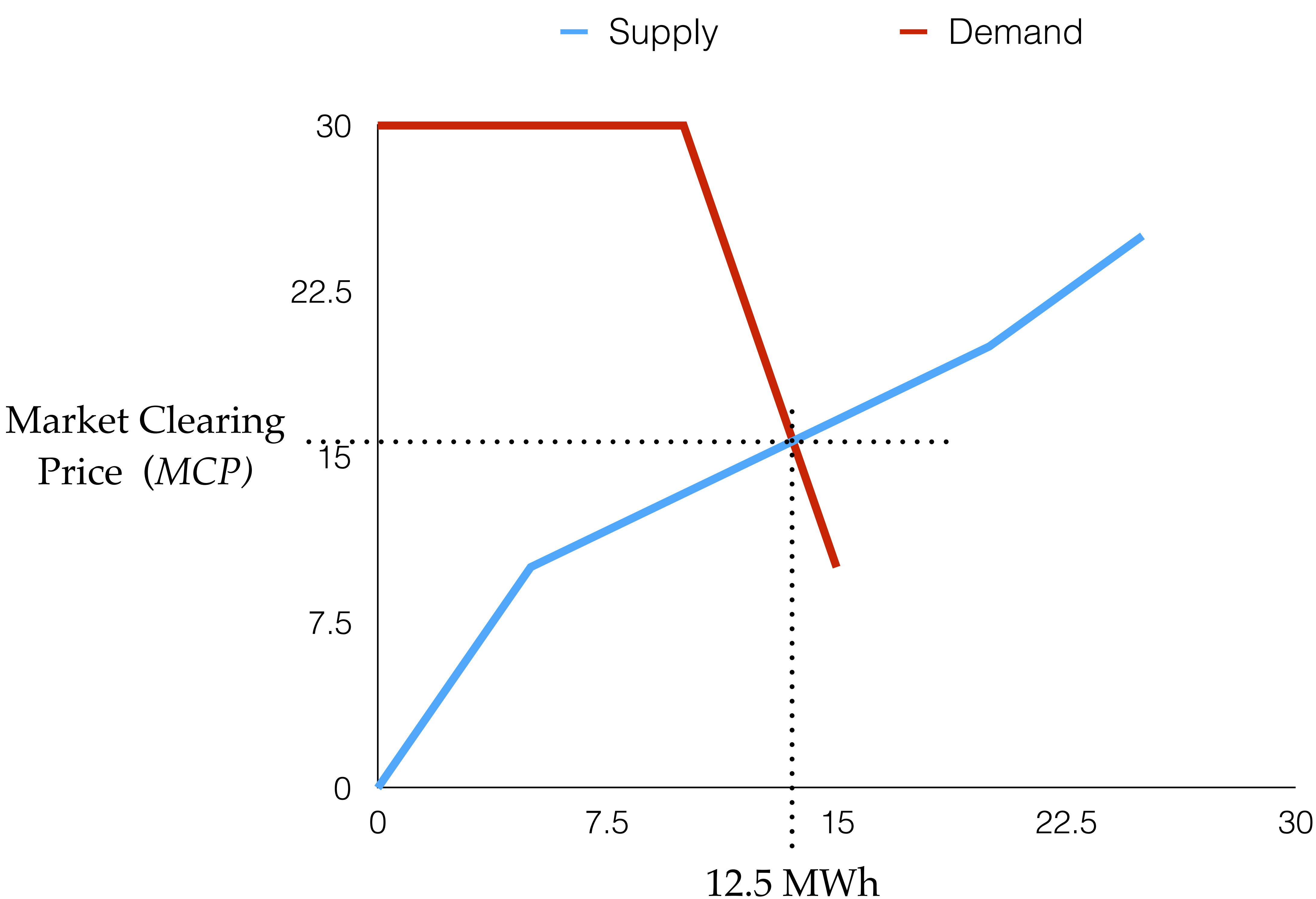
Exercise with hourly order curves

- The order book is composed of

Order	Type	Quantity	p_0	p_1
1	Supply	10	0	20
2	Supply	15	10	25
3	Demand	5	30	10
4	Demand	10	30	30

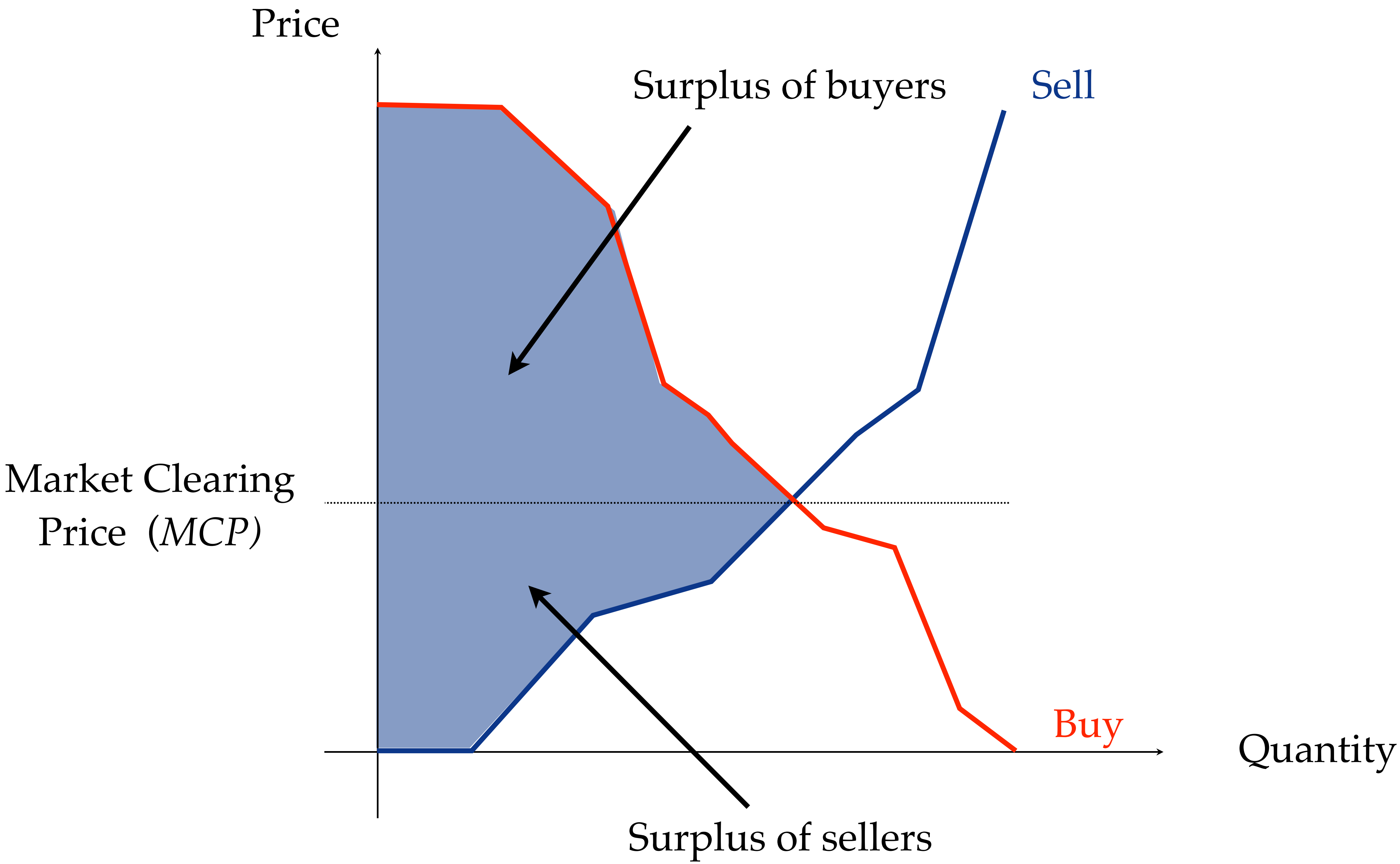
- Determine the supply and demand curves and compute the MCP.

Solution



Supply
Demand
30
22.5
15
7.5
0
Market Clearing
Price (*MCP*)
0
7.5
15
22.5
30
12.5 MWh

Graphical view of welfare (shaded area)



Block orders

A block order is defined by

- one price
- a set of periods
- quantities for those periods
- (a minimum acceptance ratio)

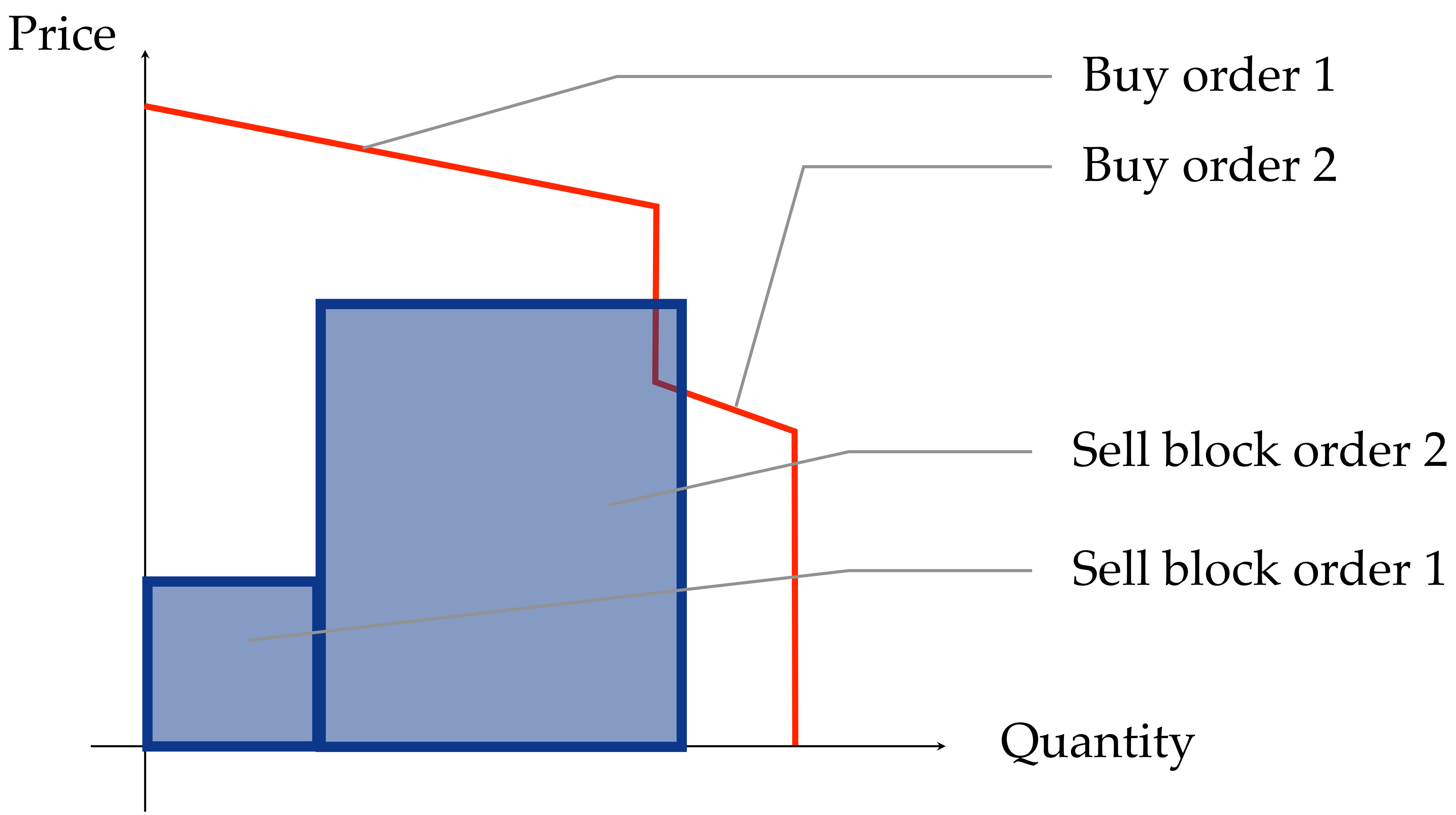
Why?

- Producers want to recover start-up costs, model technical constraints,
- consumers want to secure their base load.

Severe complications

- Couples periods
- Introduces “non-convexities”

Exercise with block orders (a single time period)



Optimizing welfare would lead to the acceptance of the two block orders.
How do you set the price? Which orders are accepted?

The rule in Europe:
no Paradoxically Accepted Block (PAB)

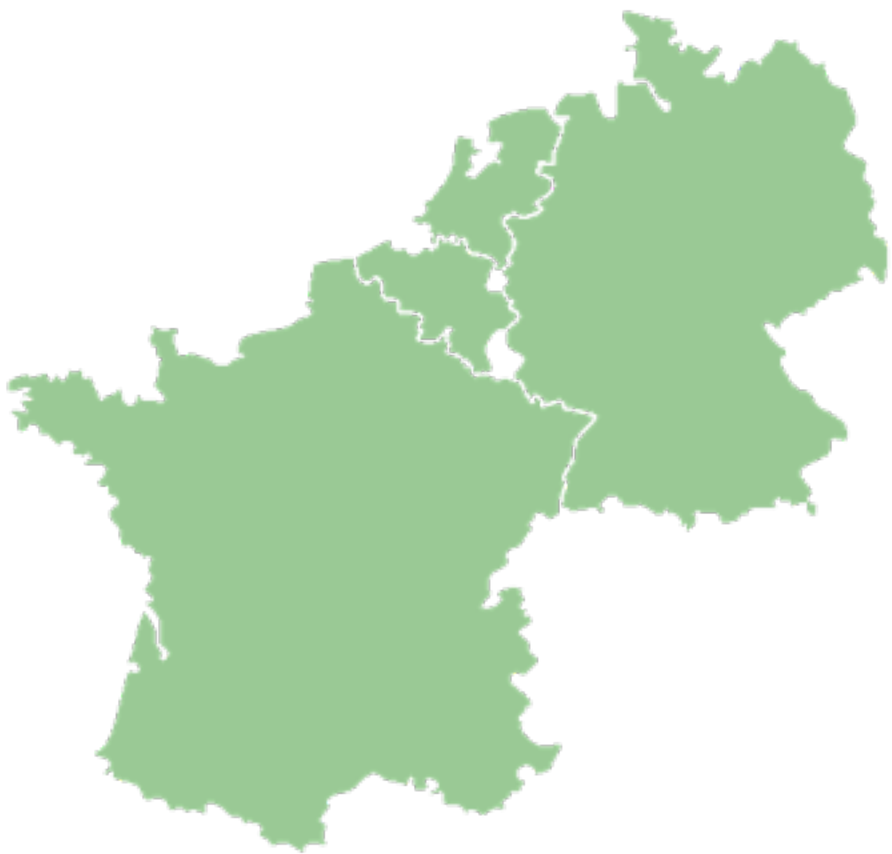
No order can be accepted while loosing money, even if it increases the total welfare.

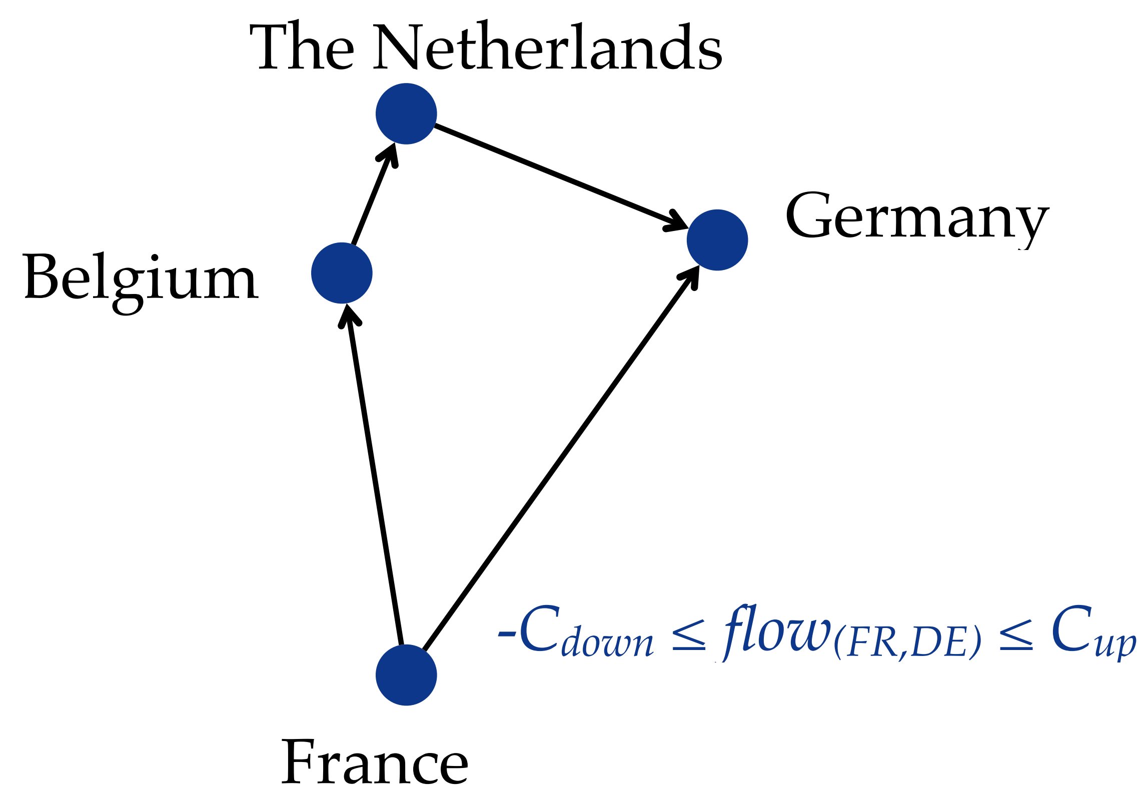
On the other hand a block order could be rejected although it is in the money: **Paradoxically Rejected** orders.

Exercise: create an example where the optimal solution contains a Paradoxically Rejected Block order.

The exchanges between markets are restricted

ATC model: connectors are defined between some pairs of bidding areas. Electricity can be exchanged via these connectors, but exchanges are limited.

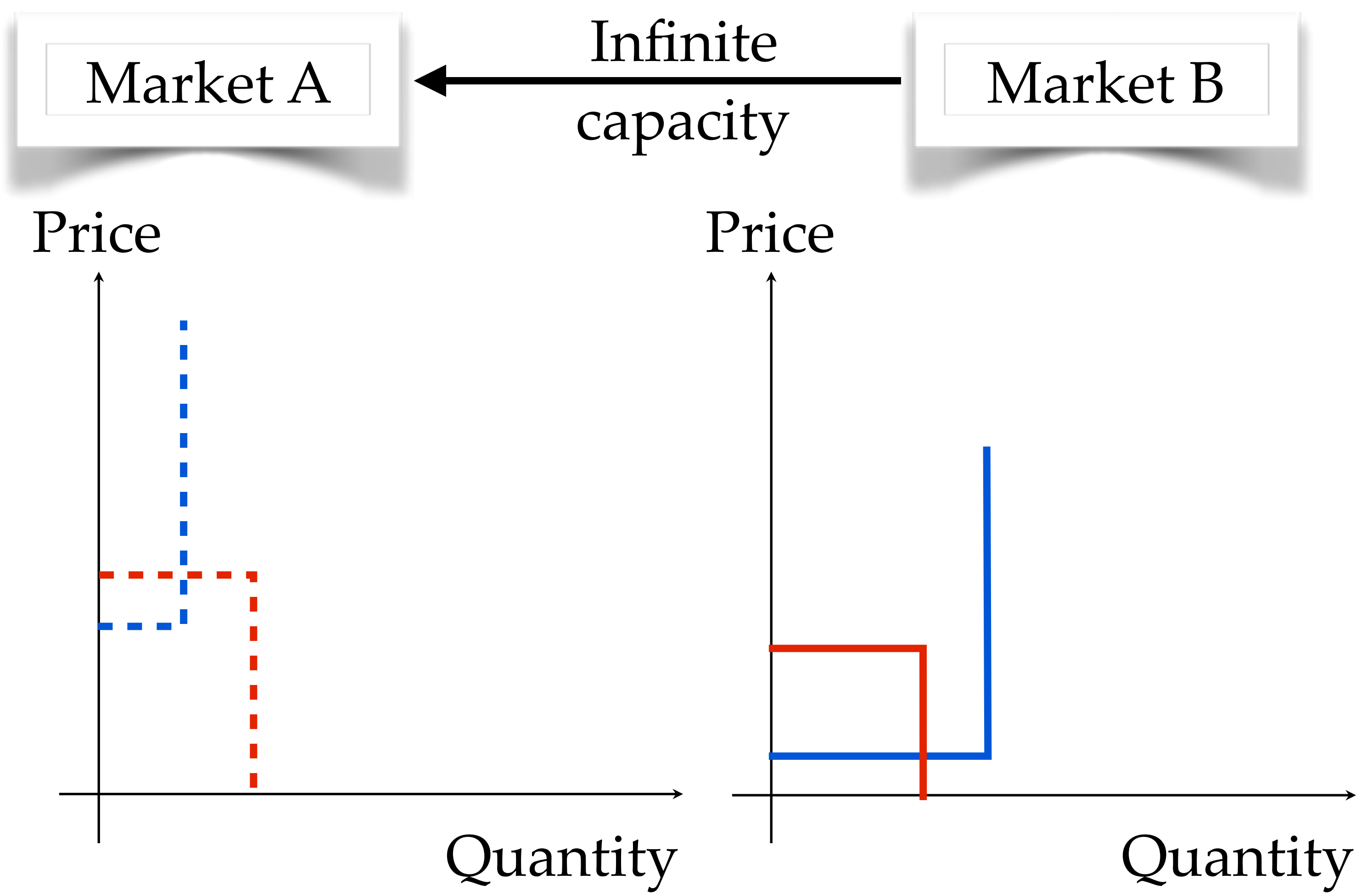




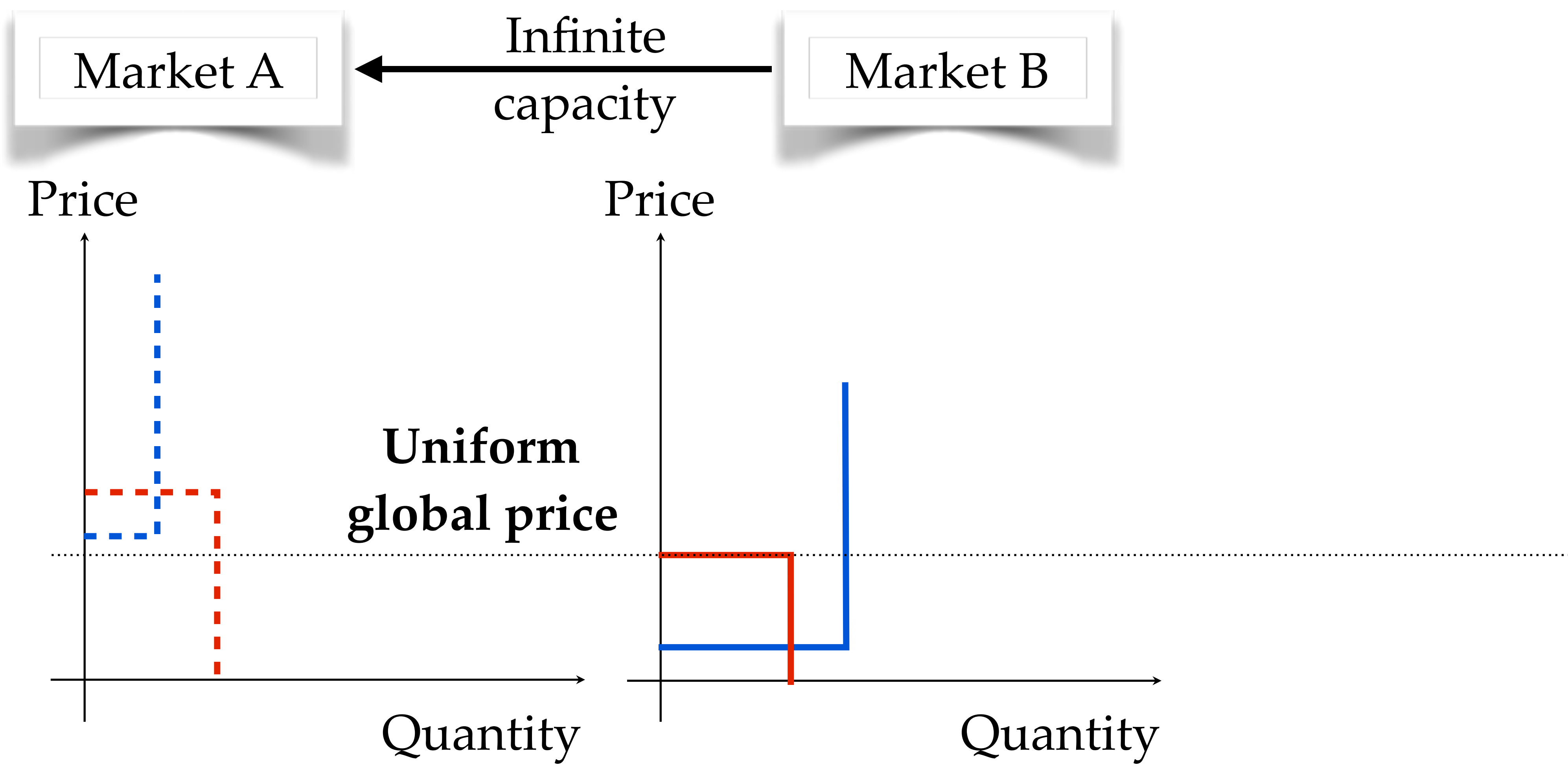
Capacitated network flow

This is not a realistic model, since there are other transmission lines and power flows according to Kirchhoff laws (non-linear).

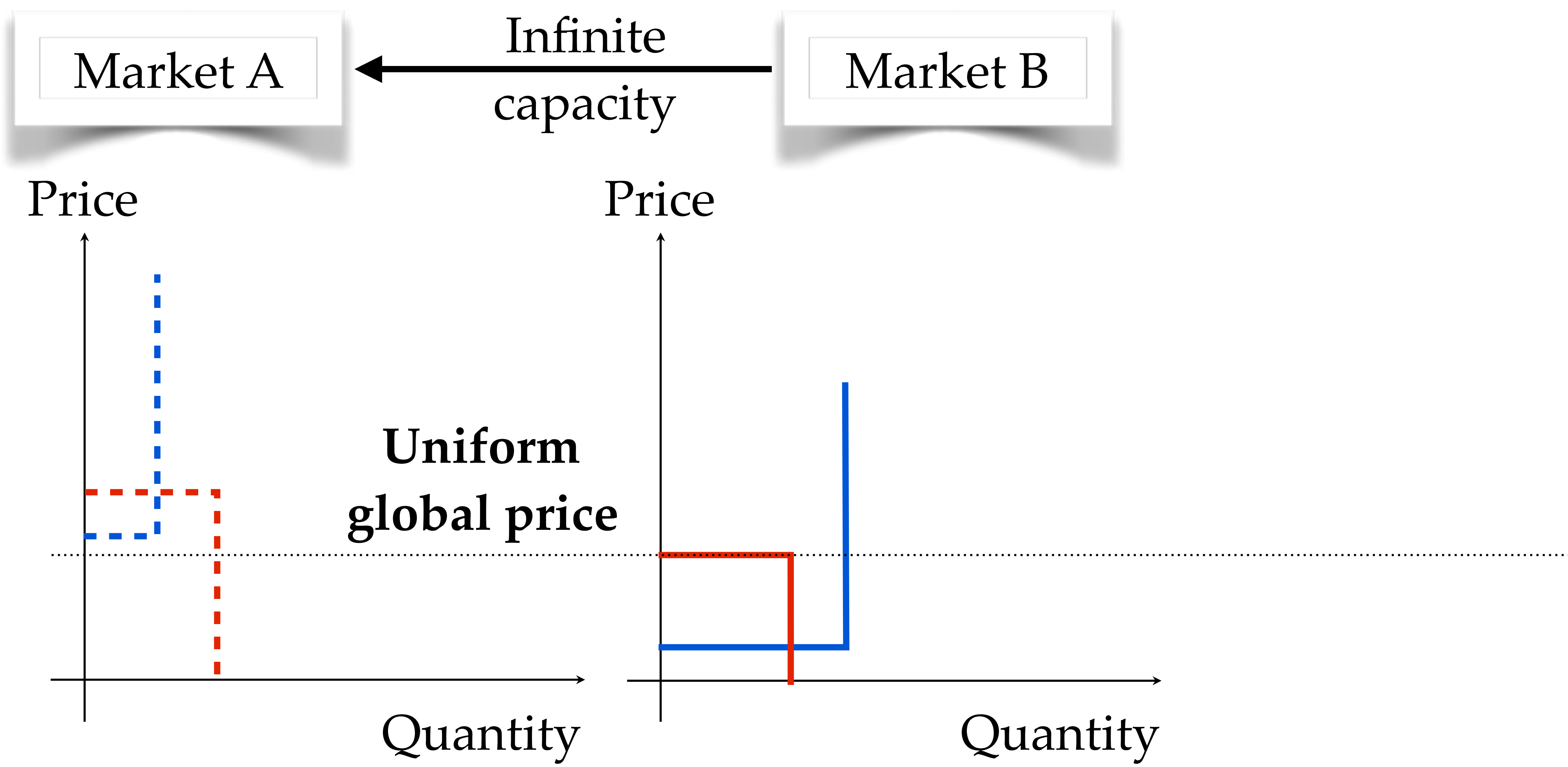
Two markets, no congestion



Two markets, no congestion

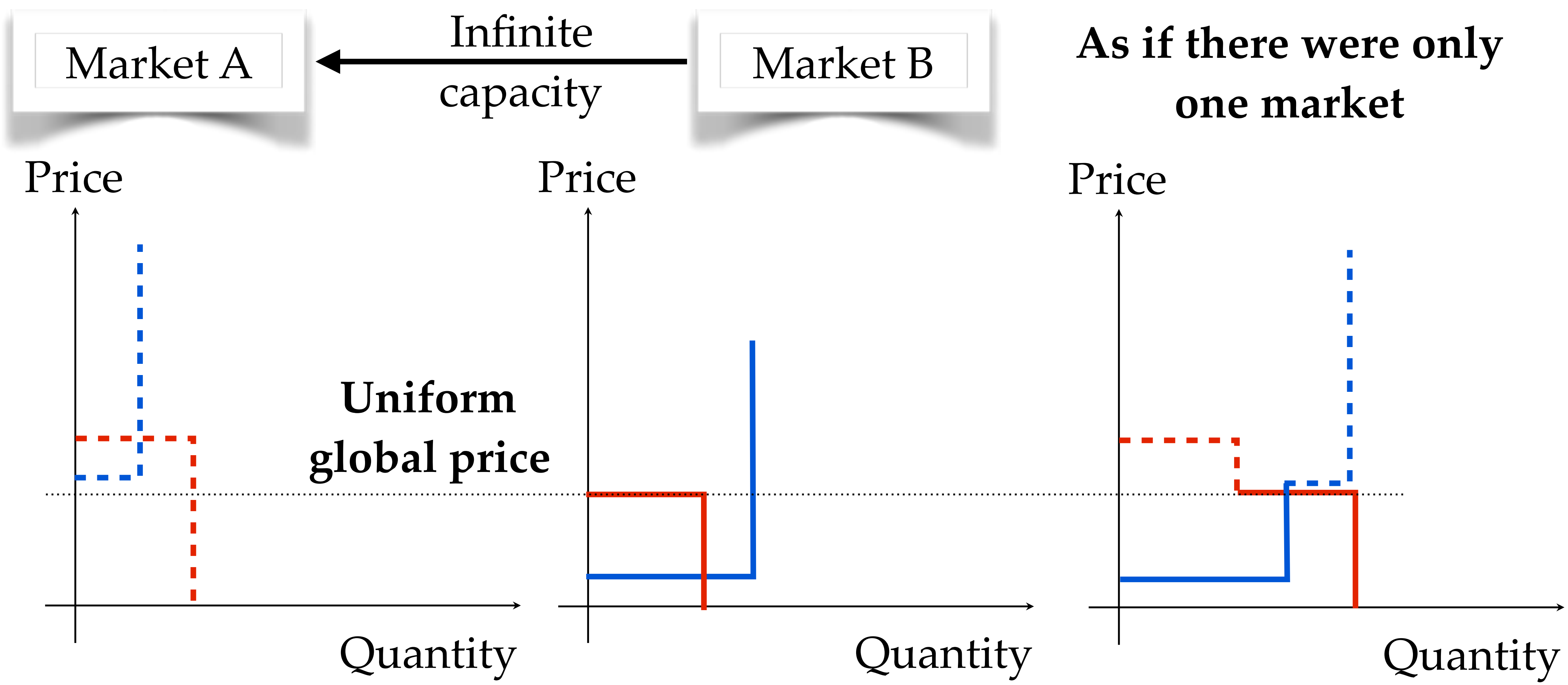


Two markets, no congestion



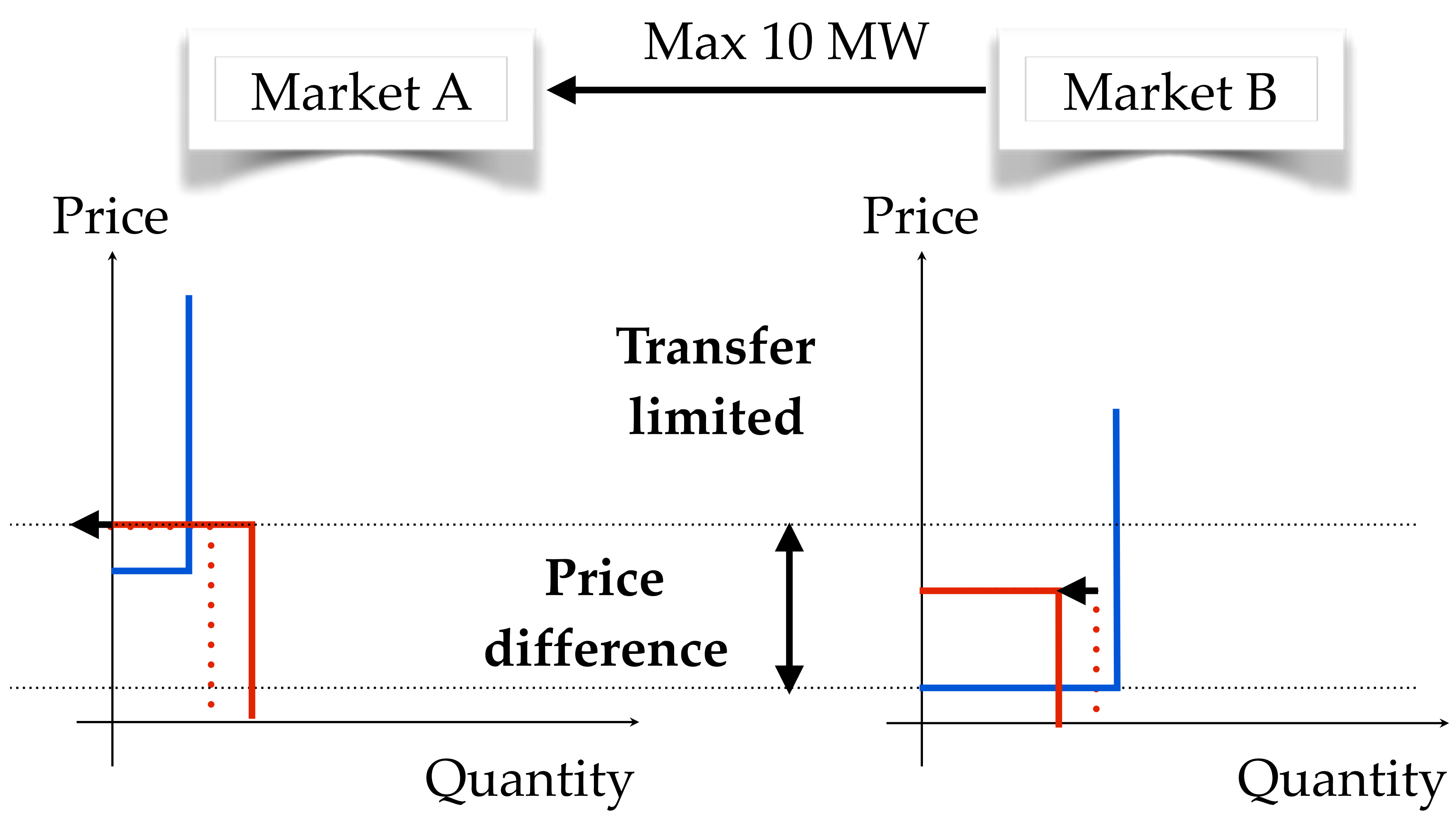
Market B exports to market A (all demand matched from supply of market B)

Two markets, no congestion

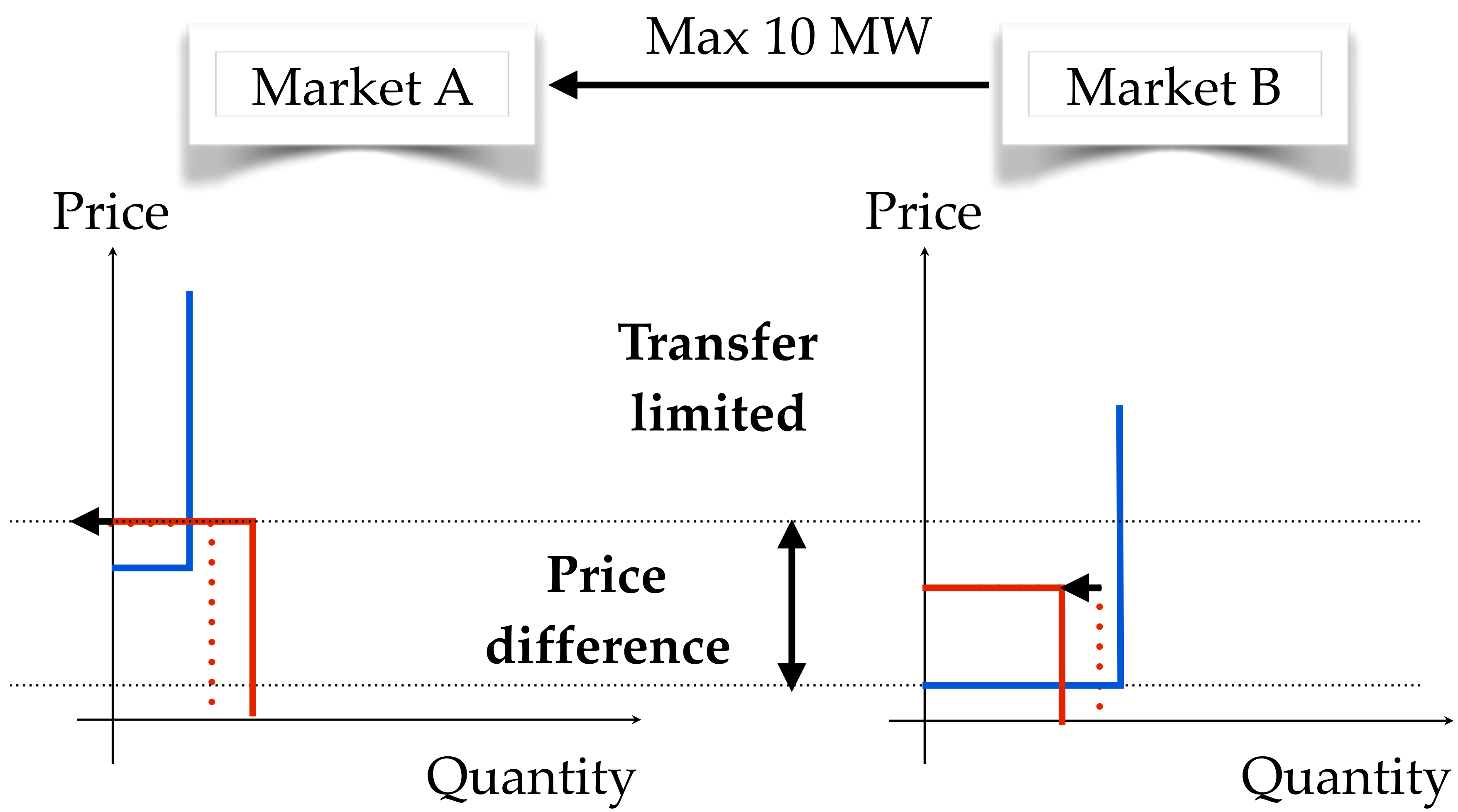


Market B exports to market A (all demand matched from supply of market B)

Two markets, congestion



Two markets, congestion



Market B exports to market A, but not enough to equalize prices

Refining the network model

Flow based model

- Instead of an ATC model, a more realistic representation is achieved by expressing linear constraints on net exports of bidding areas
- Coefficients of net exports, called **Power Distribution Coefficient Factors (PTDF)**, are obtained thanks to an approximate sensitivity analysis around the expected working point of the system
- Issues with the economic interpretation of prices

Losses and tariffs on DC inter-connectors

Network ramping, ...

"Flow based" network model

Goals:

- express **better** the physical constraints of the network
- Allocate more capacity
- increase welfare

How?

A set of *critical branches* (CBs) are considered. Critical branches are lines, cables or devices that can be heavily impacted by cross-border exchanges. These are not only cross-border lines.

The expected loading of CBs is evaluated based on long term nominations. Part of the remaining margin can be allocated to day-ahead markets.

The impact of cross border exchanges on CBs is modelled through the net export of the bidding areas in the same *balancing area*.

Balancing area: set of bidding areas for which sum of net exports is zero. Can exchange energy with other balancing area, but accounted in another variable. E.g. CWE, FR + BE + NL + DE

NordPool Spot

Linked blocks

- Acceptance of one block conditioned by acceptance of other blocks

Flexible blocks

- e.g. a block of one hour that may be accepted at any period

GME (Italy)

Italy is split in several sub-markets

We must determine one common clearing price for all demand orders whatever the sub-market m: PUN (Prezzo Unico Nazionale)

Supply orders are remunerated at zonal price P_m

Assume Q_m is the quantity matched in zone m

Goal: zero imbalance

$$PUN \sum_m Q_m = \sum_m P_m Q_m$$

OMIE
(Spain – Portugal – Morocco)

"Complex Order" defined by

- Several supply curves for several periods

- A Minimum Income Condition

quantity matched   market price   Minimum Income

$$\sum_h Q_h P_h \geq \boxed{F} + \boxed{V} \sum_h Q_h$$

period

- Bounded variations between consecutive periods

$$|Q_h - Q_{h-1}| \leq \boxed{LG}$$

Price / volume indeterminacies

- When curves cross on a vertical segment, there is a price indeterminacy
  - Rule: try to be as close as possible from the mid point of the intersection interval
- Similarly, when curves cross on a horizontal segment, there is a volume indeterminacy
  - Rule: maximize traded energy
  - Note: there are other curtailment rules (local matching, ...)

Outline of the lecture

1. Definitions and market rules
2. **Selected topics in Mathematical Programming**
3. Formalization of the day-ahead market coupling problem
4. A few words about the solution method implemented in EUPHEMIA

Optimization

Find the best solution to a problem (the best according to a pre-specified measure, a function) and satisfying some constraints.

Example: find the longest meaningful word composed of letters in the set{T,N,E,T,E,N,N,B,A}

https://www.youtube.com/watch?v=lsFAokXCxTI

The problem is usually cast in mathematical language.

The solution method is usually automatic, that is an algorithm implemented on a computer.

Linear programing (LP)

Objective and constraints are linear expressions, and variables have continuous domains.

Example:

$$\max_{x_1, x_2 \geq 0} \quad 5x_1 + 11x_2 \tag{4.2}$$

$$\text{s.t.} \quad x_1 \leq 6 \tag{4.3}$$

$$x_1 - 3x_2 \geq 1 \tag{4.4}$$

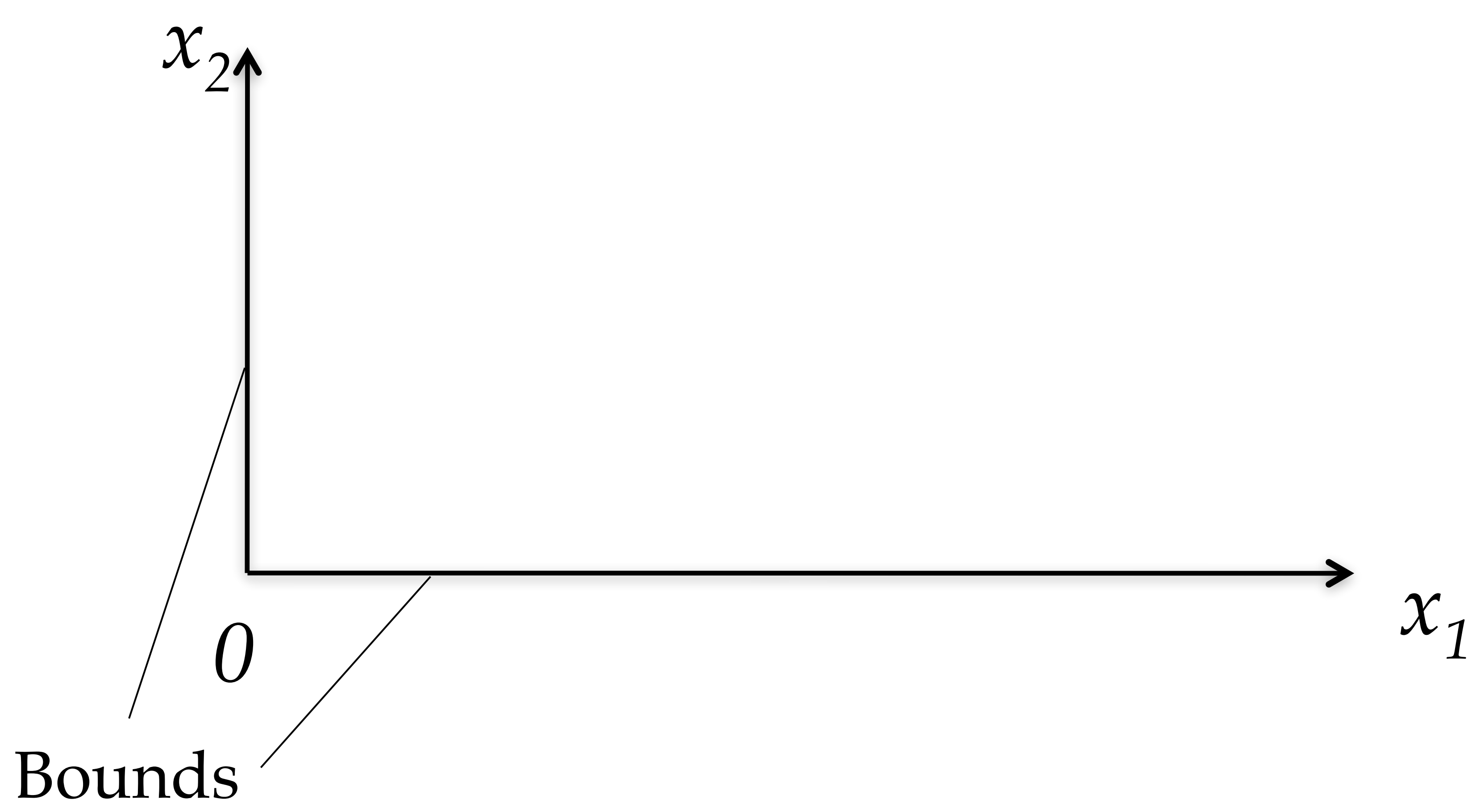
$$3x_1 + 2x_2 \leq 19 \tag{4.5}$$

Properties:

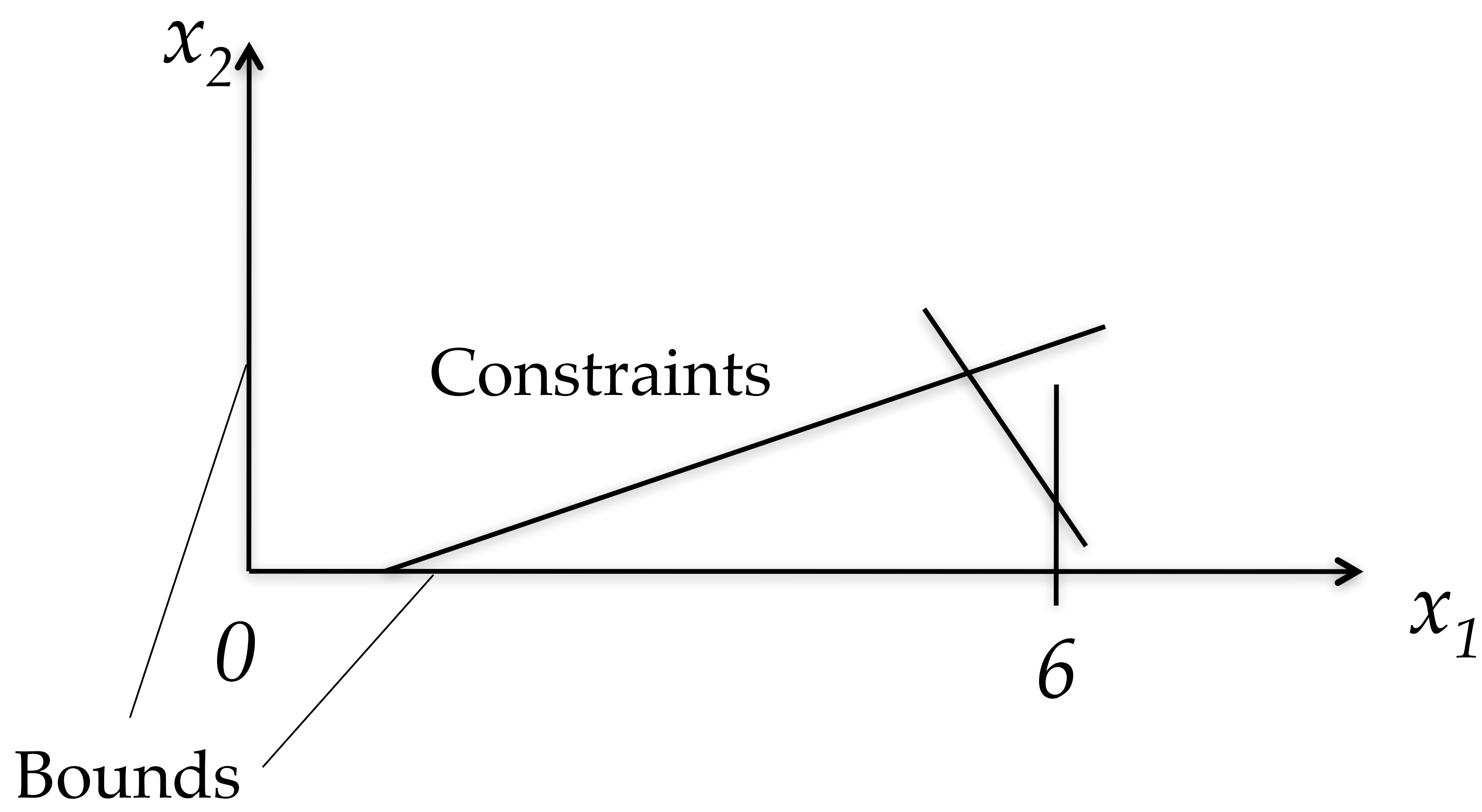
- The feasible domain is a polyhedron.
- Optimal solution(s) lie on the boundary of that polyhedron.

Graphical representation of a Linear Program

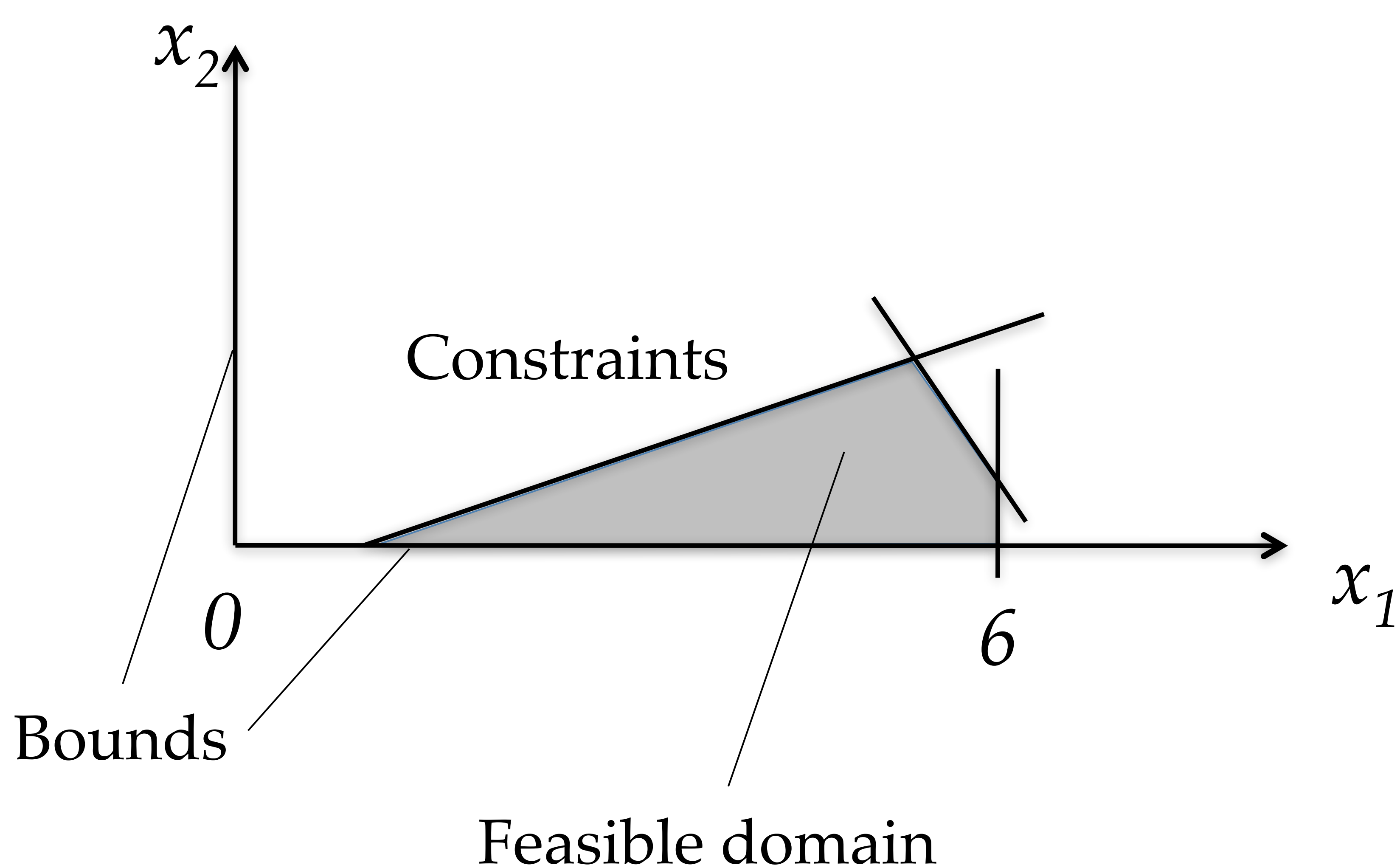
Graphical representation of a Linear Program



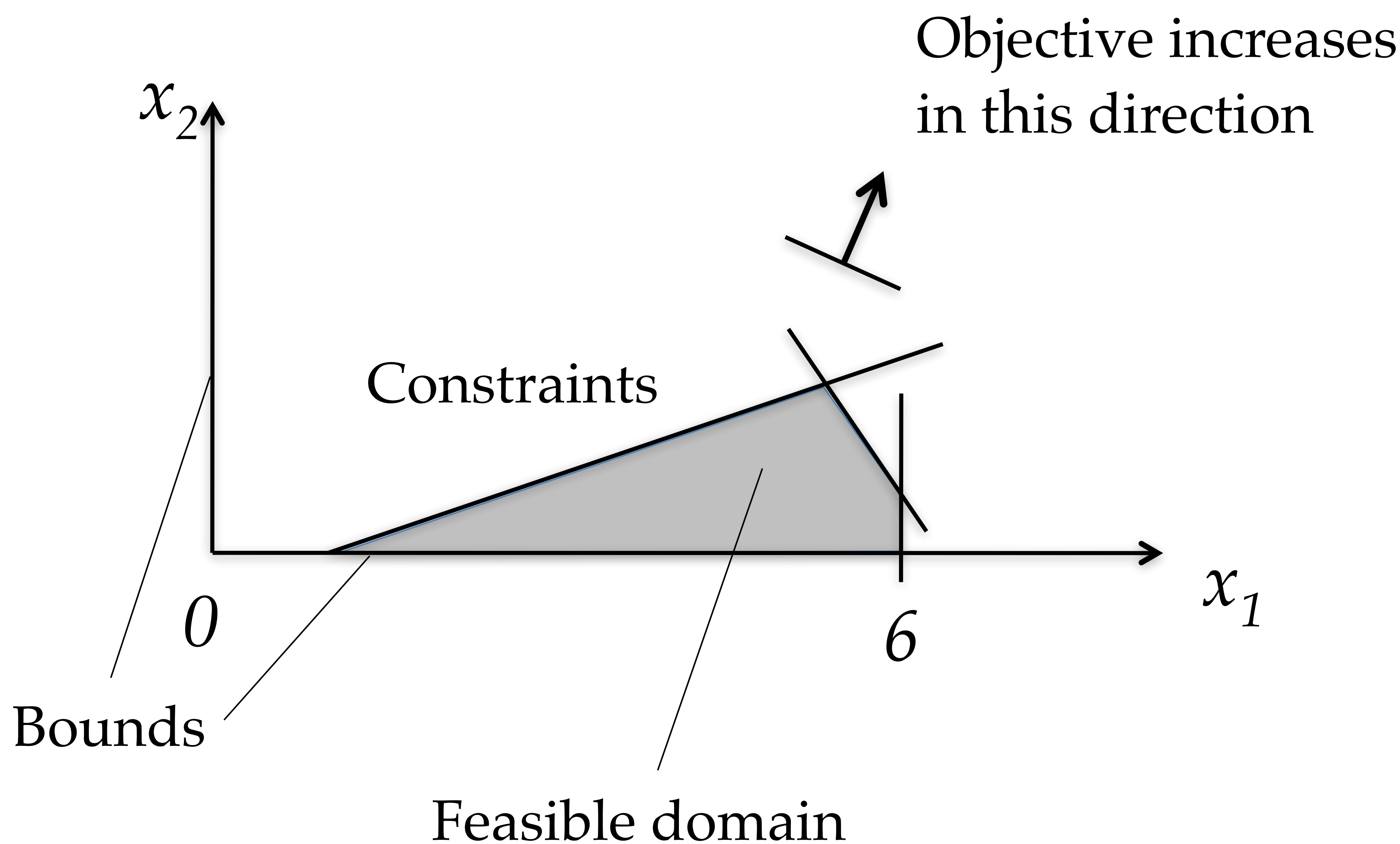
Graphical representation of a Linear Program



Graphical representation of a Linear Program



Graphical representation of a Linear Program



Duality in linear programing

To each LP (called the primal) corresponds a dual problem

Primal	**Dual**
$\max\limits_{x} c^\top x$	$\min\limits_{y} b^\top y$
s.t. $Ax \leq b$	s.t. $A^\top y \geq c$
$x \geq 0$	$y \geq 0$

- *A* is the constraints matrix, *c* a vector of cost coefficients and b a vector of right hand side coefficients
- each variable (x_i) in the primal corresponds to a constraint in the dual
- each constraint of the primal corresponds to a variable (y_j) in the dual

Complementary slackness

At optimality, the following relations hold:

$$y_i(a_i^\top x - b_i) = 0$$
$$x_j(A_j^\top y - c_j) = 0$$

For all rows i and all columns j of A, where a_i is row i of matrix A and A_j is column j of matrix A (vectors are always understood as column vectors)

This means that, at optimality, either a primal (resp. dual) constraint is tight (satisfied to equality) or the corresponding dual (resp. primal) variable is zero.

Solving very large LPs

Simplex

- moves from one vertex (extreme point) of the feasible domain to another until objective stops decreasing
- very efficient in practice but can be exponential on some special problems
- can keep information of one solution to quickly compute a solution to a perturbed problem (useful in a B&B setting), dual simplex, ...

Barrier

- iteratively penalizes the objective with a function of constraints, to force successive points to lie within the feasible domain
- polynomial time, very efficient especially for large sparse systems
- but no extremal solution hence crossover required in a B&B setting

Convex optimization

Those results generalize to problems more general than LP, that is when the objective and the feasible domain are convex.

There is a theoretical guarantee that there exist algorithms to solve those problems efficiently.

Example: (convex) Quadratic Programming (QP) are problems where the objective is quadratic and constraints are linear. The simplex and barrier algorithms can be adapted to QP.

Mixed Integer programing (MIP)

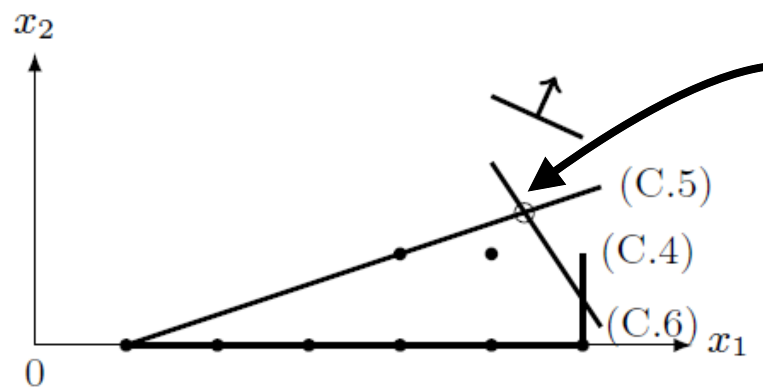
Idem as before, except that **some variables must take integer values**.

In general, **relaxing** the integrality requirement and solving the resulting continuous optimization problem does not yield a feasible solution to the original problem. Simple rounding procedures do not necessarily restore feasibility, and even if it does, do not guarantee optimality. However, the **continuous relaxation** provides **a bound on the optimum of the original problem**.

Simple **enumeration** of combinations of integer variable values is computationally undoable. **Branch-and-bound** is a clever way to do enumeration. It progressively imposes integer values and uses the solution to intermediate continuous relaxations to obtain bounds and thus avoid exploring some combinations, without losing optimal solutions.

Example of MIP

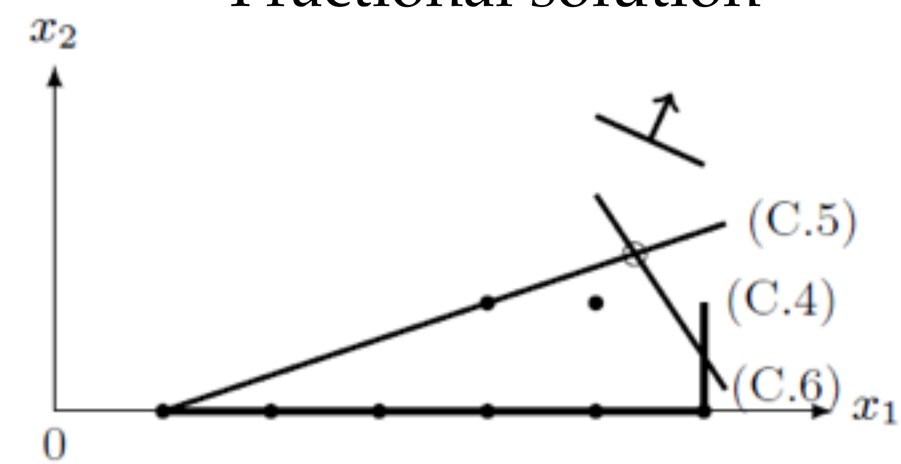
$$
\begin{aligned}
\max \quad & 5x_1 + 11x_2 && \text{(C.3)}\\
\text{s.t.} \quad & x_1 \le 6 && \text{(C.4)}\\
& x_1 - 3x_2 \ge 1 && \text{(C.5)}\\
& 3x_1 + 2x_2 \le 19 && \text{(C.6)}\\
& x_1, x_2 \in \mathbb{Z}_+ && \text{(C.7)}
\end{aligned}
$$



The solution of the continuous relaxation is not integer

Branch and bound example

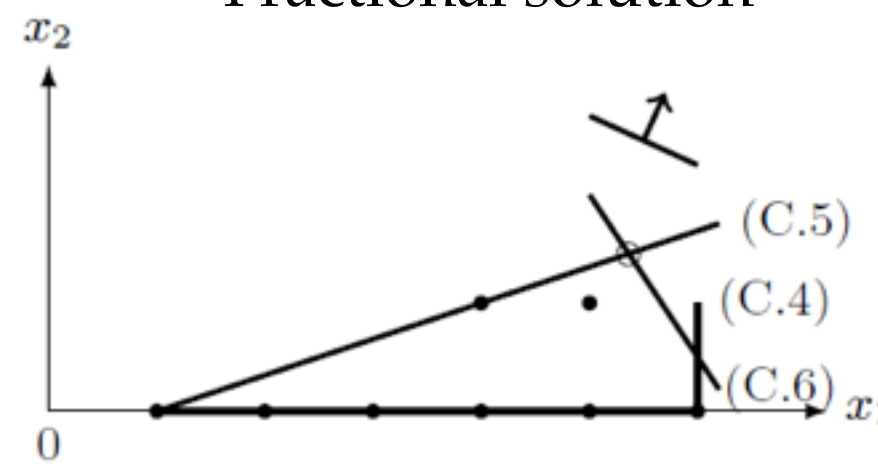
Fractional solution



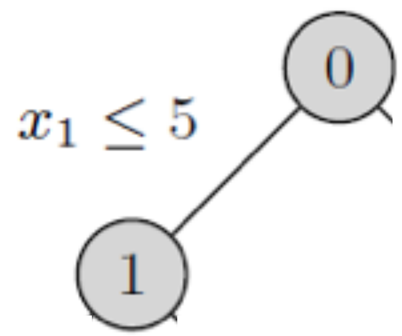
(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

Branch and bound example

Fractional solution

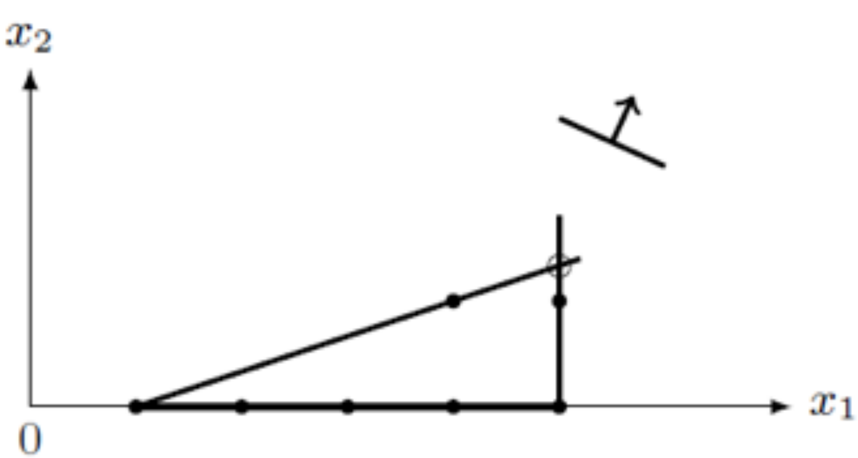


(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

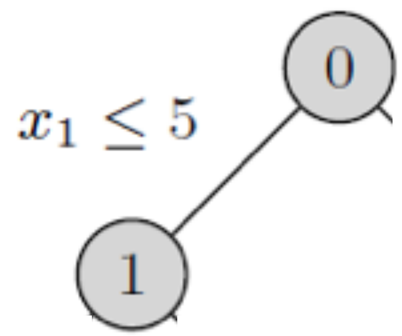


Branch and bound example

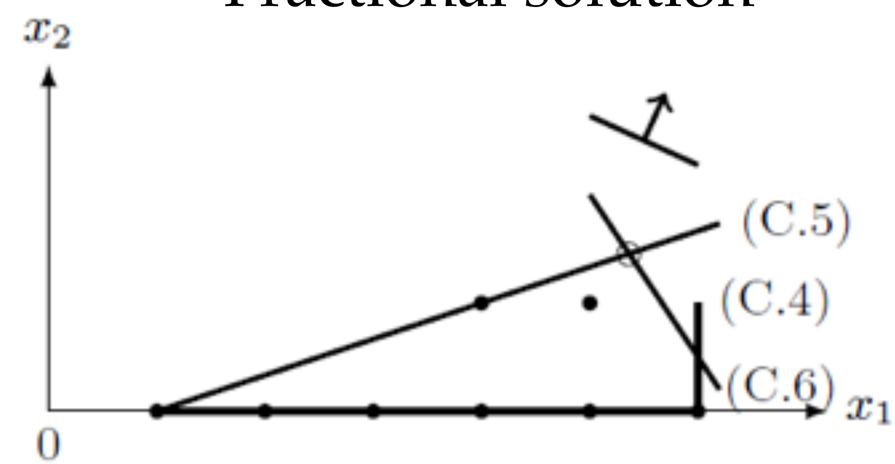
Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.



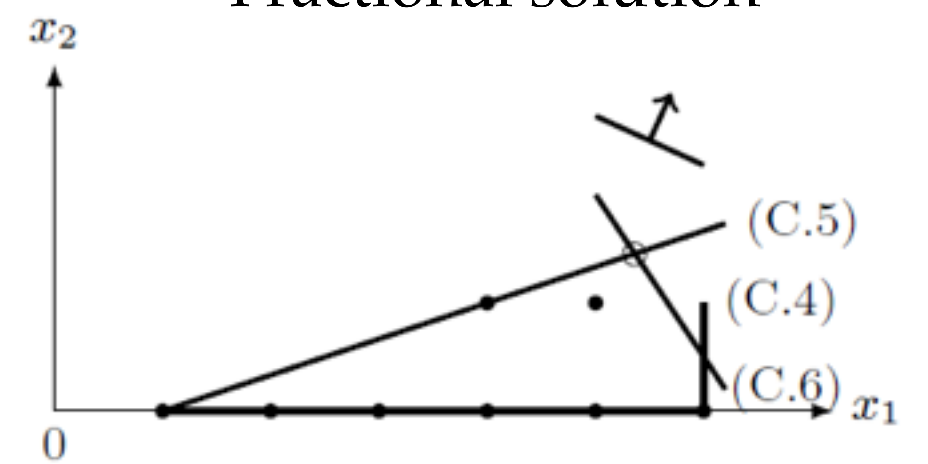
Fractional solution



(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

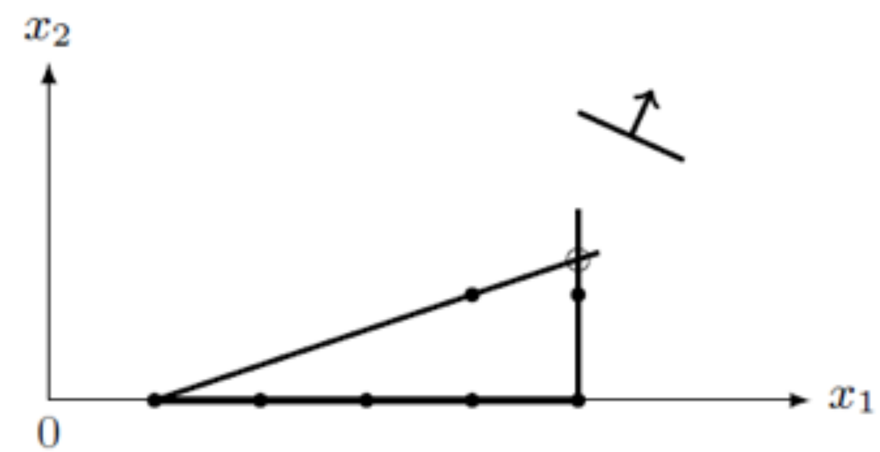
Branch and bound example

Fractional solution

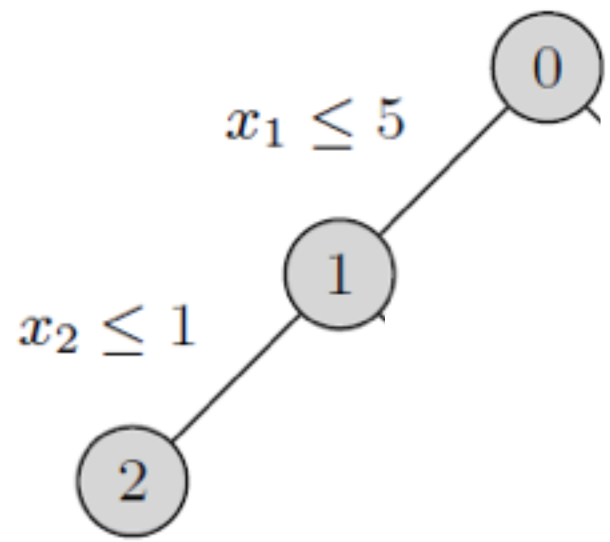


(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

Fractional solution

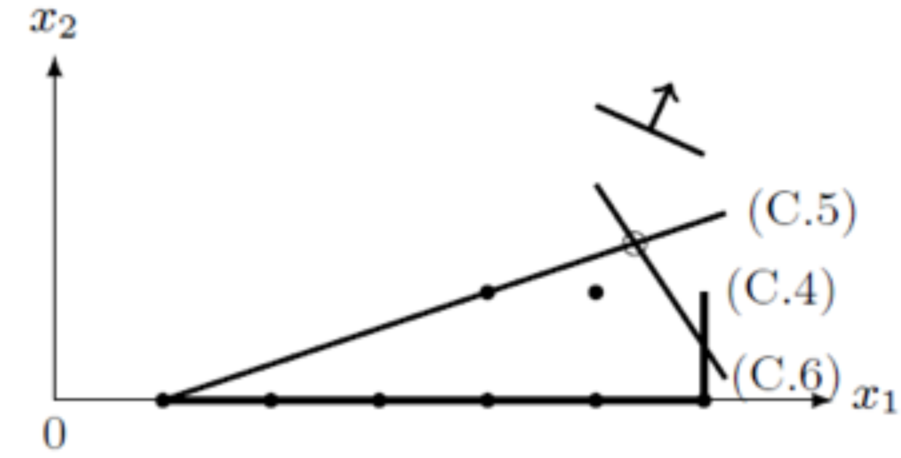


(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.

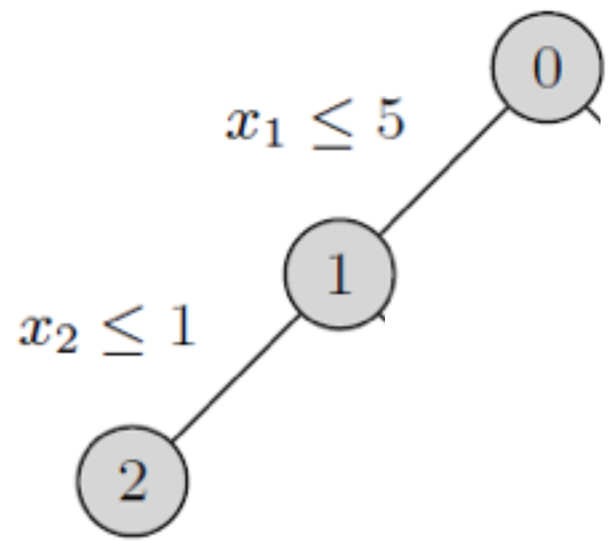


Branch and bound example

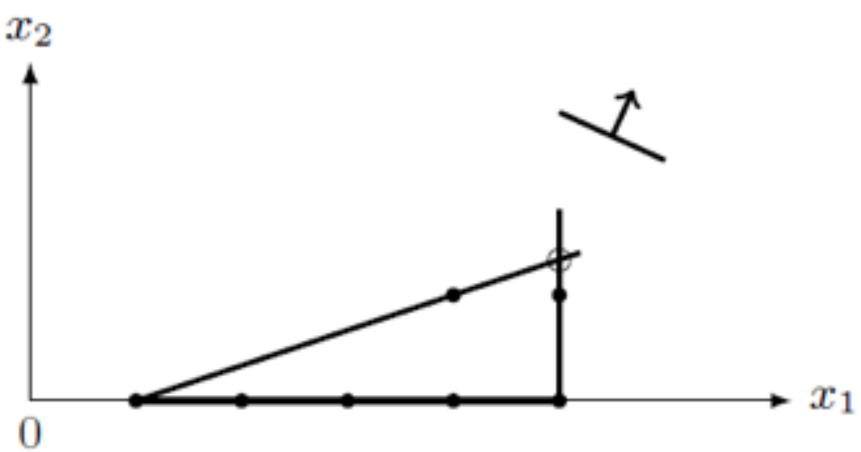
Fractional solution



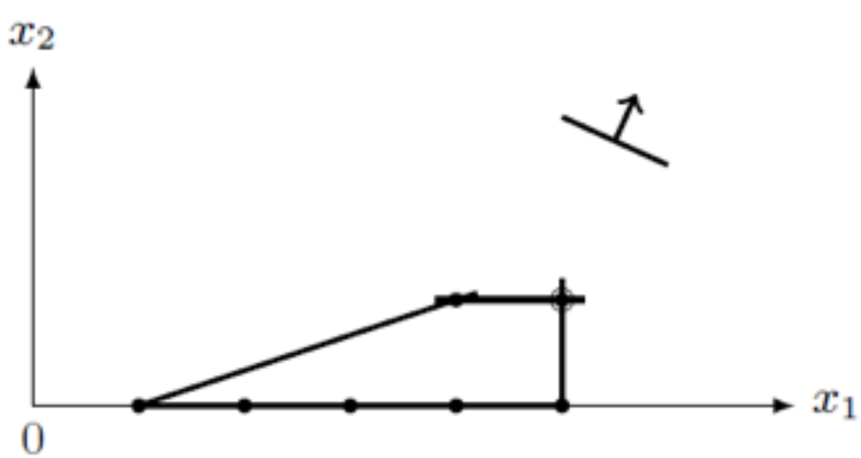
(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.



Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.

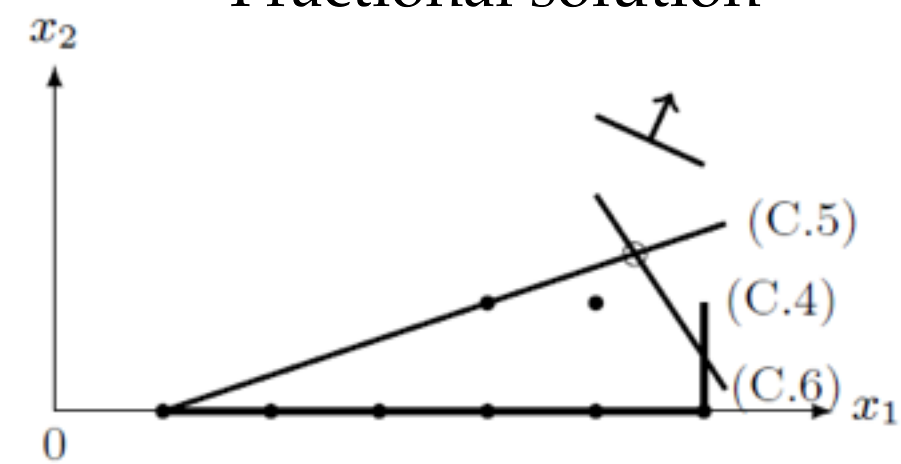


(c) Node 2: $z^{\star,2} = 36$ and $x^{\star,2} = (5, 1)$.

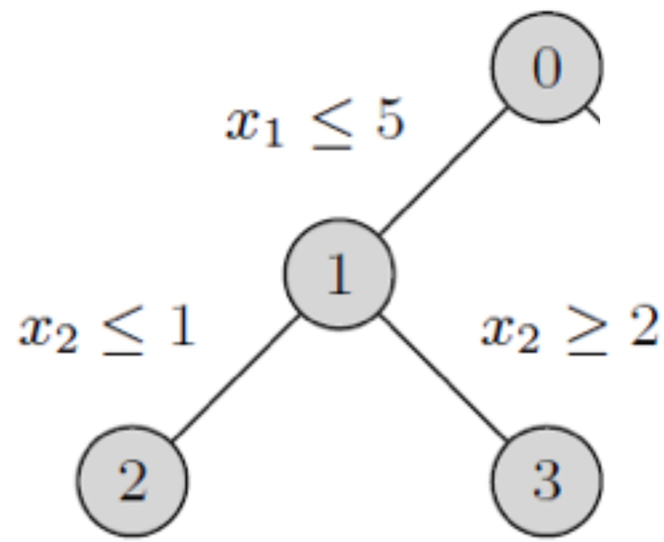
Integer solution

Branch and bound example

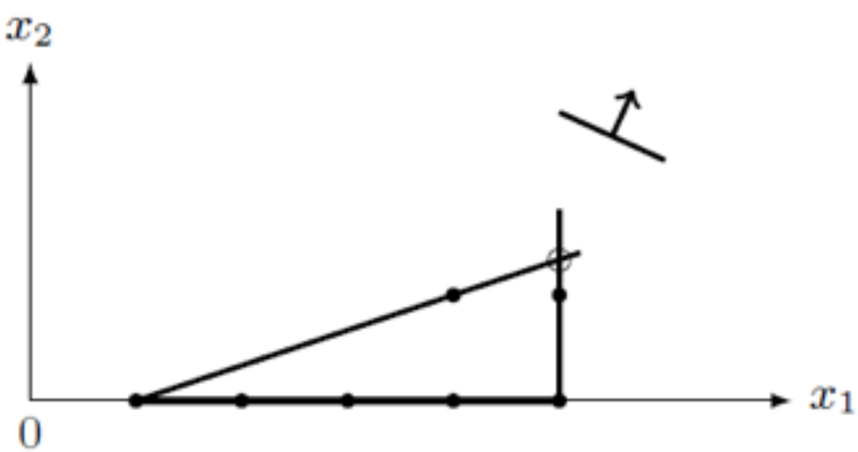
Fractional solution



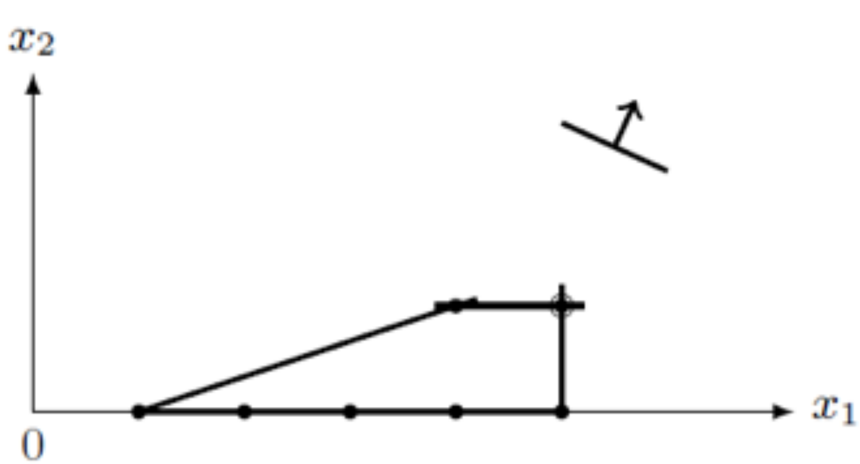
(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.



Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.

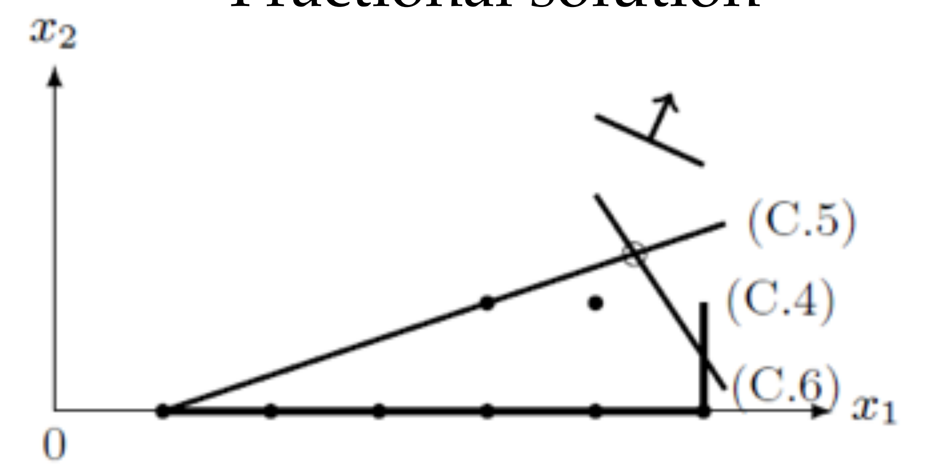


(c) Node 2: $z^{\star,2} = 36$ and $x^{\star,2} = (5, 1)$.

Integer solution

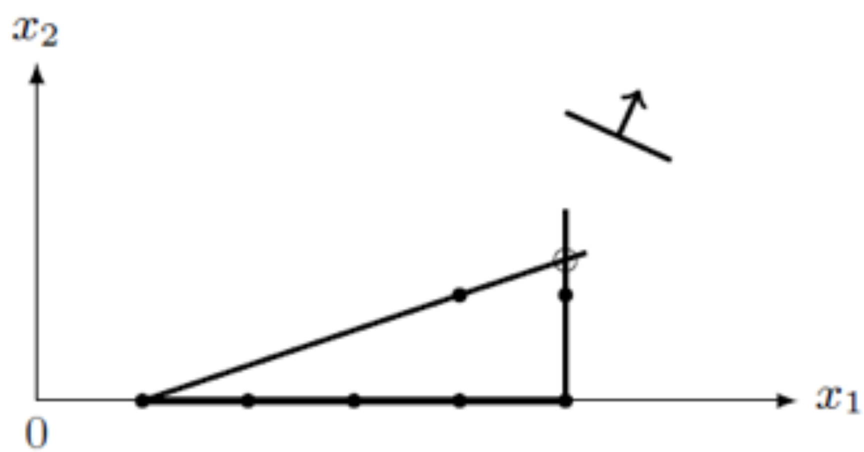
Branch and bound example

Fractional solution

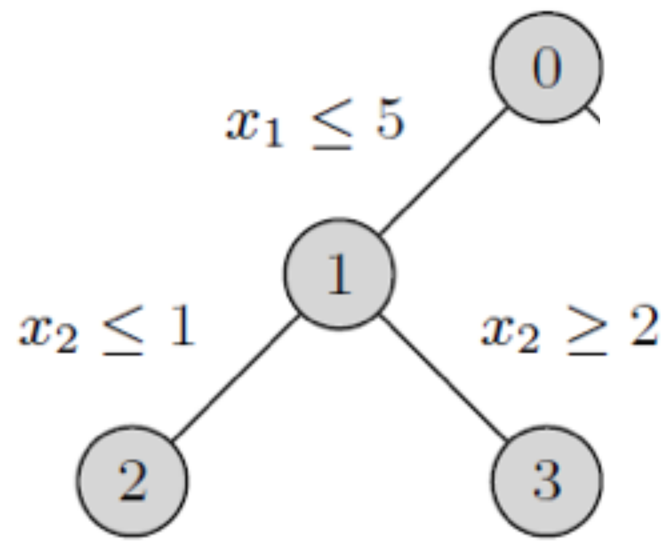


(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

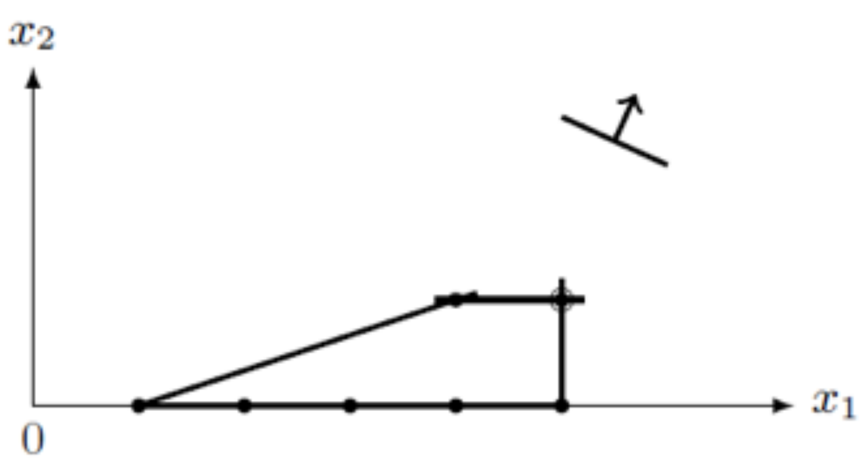
Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.



Prune by infeasibility

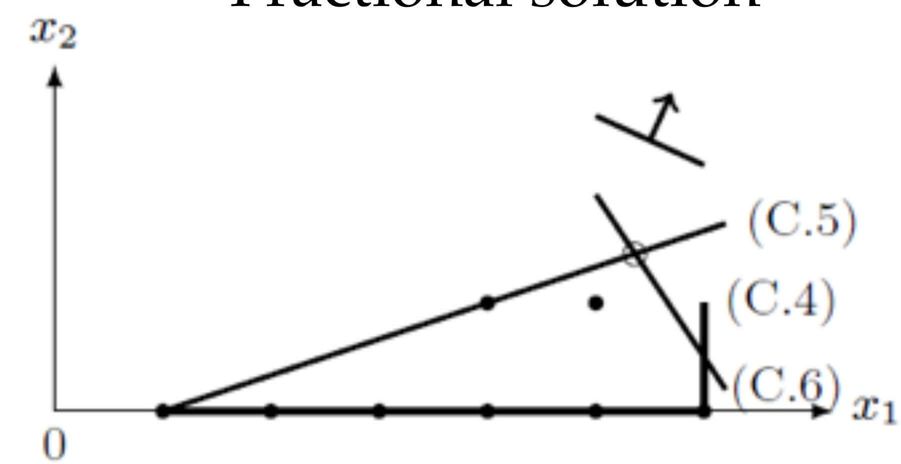


(c) Node 2: $z^{\star,2} = 36$ and $x^{\star,2} = (5, 1)$.

Integer solution

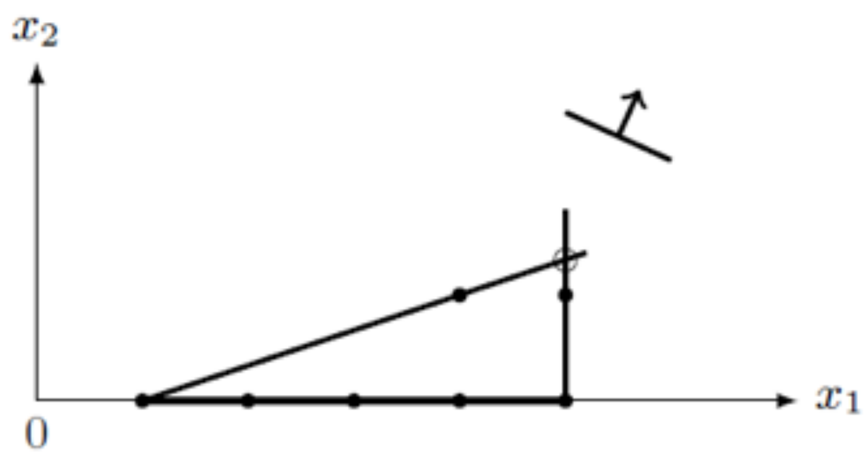
Branch and bound example

Fractional solution

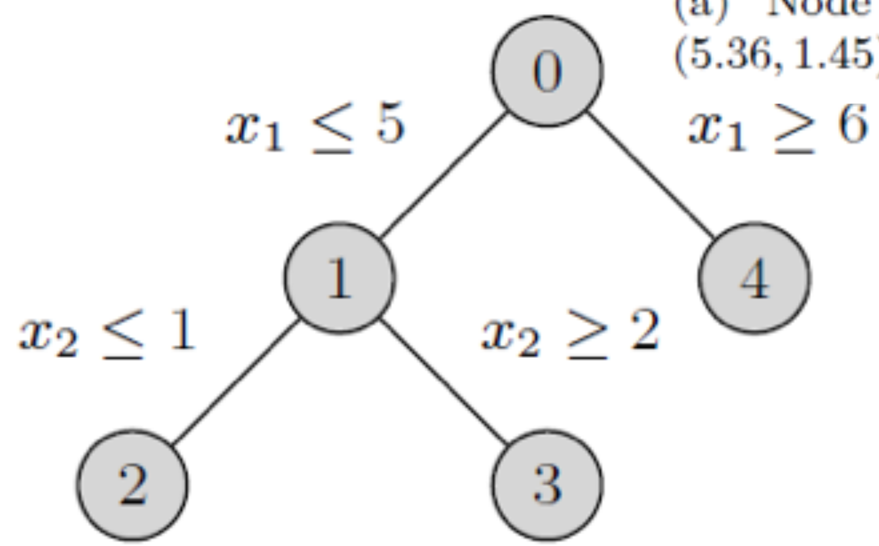


(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

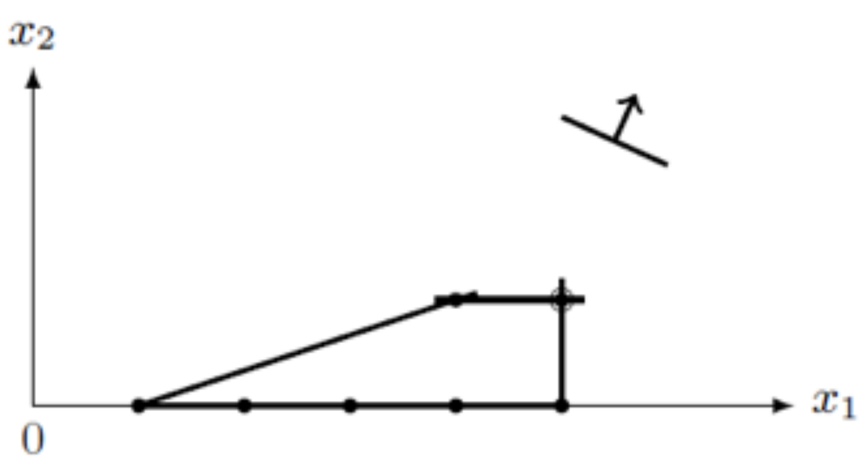
Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.



Prune by infeasibility

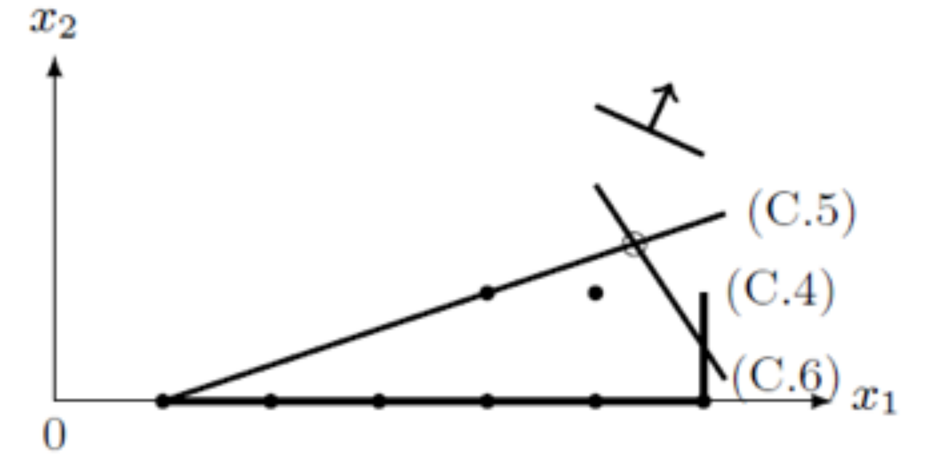


(c) Node 2: $z^{\star,2} = 36$ and $x^{\star,2} = (5, 1)$.

Integer solution

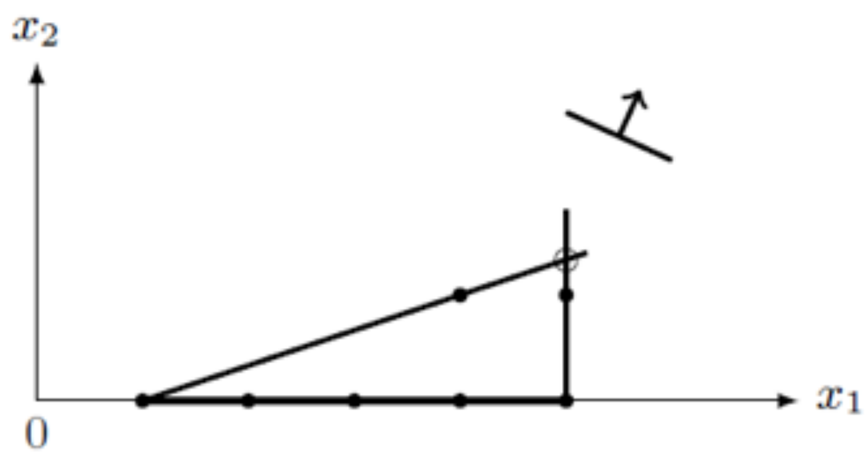
Branch and bound example

Fractional solution

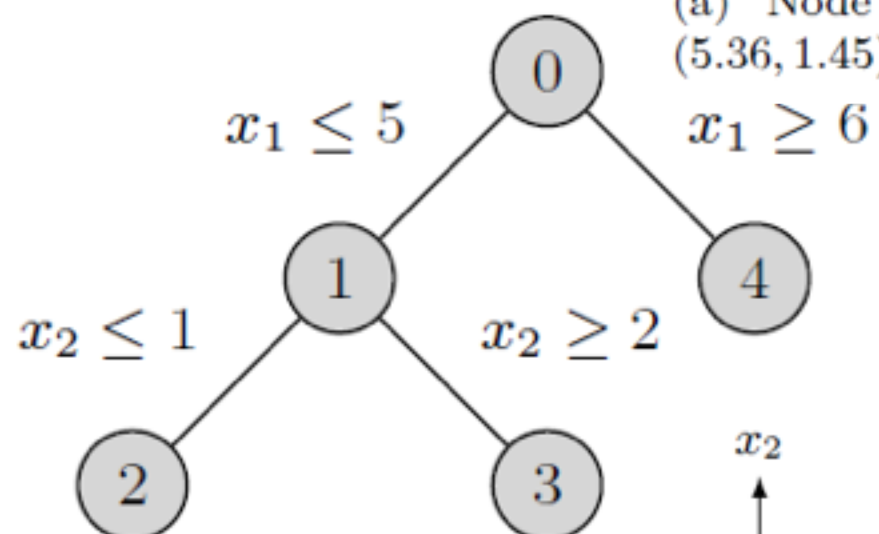


(a) Node 0: $z^{\star,0} \approx 42.82$ and $x^{\star,0} \approx (5.36, 1.45)$.

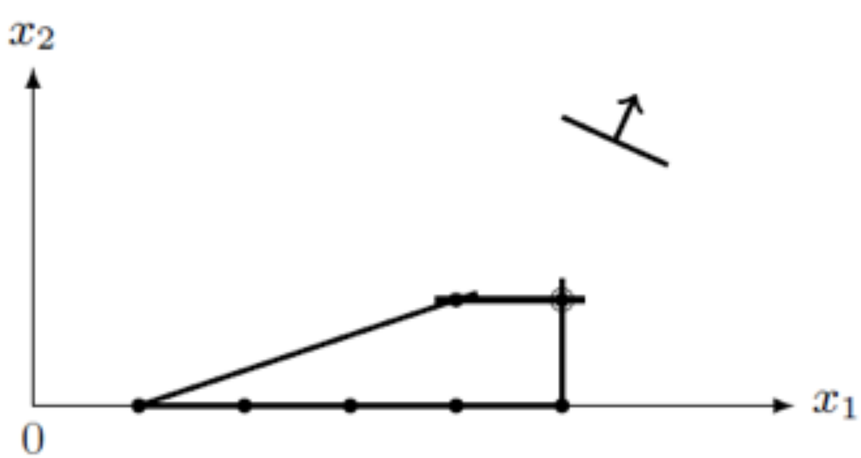
Fractional solution



(b) Node 1: $z^{\star,1} \approx 39.67$ and $x^{\star,1} = (5, 4/3)$.

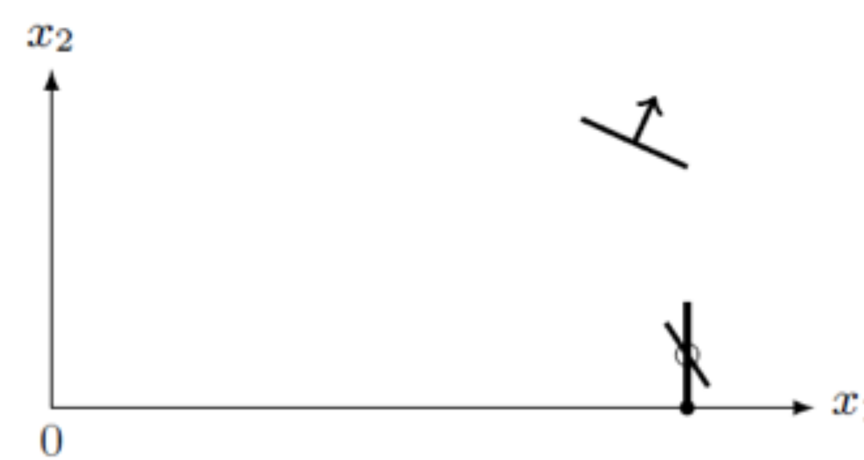


Prune by infeasibility



(c) Node 2: $z^{\star,2} = 36$ and $x^{\star,2} = (5, 1)$.

Integer solution



(d) Node 4: $z^{\star,4} = 35.5$ and $x^{\star,4} = (6, 1/2)$.

Prune by bound

Outline of the lecture

1. Definitions and market rules

2. Selected topics in Mathematical Programming

3. **Formalization of the day-ahead market coupling problem**

4. A few words about the solution method implemented in EUPHEMIA

Features considered in the remainder of this lecture

- ATC coupling problem with hourly orders and block orders only.
- I.e. we do not consider GME, OMIE, smart orders, nor flow based network model.

Nature of the mathematical problem

It is a mathematical program with complementarity constraints (MPCC) and in addition it contains integer decision variables.

It enters the category of Mixed Integer Non-Linear Programs, meaning that the continuous relaxation of the problem is non-convex.

As we will see in the next section, Euphemia approximates the problem as a Mixed Integer Quadratic Program (MIQP) that is convex (Q is positive semi-definite) and then checks the solution is compliant with the “true” problem.

The **primal** market coupling problem

maximize	Welfare
subject to	1. Clearing Constraints
	2. Network constraints
	3. Order definition constraints

Clearing constraints express the equality of generation and demand

Main decision variables:

- acceptance ratio of orders

Objective function: maximize social welfare

Exercise (solution on next page): Assume each order i of a set I is defined by

- its quantity q_i
- its start and end prices p_i^0 and p_i^1
- its type (supply or demand)

Write down the expression of welfare as a function of accepted quantities (for simplicity, do not account for block orders).

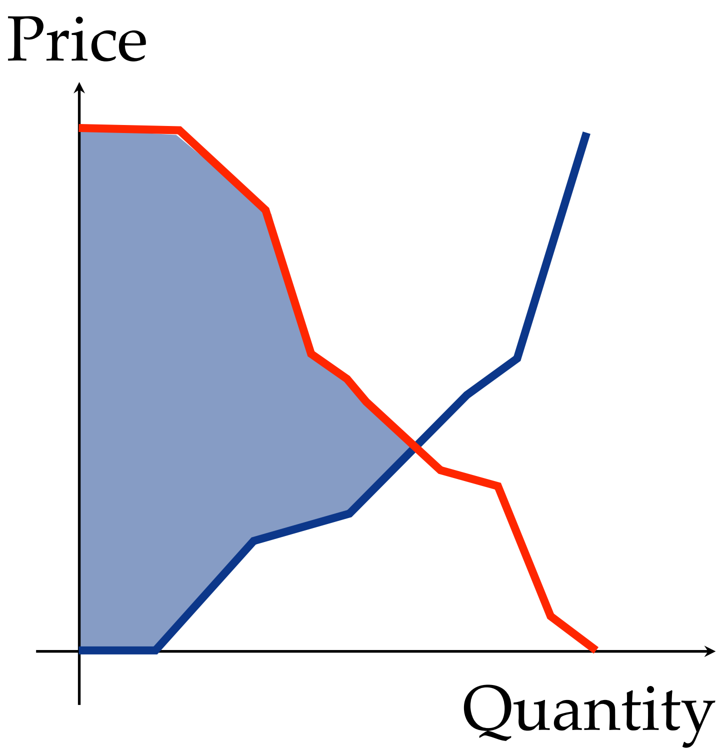
Welfare is a quadratic expression

=

-

$$\text{Welfare} = \boxed{\sum_{i \in I_{demand}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)} - \boxed{\sum_{i \in I_{supply}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)}$$

Welfare is a quadratic expression

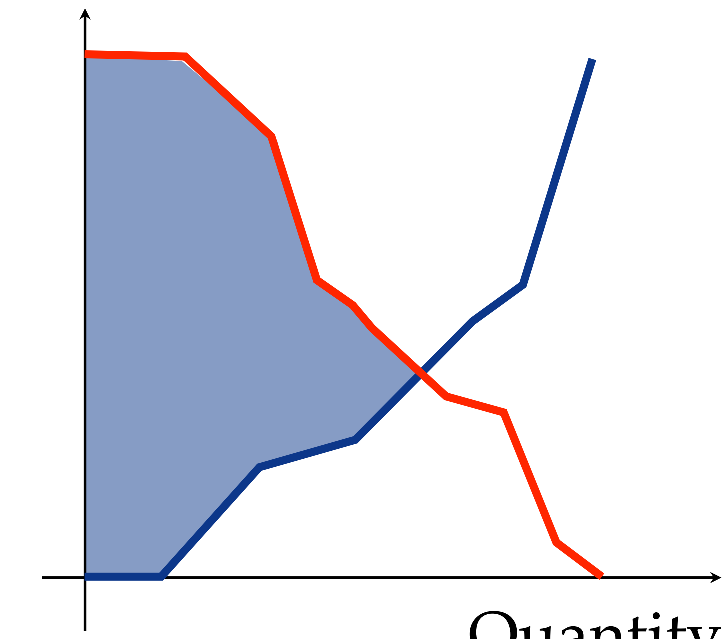


=

-

$$\text{Welfare} = \boxed{\sum_{i \in I_{demand}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)} - \boxed{\sum_{i \in I_{supply}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)}$$

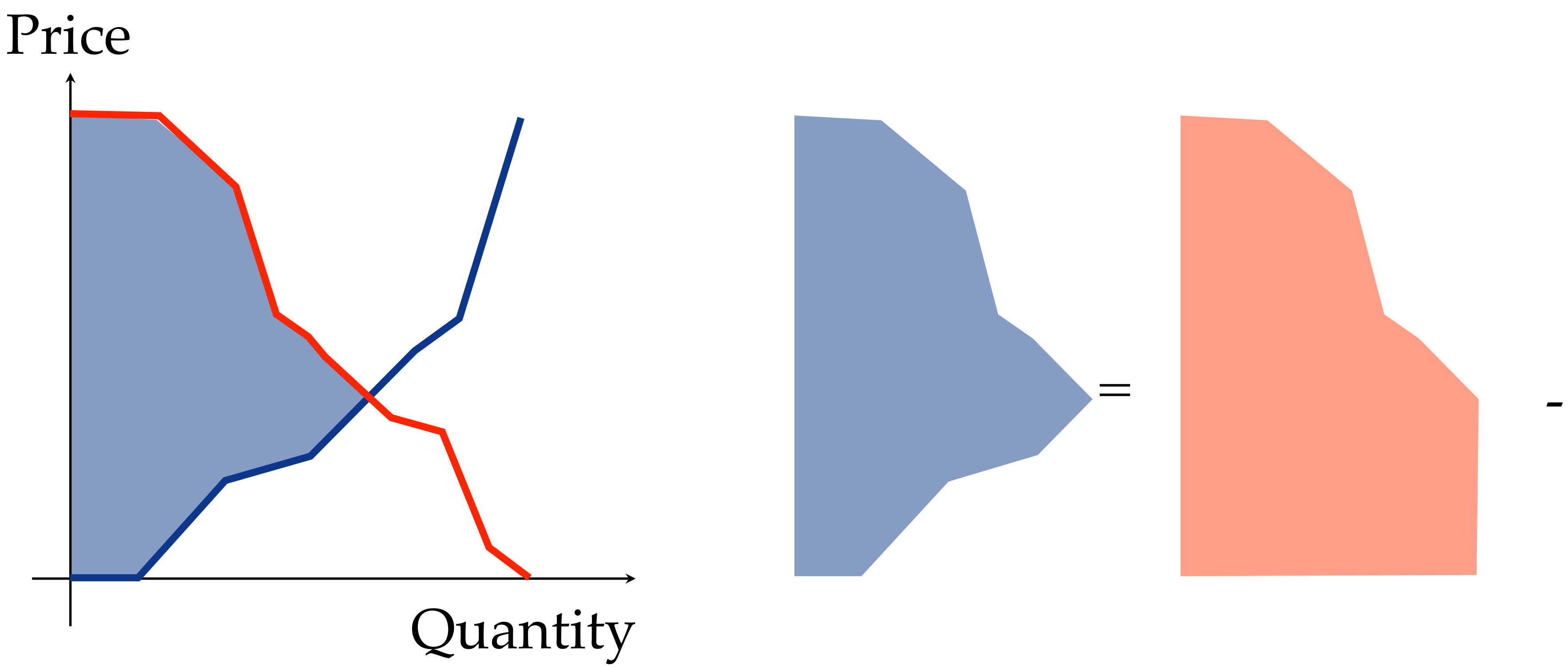
Welfare is a quadratic expression



$=$ -

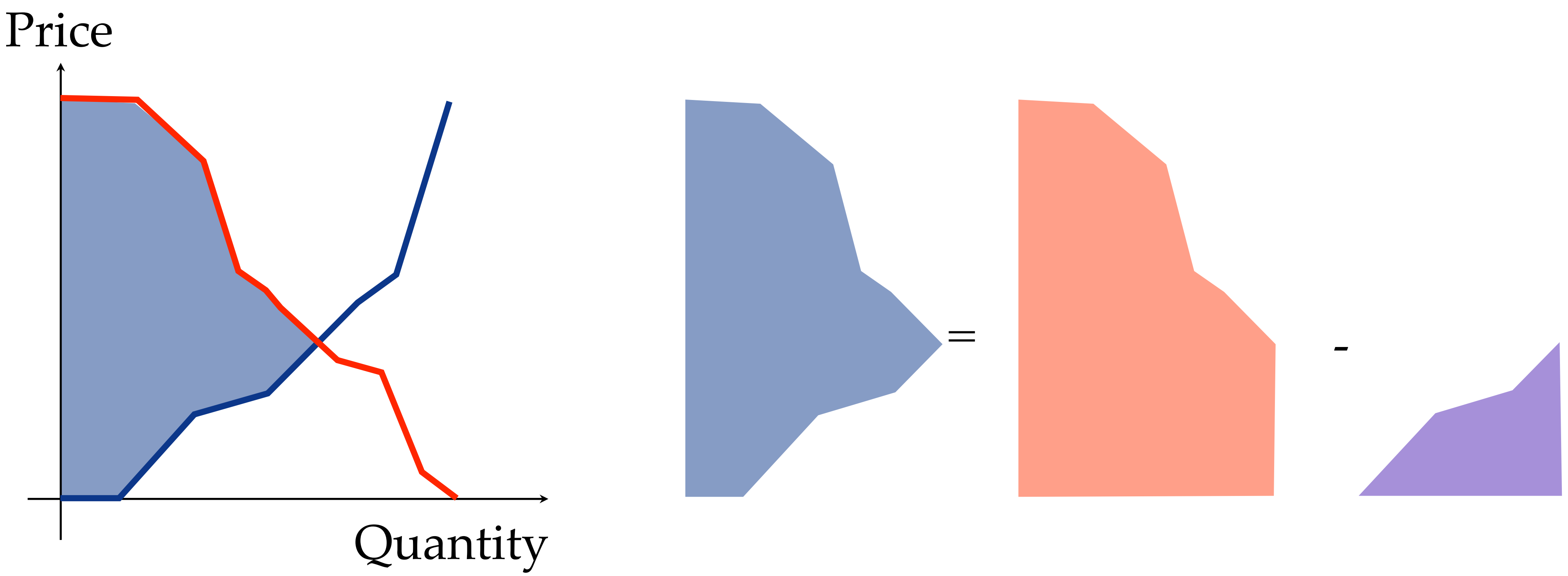
$$\text{Welfare} = \boxed{\sum_{i \in I_{demand}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)} - \boxed{\sum_{i \in I_{supply}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)}$$

Welfare is a quadratic expression



$$\text{Welfare} = \boxed{\sum_{i \in I_{demand}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)} - \boxed{\sum_{i \in I_{supply}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)}$$

Welfare is a quadratic expression



$$\text{Welfare} = \boxed{\sum_{i \in I_{demand}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)} - \boxed{\sum_{i \in I_{supply}} q_i x_i \left(p_i^0 + \frac{p_i^1 - p_i^0}{2} x_i \right)}$$

Exercise

Define the necessary variables and formulate all the primal constraints (clearing, ATC, block order definition).

Without block orders, the problem has the following properties

- The formulation is convex (QP) and can be decomposed hour by hour
- Dual solution yields market prices and congestion prices

Primal constraints	**Dual variables**
Clearing	Market Clearing Price (*MCP*)
Inter-connector capacity	Inter-connector congestion price
Orders acceptance UB	Order “surplus” (S_i)

Example: dual model without network and block orders and interpretation of complementary slackness

$$\begin{aligned} \min_{MCP, S_i \forall i \in \mathcal{I}} \quad & \sum_{i \in \mathcal{I}} S_i \\ s.t. \quad & S_i \geq q_i(MCP - p_i) \\ & S_i \geq 0 \end{aligned}$$

Complementarity relations

No Paradoxically Accepted (PA) order

$$\boxed{Accept_i(S_i - q_i(MCP - p_i)) \quad = \quad 0}$$

$$\boxed{S_i(1 - Accept_i)) \quad = \quad 0}$$

No Paradoxically Rejected (PR) order

Exercise: from the primal model of the previous exercise, find back the dual model expressed above.

Solutions naturally satisfy market rules

Order type	**MCP**	**Acceptance Ratio**
supply	$> p^1$	1
supply	$< p^0$	0
any	$>= p^0$ and $<= p^1$	$(MCP-p^0) / (p^1-p^0)$
demand	$< p^1$	1
demand	$> p^0$	0

Incorporating block orders

Complications

- integer variables, thus QP → MIQP
- time coupling
- **Paradoxically Accepted Blocks (PAB)**
  - Market says it is not acceptable to lose money, but it is acceptable to be rejected although could have made money, without compensation

Outline of the lecture

1. Definitions and market rules
2. Selected topics in Mathematical Programming
3. Formalization of the day-ahead market coupling problem
4. **A few words about the solution method implemented in EUPHEMIA**

About Euphemia

- Property of PCR members
- Developed by n-Side
- In operation since February 4, 2014. Before that the former solution COSMOS operated from November 9 2010 in the CWE region.
  - Now almost 4 years without failure
- Typical size of instances:
  - 50.000 orders
  - 700 blocks
- COSMOS solved real instances in less than 10 seconds. Now the algorithm takes several minutes.

Main idea

For a fixed selection of blocks, the Market Coupling Problem can be written as a QP

Solving this problem yields

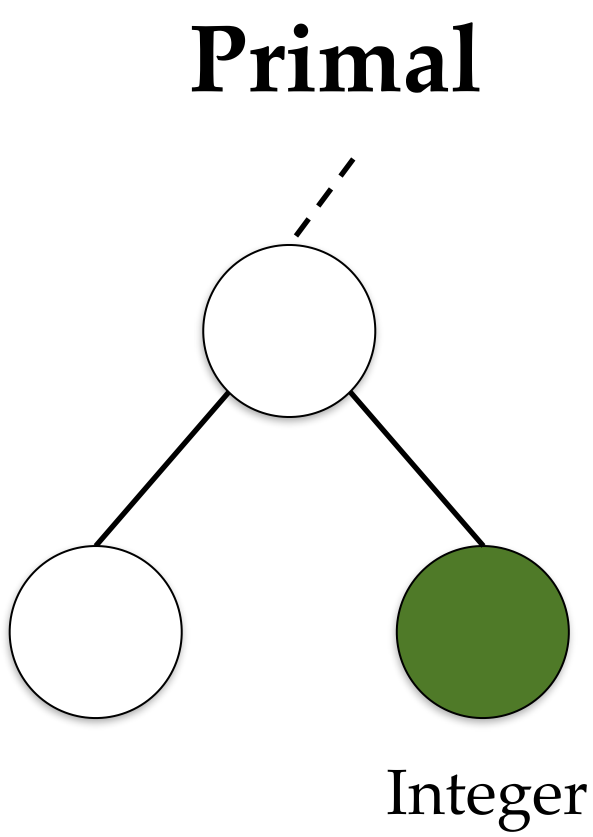
- quantities (primal)
- prices (dual)

If there is no PAB with respect to those prices

- the block selection and the prices form a feasible solution to the Market coupling problem

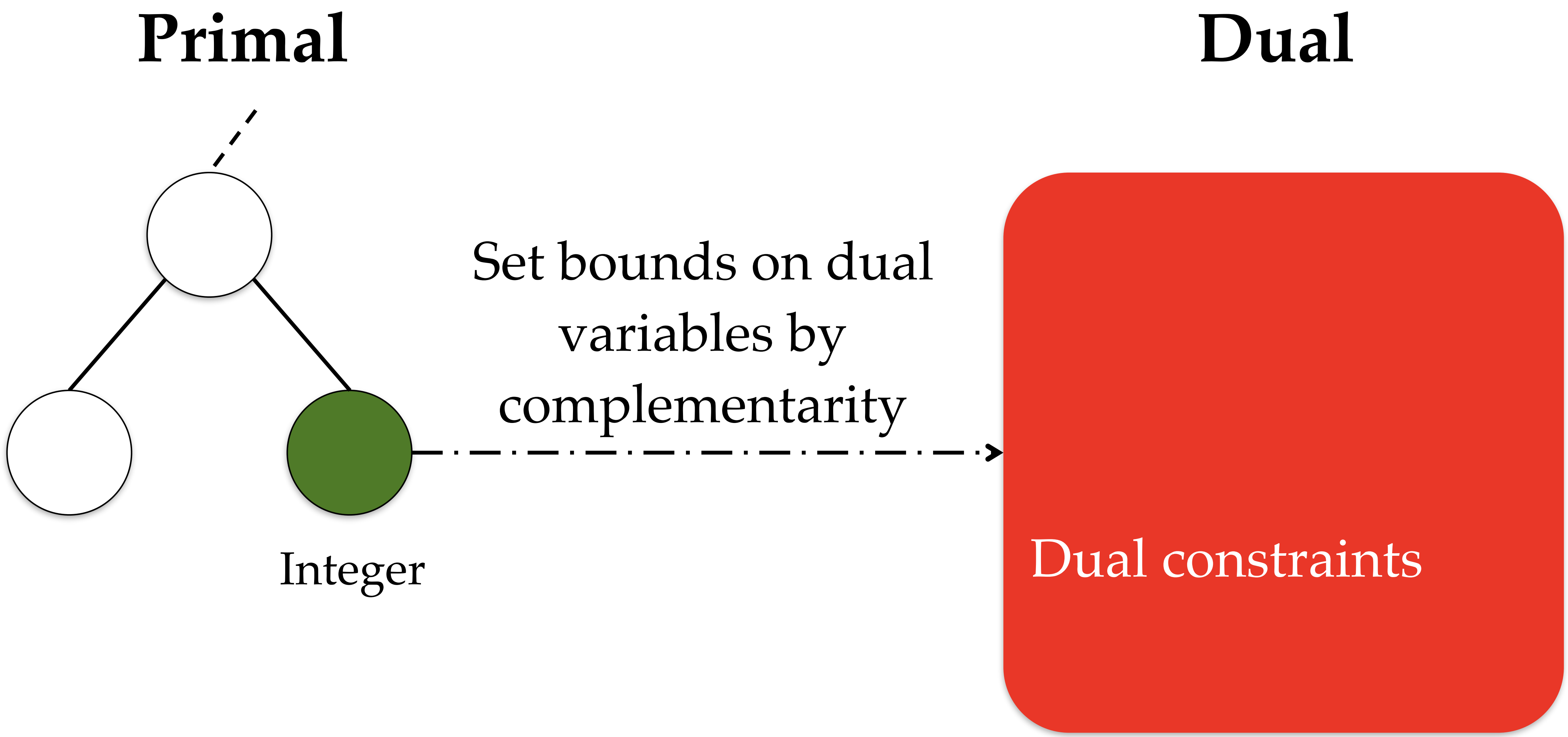
Else we must find another block selection.

Branch-and-cut algorithm description (1)



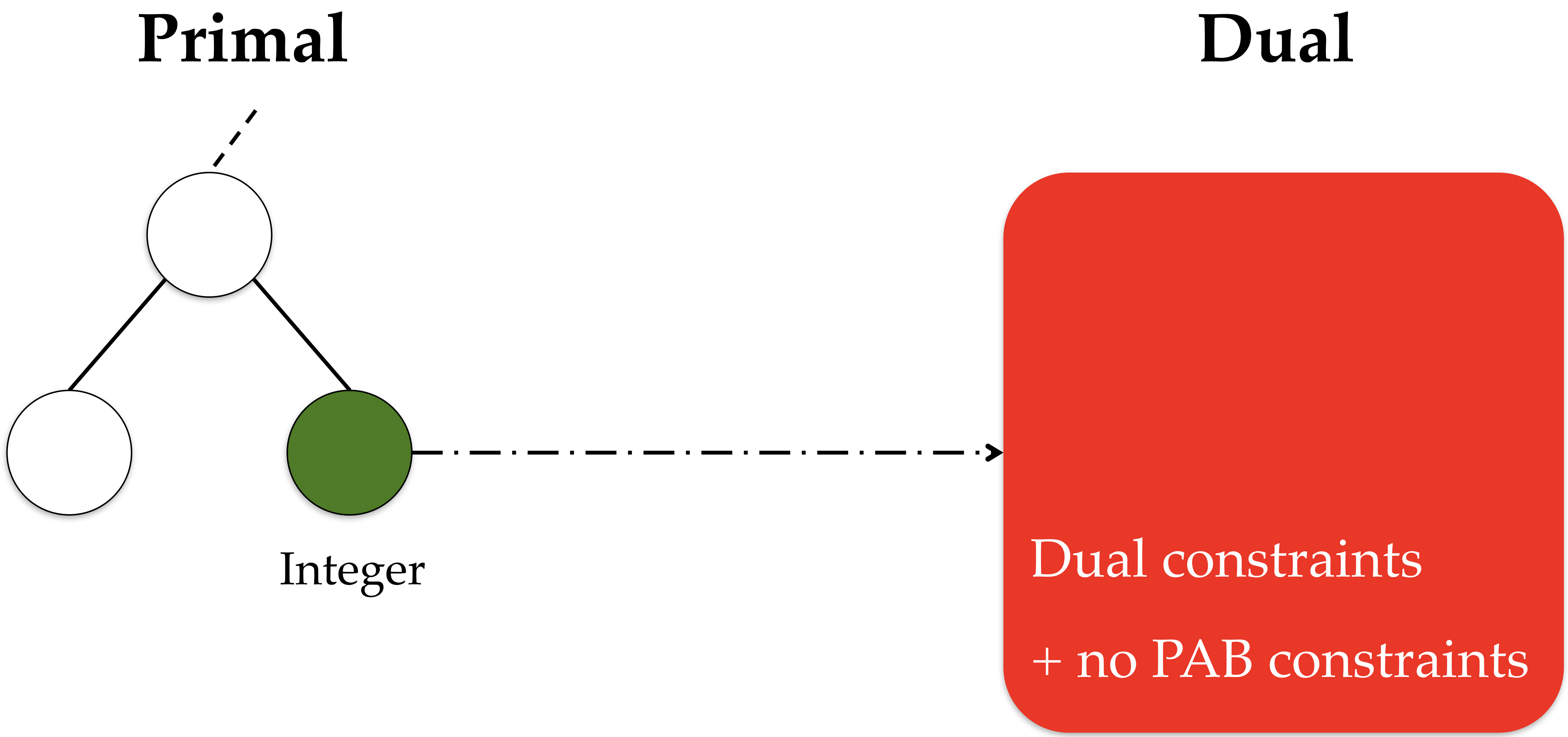
When a node yields an integer solution for the primal

Branch-and-cut algorithm description (2)



The dual of the relaxed problem (integer variables fixed) is constructed from the primal solution by complementarity

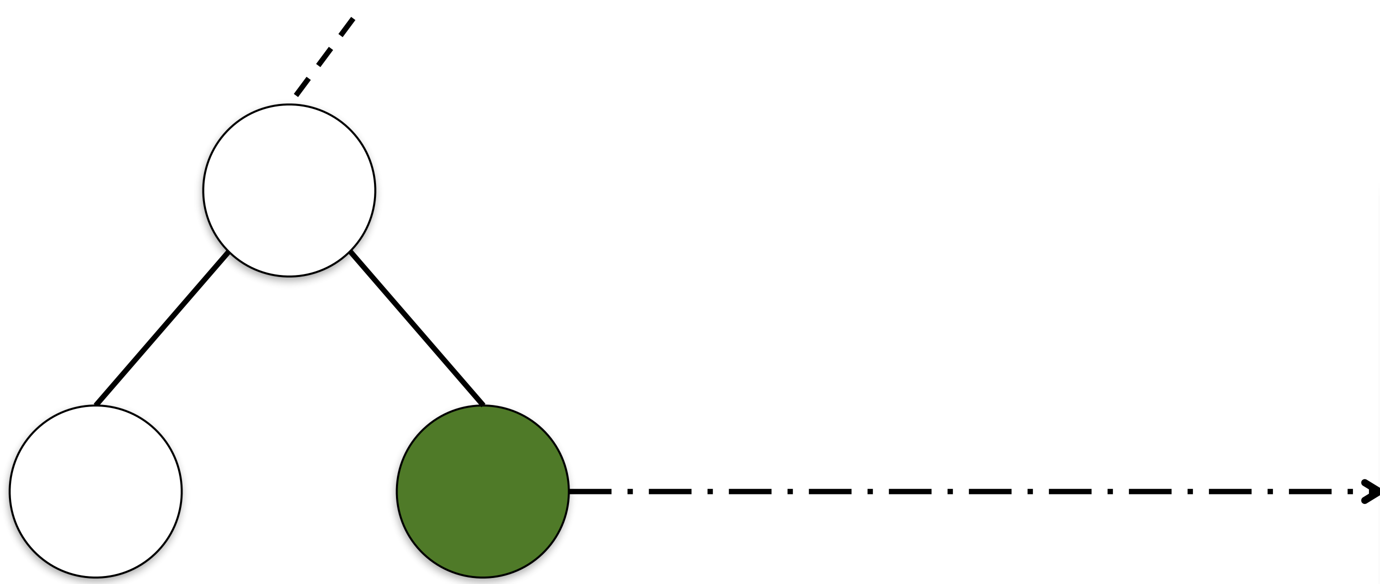
Branch-and-cut algorithm description (3)



A constraint preventing prices that cause PAB is appended to the dual problem

Branch-and-cut algorithm description (4)

Primal | **Dual**



Price problem

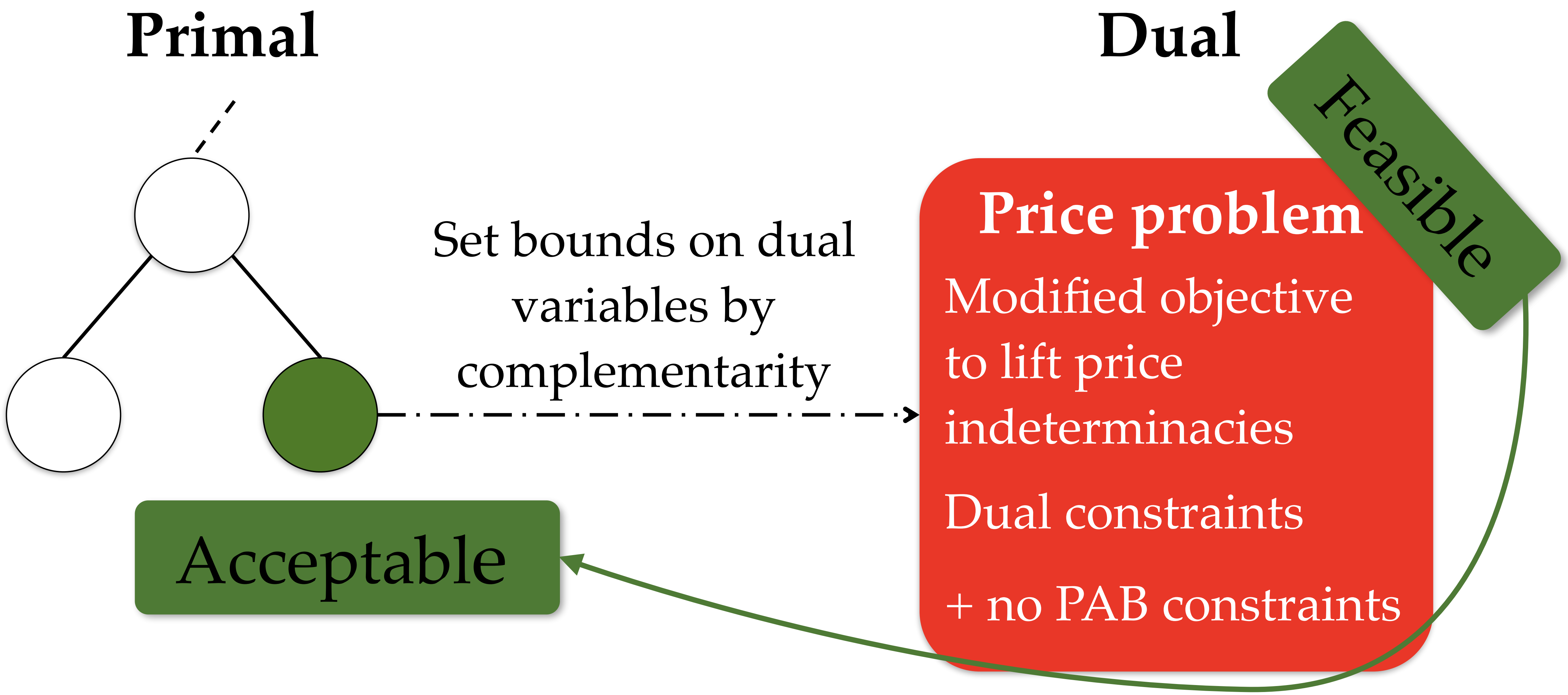
Modified objective to lift price indeterminacies

Dual constraints

+ no PAB constraints

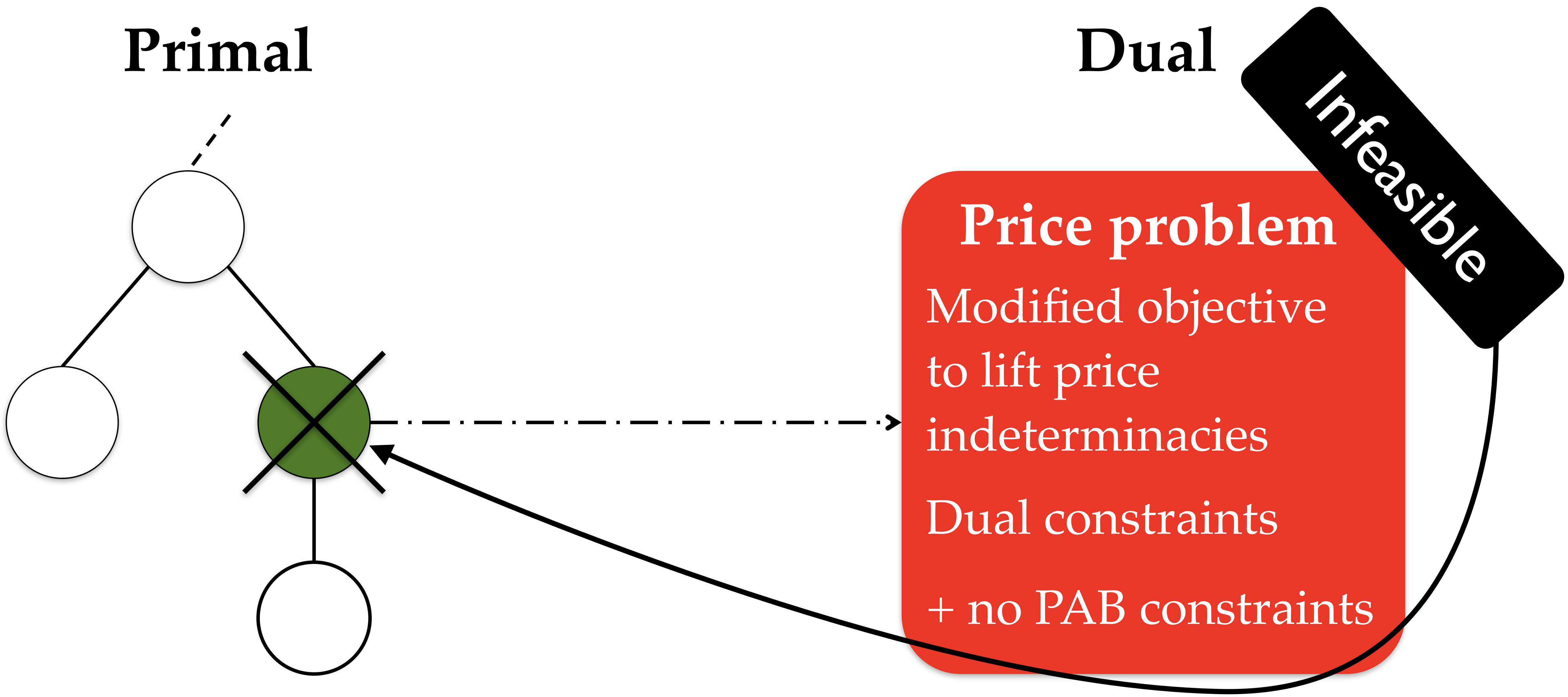
The objective is modified to yield prices as close as possible to the center of the price indeterminacy intervals

Branch-and-cut algorithm description (5)



if that problem is feasible, we have a candidate solution for the market coupling problem

Branch-and-cut algorithm description (6)



Else **a cut** is added to the current node to prevent this block selection

About the implementation in Euphemia

- Implemented
  - in Java
  - using CPLEX and Concert Technology
- tuned cutting and node selection mechanisms
- achieves a precision of 10-5 on all constraints
  - Embedded mechanisms to repair numerically difficult problems

References

[1] EPEX spot annual report 2013

[2] Eurostat: http://appsso.eurostat.ec.europa.eu/nui/setupDownloads.do