PRICE COUPLING OF REGIONS

EUPHEMIA Public Description

PCR Market Coupling Algorithm

Version	0.6	
Date	October 2, 2013	
Status	☐ Draft	☒ Final

Document creation and distribution

Document Owner	EPEX Spot – APX – Belpex – Nord Pool Spot – OMIE – Mercatoelettrico (GME) – OTE
Function	
File location	
Distribution	Publicly available

Previous versions

Version	Date	Author	Summary of changes
0.1	5/10/2013	Epex Spot	Draft creation
0.2	5/29/2013	ALL	First draft
0.6	10/02/2013	ALL	Finalized version

Supported by:

Contents

1. Introduction

Price Coupling of Regions (PCR) project is an initiative of seven Power Exchanges (PXs): APX, Belpex, EPEX SPOT, GME, Nord Pool Spot, OMIE and OTE, covering the electricity markets in Austria, Belgium, Czech Republic, Denmark, Estonia, Finland, France, Germany, Italy, Latvia, Lithuania, Luxembourg, the Netherlands, Norway, Poland, Portugal, Spain, Slovenia, Sweden and the UK.

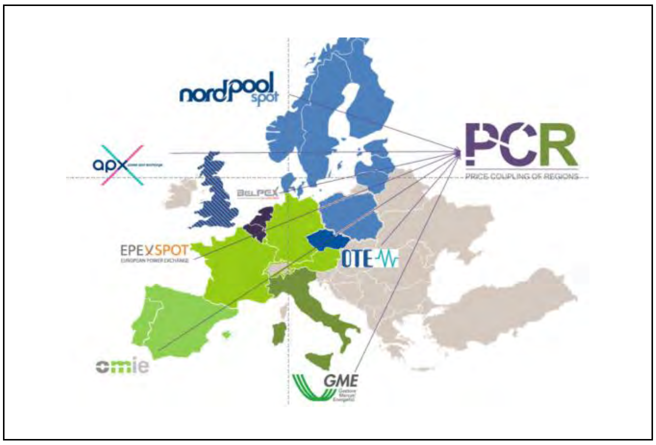


Figure 1 – PXs promoting PCR project

One of the key elements of PCR project is the development of a single price coupling algorithm, which will adopt the name of EUPHEMIA (acronym of Pan-European Hybrid Electricity Market Integration Algorithm). It will be used to calculate energy allocation and electricity prices across Europe, maximizing the overall welfare and increasing the transparency of the computation of prices and flows.

In the past, several algorithms were used locally by the involved PXs. All of them (COSMOS, SESAM, SIOM and UPPO) have been focusing on the features of the corresponding PX, but none was able to cover the whole set of requirements. This made the implementation of the new algorithm (EUPHEMIA) necessary, to cover all the requirements at the same time and give solutions within a reasonable time frame.

2. Day-Ahead Market Coupling Principle

Market coupling (MC) is a way to join and integrate different energy markets into one cross-border market. In a coupled market, demand and supply orders in one market are no longer confined to the local territorial scope. On the contrary, in a market coupling approach energy transactions can involve sellers and buyers from different areas, only restricted by the electricity network constraints.

The main benefit of the Market Coupling approach lies in the improvement of the market liquidity combined with the beneficial side effect of less volatile electricity prices. Market coupling is beneficial for Market players too. They no longer need to acquire transmission capacity rights to carry out cross-border exchanges, since these cross-border exchanges are given as a result of the MC mechanism. They only have to submit a single order in their market (via their corresponding PX) which will be matched with other competitive orders in the same market or other markets (provided the electricity network constraints are respected).

3. Introducing EUPHEMIA

This section introduces the algorithm that has been developed to solve the problem associated with the coupling of the day-ahead power markets in the PCR region: EUPHEMIA.

Market participants submit orders to their respective power exchange. The goal is to decide which orders to execute and which to reject and publish prices such that:

- The *social welfare* (consumer surplus + producer surplus + *congestion rent* across the regions) generated by the executed orders is maximal.
- The power flows induced by the executed orders, resulting in the *net positions* do not exceed the capacity of the relevant network elements.

The EUPHEMIA algorithm handles standard and more sophisticated order types with all their requirements. It aims at rapidly finding a good solution from which it continues to improve and increase the overall welfare. EUPHEMIA is a generic algorithm: there is no hard limit on the number of markets, orders or network constraints; all orders of the same type submitted by the participants are treated equally.

The development of EUPHEMIA started in July 2011 using one of the existing local algorithms COSMOS (being in use in CWE since November 2010) as starting point. The first stable version able to cover the whole PCR scope was internally delivered one year after (July 2012). Since then, the product has been evolving, including both corrective and evolutionary changes.

In the two following chapters, we explain which network models and market products can be handled by EUPHEMIA. Chapter 6 gives a high-level description of how EUPHEMIA works.

4. Power Transmission Network

EUPHEMIA receives information about the power transmission network which is modeled in the form of constraints to be respected in the final solution.

This information will be mainly provided by TSOs as an input to the algorithm.

4.1. Bidding Areas

A *bidding area* is the smallest entity representing a given market where orders can be submitted. EUPHEMIA computes a market clearing price for every *bidding area* per period and a corresponding *net position* (calculated as the difference between the matched supply and the matched demand quantities belonging to that *bidding area*).

Bidding areas can exchange energy between them in an ATC model (Section 4.2), a flow based model (Section 4.3) or a hybrid model (hybrid of the other two).

The *net position* of a *bidding area* can be subject to limitations in the variation between periods.

4.1.1. *Net position* ramping (hourly and daily)

The algorithm supports the limitation on the variations of the *net position* from one hour to the next. There are two ramping requirements on the *net position*.

- Hourly *net position* ramping: this is a limit on the variation of the *net position* of a *bidding area* from each hour to the next.
- Daily (or cumulative) *net position* ramping: this is a limit on the amount of reserve capacity used during the day.

Reserve capacity is needed as soon as the variation of the *net position* from one hour to the next exceeds a certain threshold. There is a fixed limit on the total amount of reserve that can be used during the day. Reserve capacity is defined separately for each direction (increase/decrease).
By including the *net position* of the last hour for the previous (delivery) day, overnight ramping can be taken into account.

4.2. ATC Model

In an ATC model, the *bidding areas* are linked by interconnectors (lines) representing a given topology. The energy from one *bidding area* to another can only flow through these lines and is limited by the available transfer capacity (ATC) (Section 4.2.1) of the line.

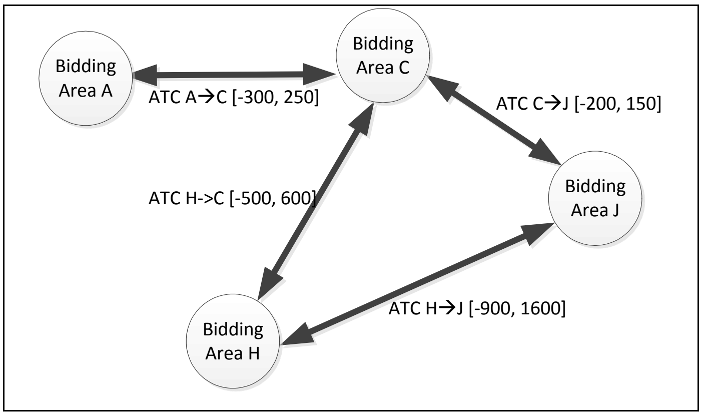


Figure 2 – *Bidding areas* connected in ATC model

Additional restrictions may apply to interconnectors:

- The flow on a line can be subject to losses (Section 4.2.2)
- The flow on a line can be subject to tariffs (Section 4.2.3)
- The flow variation between two consecutive hours can be restricted by an hourly flow ramping limit (Sections 4.2.4 and 4.2.5)

4.2.1. Available Transfer Capacity (ATC)

ATC limitations constrain the flow that passes through the interconnectors of a given topology.

In EUPHEMIA, lines are oriented from a source *bidding area* (A) to a sink *bidding area* (C). Thus, in the examples hereafter, a positive value of flow on the line indicates a flow from A to C, whereas a negative value indicates a flow from C to A.

The available transfer capacity of a line can be different per period and directions of the line (Figure 2).

- As an example, let us consider two *bidding areas* A and C connected by a single line defined from A to C (A→C). For a given period, the ATC in the direction (A→C) is assumed to be equal to 250 MW and equal to 300 MW in the opposite direction (C→A). In practice, this implies that the valid value for the algebraic flow on this line in this period shall remain in the interval [-300, 250].

ATC limitations can also be negative. A negative ATC value in the same direction of the definition of the line A→C (respectively, in the opposite direction C→A) is implicitly indicating that the flow is forced to only go in the direction C→A (respectively, A→C).

- In the previous example, if the ATC was defined to be equal to -250 MW instead of 250 MW in the direction A→C then this would imply that the valid value for the flow will now be in the interval [-300, -250], forcing the flow to be in the C→A direction (negative values of the flow on a line defined as A→C).

4.2.2. Losses

Flow on a line between *bidding areas* may be subject to losses. In this case, part of the energy that is injected in one side of the line is lost, and the energy received at the end of the cable is less than the energy initially sent (Figure 3).

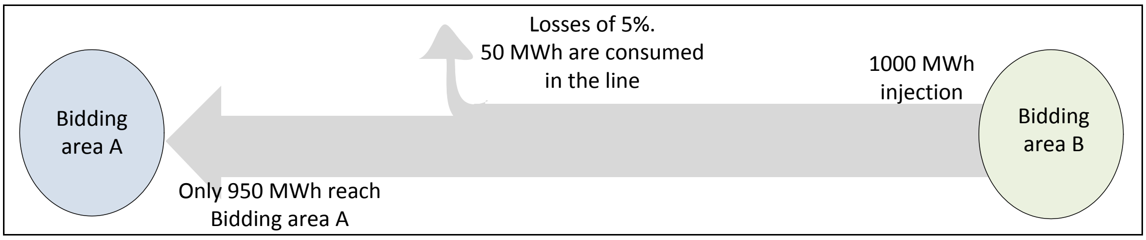


Figure 3 – Example of the effect of losses in one line.

4.2.3. Tariffs

In an ATC network model, the DC cables might be operated by merchant companies, who levy the cost incurred for each passing MWh in the cable. In the algorithm, these costs can be represented as flow tariffs.

The flow tariff is included as a loss with regard to the *congestion rent*. This will show in the results as a threshold for the price difference. If the difference between the two corresponding market clearing prices is less than the tariff then the flow will be zero. If there is a flow the price difference will be exactly the flow tariff, unless there is congestion. Once the price difference exceeds the threshold the *congestion rent* becomes positive.

4.2.4. Hourly Flow Ramping Limit on Individual Lines

The hourly variation of the flows over an interconnector can be constrained by a ramping limit. This limitation confines the flow in an "allowed band" between the hours when moving from one hour to the next (Figure 4). The ramping limit constrains the flow that can pass through the line in hour h depending on the flow that is passing in hour h-1.

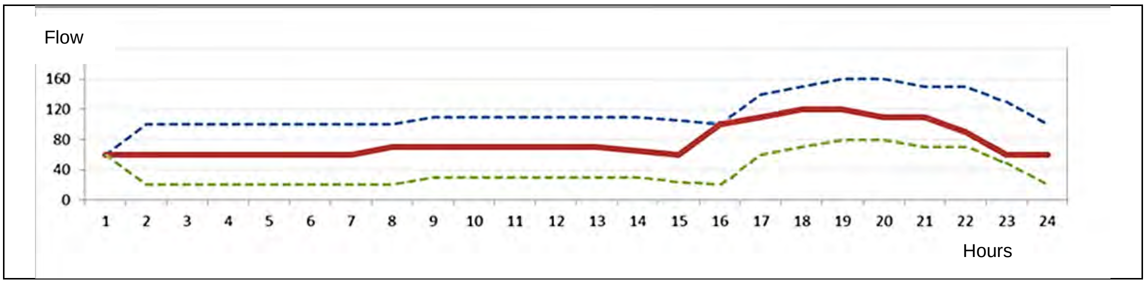


Figure 4 – Effect of the hourly flow ramping limit. The flow stays in the allowed band between hours.

The ramping limit is defined by: The maximum increment of flow from hour h-1 to hour h (called ramping-up), and the maximum decrement of flow from hour h-1 to hour h (called ramping-down). The ramping limit may be different for each period. For period 1, the limitation of flow takes into account the value of the flow of the last hour of the previous day.

4.2.5. Hourly Flow Ramping Limit on Line Sets

Flow ramping constraints can apply to a group of interconnectors at once, i.e. the sum of the flows over a set of lines can be restricted by ramping limits.

4.3. Flow Based Model

The Flow Based (FB) model is an alternative to ATC network constraints. Modeling network constraints using the flow based model allows a more precise modeling of the physical flows.

The FB constraints are given by means of two components:

- **Remaining Available Margin (RAM):** number of MW available for exchanges
- **Power Transfer Distribution Factor (PTDF):** ratio which indicates how much MWh are used by the *net position*s resulting from the exchanges

PTDFs can model different network constraints that constrain the exchanges allowed. Each constraint corresponds to a single row in the $PTDF$ matrix, and has one corresponding margin (one value of the RAM vector). The $PTDF$ matrix has columns for each hub where it applies to (e.g. FB in CWE has columns for the *net position*s of all CWE hubs: BE, DE, FR and NL).

Therefore the constraint that is being imposed is the following:

$$PTDF \cdot nex \leq RAM$$

Here nex is the vector of *net position*s which are subject to the flow based constraints. The flow based modeling has some consequences to price formation, and can potentially result in "non-intuitive" situations that happen when the energy goes from high priced areas to low priced areas.

Example:
Consider a three market example (Figure 5), with a single PTDF constraint:

$$0.25 \cdot nex_A - 0.5 \cdot nex_B - 0.25 \cdot nex_C \leq 125$$

And consider the market outcome shown in Figure 5 below.

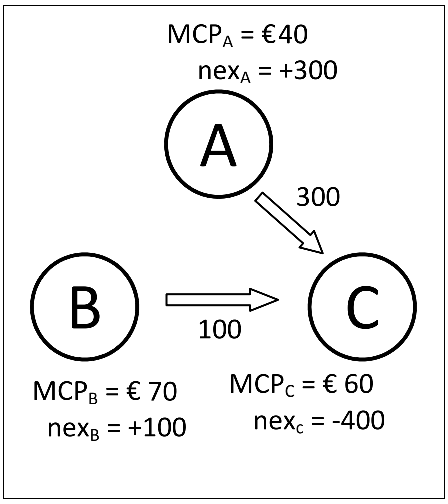


Figure 5 – Example of *net position*s decompositions into flows

In the representation of the result, "bilateral exchanges" between *bidding areas* have been indicated. This is merely one potential decomposition of *net position*s into flows out of many. Alternative flows could have been reconstructed too. However since market B is exporting energy, whereas it is the most expensive market, any breakdown into flows shall result in market B exporting energy to a cheaper market.

Intuitiveness
From the example above we see that FB market coupling can lead to non-intuitive situations. The reason is that some non-intuitive exchanges free up capacity, allowing even larger exchanges between other markets. In our example, exporting from B to C loads the critical branch with (-0.5) – (-0.25) = -0.25 MWh for each MWh exchanged, i.e. it actually relieves the line. Welfare maximization can therefore lead to these non-intuitive situations.

EUPHEMIA integrates a mechanism to suppress these non-intuitive exchanges. This mechanism seeks "flows" between areas which match the *net position*s. Rather than imposing the PTDF constraints directly on the *net position*s, in intuitive mode they are applied to these "flows". So far the two models are fully equivalent. However in case a PTDF constraint is detected that leads to a non-intuitive situation, all of its relieving effects are discarded: the impact of a “flow” from i to j actually is $PTDF_i - PTDF_j$, but is replaced by $max(PTDF_i\text{-}PTDF_j, 0)$.

5. Market Orders

The algorithm can handle a large variety of order types at the same time, which are available to the market participants in accordance with the local market rules:

- Aggregated Hourly Orders
- Complex Orders
    - MIC orders
    - Load Gradient orders
- Block Orders
    - Profiled Block Orders
    - Linked Block Orders
    - Exclusive Groups of Block Orders
    - Flexible Hourly Orders
- Merit Orders and PUN Orders.

5.1. Aggregated Hourly Orders

Demand (resp. supply) orders from all market participants belonging to the same *bidding area* will be aggregated into a single curve referred to as aggregated demand (resp. supply) curve defined for each period of the day. Demand orders are sorted from the highest price to the lowest. Conversely, supply orders are sorted from the lowest to the highest price.

Aggregated supply and demand curves can be of the following types:

- Linear piecewise curves (i.e. two consecutive points of the monotonous curve cannot have the same price, except for the first two points defined at the maximum / minimum prices of the *bidding area*).

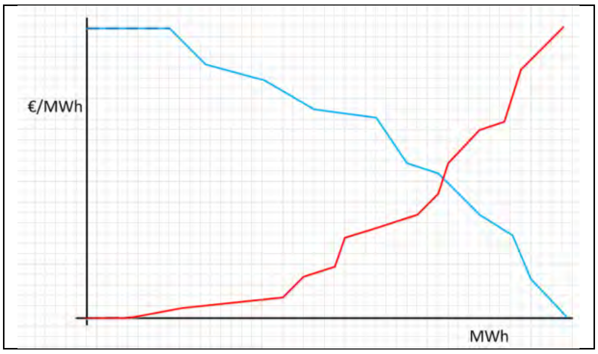


Figure 6 – Linear piecewise aggregated curve.

- Stepwise curves (i.e. two consecutive points always have either the same price or the same quantity).

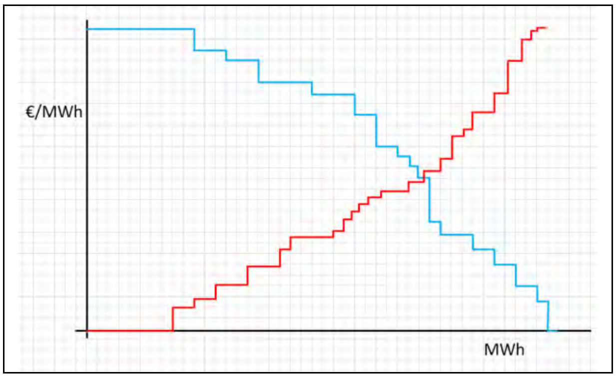


Figure 7 – Stepwise aggregated curve.

- Hybrid curves (composed by both linear and stepwise segments).

The following nomenclature is used when speaking about hourly orders[1] and market clearing prices:

- One demand (resp. supply) hourly order is said to be *in-the-money* when the market clearing price is lower (resp. higher) than the price of the hourly order.
- One demand or supply hourly order is said to be *at-the-money* when the price of the hourly order is equal to the market clearing price.
- One demand (resp. supply) hourly order is said to be *out-of-the-money* when the market clearing price is higher (resp. lower) than the price of the hourly order.
- For linear piecewise hourly orders starting at price p_0 and finishing at price p_1, p_0 is used as the order price for the nomenclature above

[1] Whenever hourly orders are mentioned through this document, we are referring to the aggregated hourly orders that are the input of EUPHEMIA.

(except for energy *at-the-money*, where the market clearing price is in the interval [p_0, p_1]).

The rules that apply for the acceptance of hourly orders in the algorithm are the following:

- Any order in-the-money must be fully accepted.
- Any order out-of-the money must be rejected.
- Orders at-the-money can be accepted (fully or partially) or rejected.

Price-taking orders, defined at the maximum / minimum prices of the *bidding area*, have additional requirements which are detailed in Section 6.5.1.

5.2. Complex Orders

A complex order is a set of simple supply stepwise hourly orders (which are referred to as hourly sub-orders) belonging to a single market participant, spreading out along different periods and are subject to a complex condition that affects the set of hourly sub-orders as a whole.

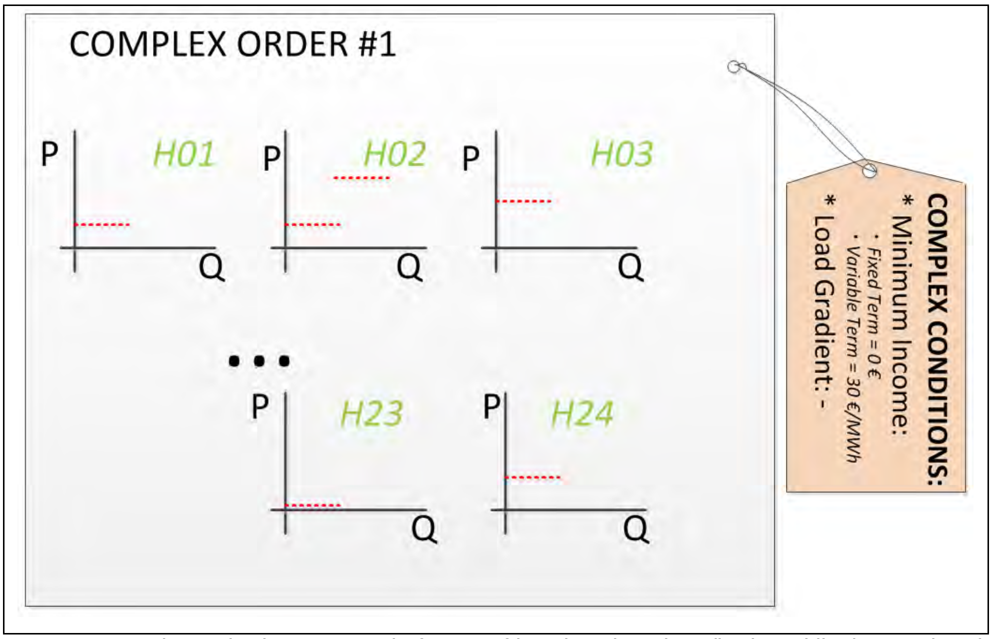


Figure 8 – A complex order is composed of a set of hourly sub-orders (in dotted line) associated with complex conditions

Complex conditions are of two types: Minimum Income (with or without scheduled stop), and Load Gradient.

5.2.1. Minimum Income Condition (MIC)

Complex orders (with their set of hourly sub-orders) subject to Minimum Income Condition constraints are called MIC orders (or MICs).

Generally speaking, the Minimum Income economical constraint means that the amount of money collected by the order in all periods must cover its production costs, which is defined by a fix term (representing the startup cost of a power plant) and a variable term multiplied by the total assigned energy (representing the operation cost per MWh of a power plant).

The Minimum Income Condition constraint is in short defined by:

- A fix term (FT) in Euros
- A variable term (VT) in Euros per accepted MWh.

In the final solution, MIC orders are activated or deactivated (as a whole):

- In case a MIC order is activated, each of the hourly sub-orders of the MIC behaves like any other hourly order, which means accepted if they are in-the-money and rejected if they are out-of-the-money.
- In case a MIC order is deactivated, each of the hourly sub-orders of the MIC is fully rejected, even if it is in-the-money (with the exception of scheduled stop, see Section 5.2.2).

The final solution given by EUPHEMIA will not contain active MIC orders not fulfilling their Minimum Income Condition constraint (also known as paradoxically accepted MICs).

5.2.2. Scheduled Stop

In case the owner of a power plant which was running the previous day offers a MIC order to the market, he may not want to have the production unit stopped abruptly in case the MIC is deactivated.

For the avoidance of this situation, the sender of a MIC has the possibility to define a "scheduled stop". Using a schedule stop will alter the deactivation of the MIC: the deactivation will not imply the automatic rejection of all the hourly sub-orders. On the contrary, the first (i.e. the cheapest) hourly sub-order in the periods that contain scheduled stop (up to period 3) will not be rejected but will be treated as any hourly order.

5.2.3. Load Gradient

Complex orders (with their set of hourly sub-orders) on which a Load Gradient constraint applies are called Load Gradient Orders.

Generally speaking, the Load Gradient constraint means that the amount of energy that is matched by the hourly sub-orders belonging to a Load Gradient order in one period is limited by the amount of energy that was matched by the hourly sub-orders in the previous period. There is a maximum increment / decrement allowed (the same value for all periods). Period 1 is not constrained.

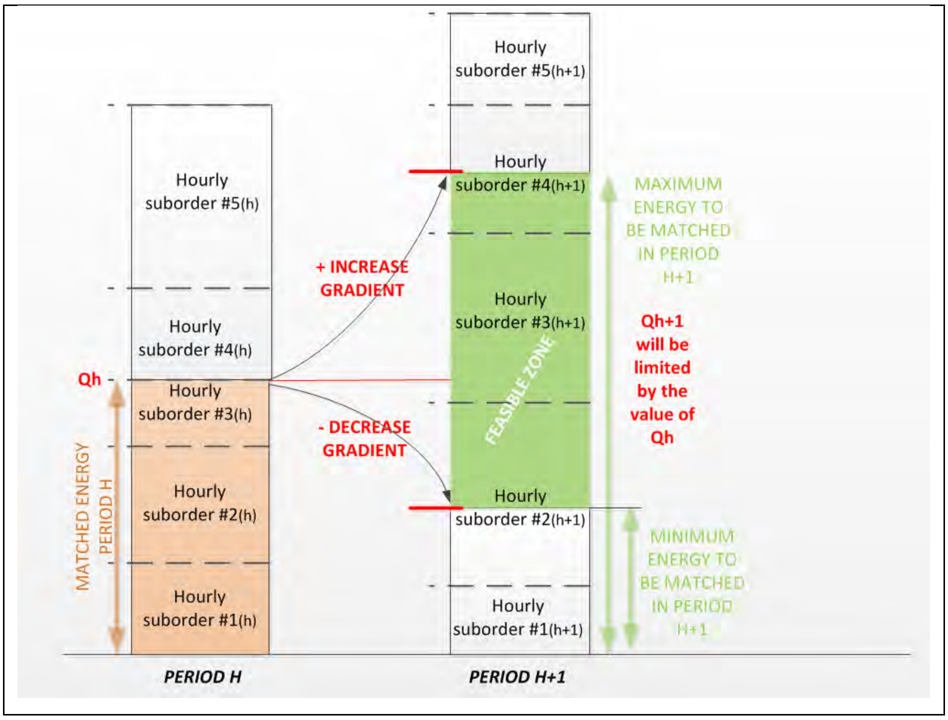


Figure 9 – A Load Gradient order. Effect produced by the amount that is matched in period (h) on period (h+1).

5.2.4. Complex orders combining Load Gradient and MIC

Complex orders (with their set of hourly sub-orders) can be subject to both load gradient and minimum income condition (with or without scheduled stop).

5.3. Block Orders

A block order can be a supply or demand order and is defined by:

- a fixed price limit (minimum price for supply block and maximum price for demand blocks),
- a number of periods,
- a volume that can be different for every period,
- the minimum acceptance ratio.

In the simplest case, a block is defined for a consecutive set of periods with the same volume for all of them and with a minimum acceptance ratio of 1 (regular fill-or-kill block orders). These are usually called regular block orders and are the type of blocks that is more frequently used. However, in general, the periods of the blocks can be non-consecutive, the volume can differ between periods and the minimum acceptance ratio can be less than 1 (partial acceptance).

5.3.1. Profiled Block Orders

A profile block order is a regular block order where the volume is allowed to differ in each period over the entire time span of the block.
Example of a (supply) Profile Block Order:

Block Order #1

- Price: 40 €/MWh
- Minimum acceptance ratio: 0.5
- Intervals: Hours (3-7), hours (8-19) and hours (22-24)
- Volume: 80 MWh in the first interval, 220 MWh in the second one, and 40 MWh in the third one.

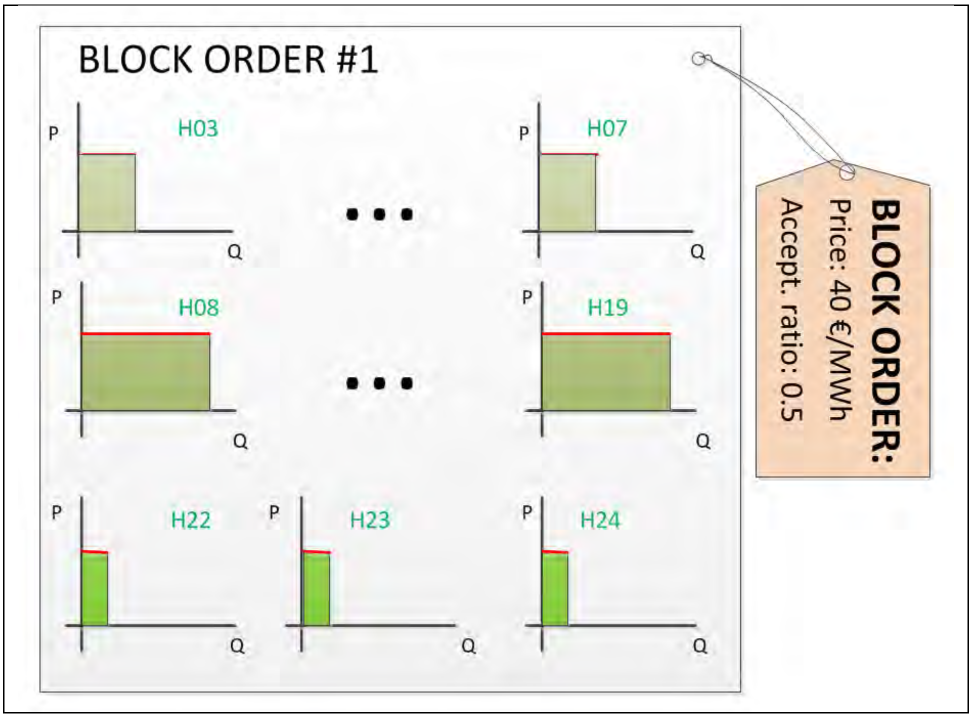


Figure 10 – Profile block order example

Acceptance of the supply block orders:

- if the block volume weighted average market clearing price for the periods during which the block is defined is above the price of the block, then the block can be entirely accepted, which means that all the energy in the block is accepted;
- if the block volume weighted average market clearing price for the periods during which the block is defined is below the price of the block, then the block must be entirely rejected;
- if the block volume weighted average market clearing price for the periods during which the block is defined is exactly the price of the block, then the Block can be either fully rejected, fully accepted or partially accepted, to the extent that the ratio "accepted volume/total submitted volume" is greater than or equal to the minimum acceptance ratio of the block (e.g. 0.5) and equal over all periods.

For demand blocks, the rules are symmetrical (above←→below).

5.3.2. Linked Block Orders

Block orders can be linked together, i.e. the acceptance of individual block orders can be made dependent on the acceptance of other block orders. The block which acceptance depends on the acceptance of another block is called "child block", whereas the block which conditions the acceptance of other blocks is called "parent block".

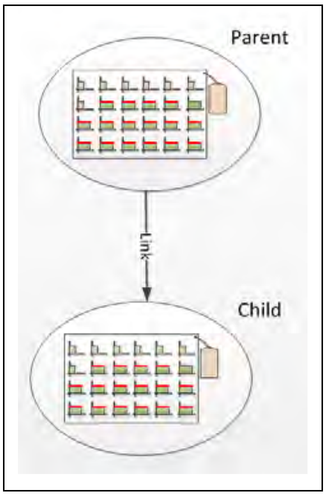


Figure 11 – Linked block orders

The rules for the acceptance of linked block orders are the following:

1. The acceptance ratio of a parent block is greater than or equal to the acceptance ratio of its child blocks
2. (Possibly partial) acceptance of child blocks can allow the acceptance of the parent block when:
   a. the surplus of a family is non-negative
   b. leaf blocks (block order without child blocks) do not generate welfare loss
3. A parent block which is *out-of-the-money* can be accepted in case its accepted child blocks provide sufficient surplus to at least compensate the loss of the parent.
4. A partially accepted child block must be *at-the-money* if it has no child blocks that are accepted.
5. A child block which is *out-of-the-money* cannot be accepted even if its accepted parent provides sufficient surplus to compensate the loss of the child, unless the child block is in turn parent of other blocks (in which case rule 3 applies).

In an easy common configuration of two linked blocks, the rules are easy. The parent can be accepted alone, but not the child that always needs the acceptance of the parent first. The child can "save" the parent with its surplus, but not the opposite.

5.3.3. Block Orders in an Exclusive group

An Exclusive group is a set of block orders for which the sum of the accepted ratios cannot exceed 1. In the particular case of blocks that have a minimum acceptance ratio of 1 it means that at most one of the blocks of the exclusive group can be accepted.

Between the different valid combinations of accepted blocks the algorithm chooses the one which maximizes the optimization criterion (*social welfare*, see Section 6.2).

5.3.4. Flexible Hourly Orders

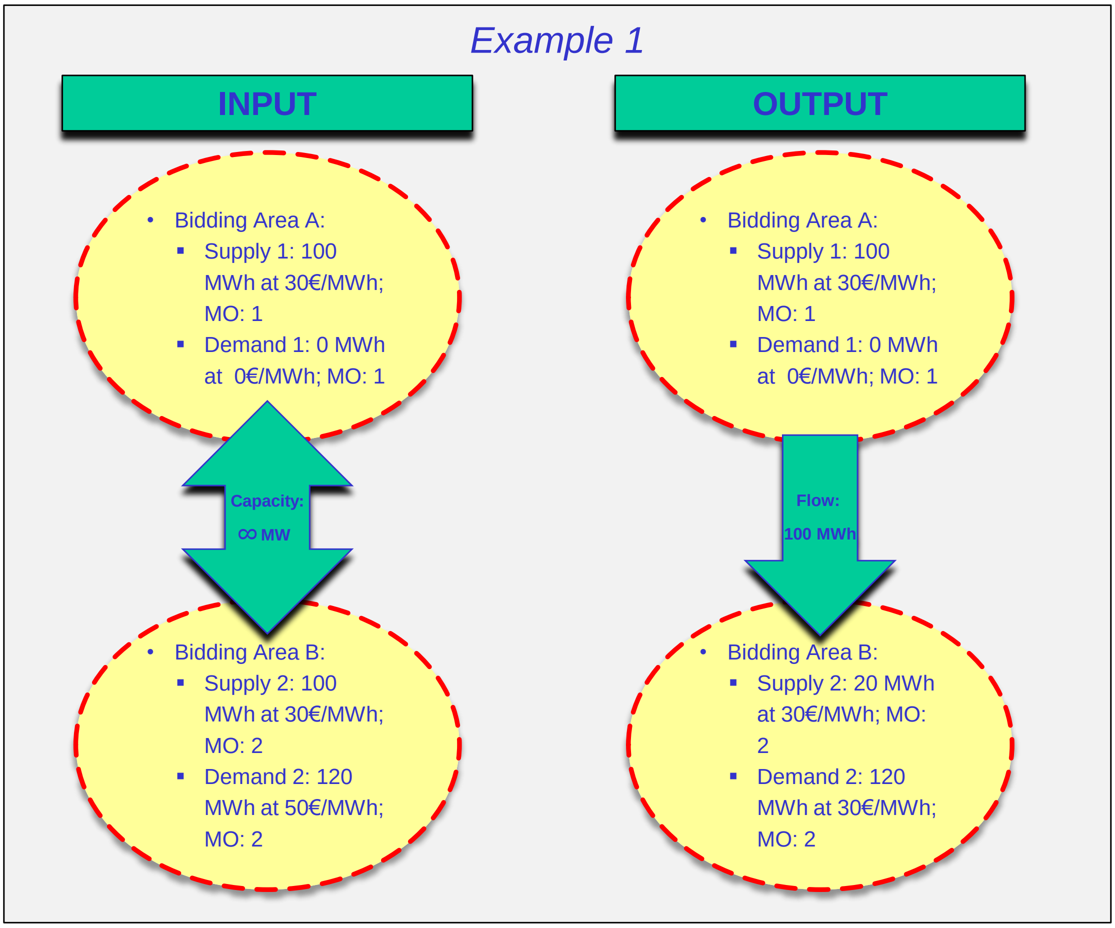
A flexible "hourly" order is a block order with a fixed price limit, a fixed volume, minimum acceptance ratio of 1, with duration of 1 hour. The hour is not defined by the participant but will be determined by the algorithm (hence the name "flexible"). The hour in which the flexible hourly order is accepted, is calculated by the algorithm and determined by the optimization criterion (see Section 6.2)

5.4. Merit Orders and PUN Orders

5.4.1. Merit Orders

Merit orders are individual step orders defined at a given period for which is associated a so-called merit order number.

A merit order number is unique per period and order type (Demand; Supply; PUN) and is used for ranking merit orders in the *bidding areas* containing this order type. The lower the merit order number, the higher the priority for acceptance. More precisely, when, within an uncongested set of adjacent *bidding areas*, several merit orders have a price that is equal to the market clearing price, the merit order with the lowest merit order number should be accepted first unless constrained by other network conditions.


Example 1
INPUT
OUTPUT
Bidding Area A:
Supply 1: 100 MWh at 30€/MWh; MO: 1
Demand 1: 0 MWh at 0€/MWh; MO: 1
Capacity: ∞ MW
Bidding Area B:
Supply 2: 100 MWh at 30€/MWh; MO: 2
Demand 2: 120 MWh at 50€/MWh; MO: 2
Bidding Area A:
Supply 1: 100 MWh at 30€/MWh; MO: 1
Demand 1: 0 MWh at 0€/MWh; MO: 1
Flow: 100 MWh
Bidding Area B:
Supply 2: 20 MWh at 30€/MWh; MO: 2
Demand 2: 120 MWh at 30€/MWh; MO: 2

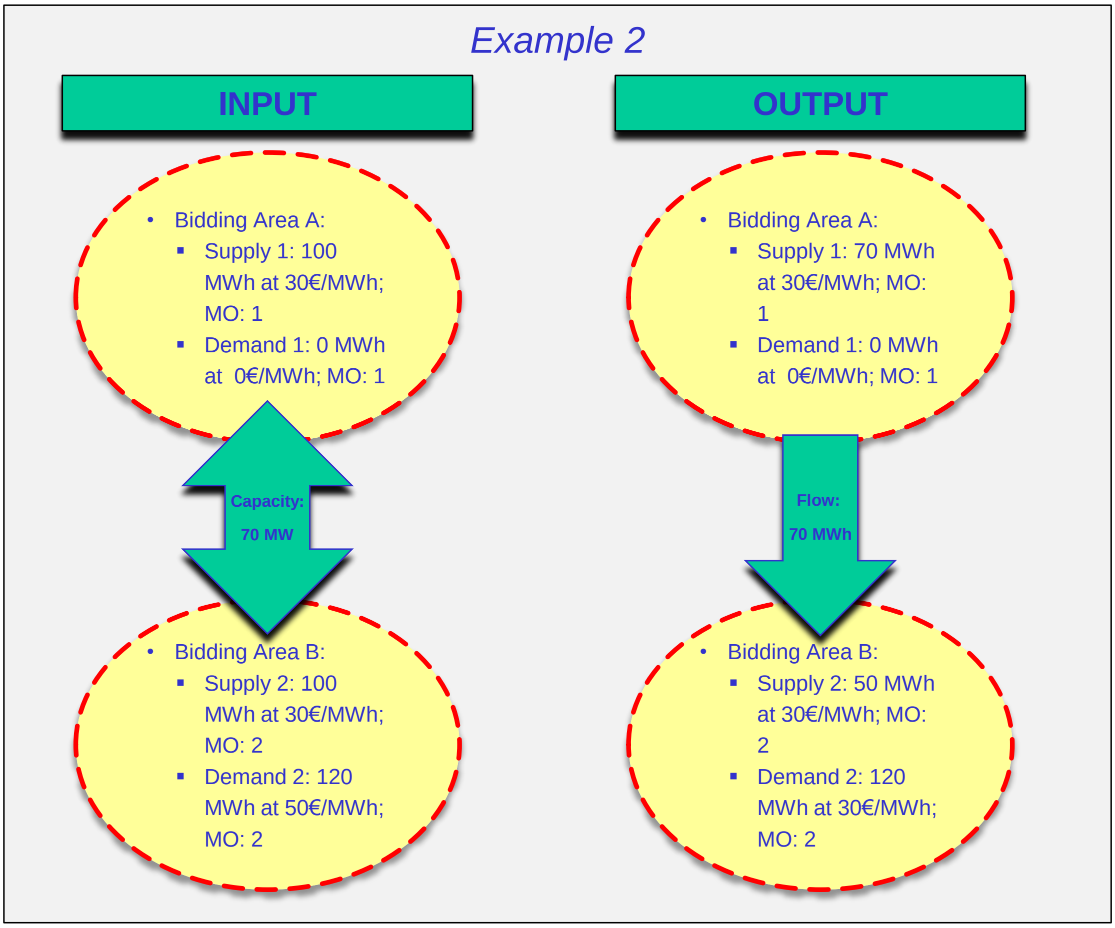


Figure 12: Merit Orders examples

5.4.2. PUN Orders

PUN orders are a particular type of demand merit orders. They differ from classical demand merit orders in such sense that they are cleared at the *PUN price* (PUN stands for "Prezzo Unico Nazionale") rather than the *bidding area* market clearing price (i.e. a PUN order with an offered price lower than market clearing price of its associated *bidding area*, but higher than *PUN price* would be fully accepted by EUPHEMIA).

For each period, the values of the accepted PUN merit orders volumes multiplied by the *PUN price* is equal to the value of the accepted PUN merit orders volumes multiplied by the corresponding *market clearing prices* (up to a defined tolerance named PUN imbalance[2]), according to the following Formula:

[2] In other words, the value (PUN Volume * *PUN price*) must be able to refund producers (who receives the price of their bidding area), congestion rents and a PUN imbalance.

$P_{PUN} \times \Sigma_z Q_z = \Sigma_z P_z \times Q_z \pm \Delta$

With:

- P_{PUN}: *PUN price*
- Q_z: Volumes consumed in *bidding area* z
- P_z: Price of *bidding area* z
- Δ: PUN imbalance

In case of more than one PUN order submitted at a price equal to *PUN price*, the merit order number rule is applied to PUN orders as well.

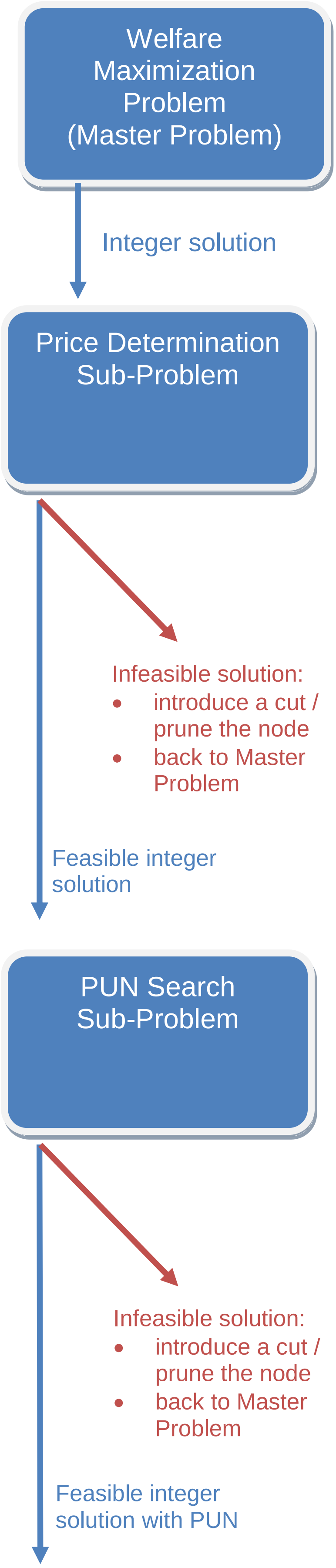
6. EUPHEMIA Algorithm

6.1. Overview

As mentioned previously, EUPHEMIA is the algorithm that has been developed to solve the Day-Ahead European Market Coupling problem. EUPHEMIA matches energy demand and supply for all the periods of a single day at once while taking into account the market and network constraints. The main objective of EUPHEMIA is to maximize the *social welfare*, *i.e.* the total market value of the Day-Ahead auction expressed as a function of the *consumer surplus*, the *supplier surplus*, and the *congestion rent* including tariff rates on interconnectors if they are present. EUPHEMIA returns the *market clearing prices*, the matched volumes, and the *net position* of each *bidding area* as well as the flow through the interconnectors. It also returns the selection of block, complex, merit, and PUN orders that will be executed.

By ignoring the particular requirements of the block, complex, merit and PUN orders, the market coupling problem resolves into a much simpler problem which can be modeled as a Quadratic Program (QP) and solved using commercial off-the-shelf solvers. However, the presence of these orders renders the problem more complex. Indeed, the "kill-or-fill" property of block orders and the minimum income condition (MIC) of complex orders require the introduction of binary (i.e. 0/1) variables. Moreover, the strict consecutiveness requirement of merit and PUN orders adds up to the complexity of the problem.

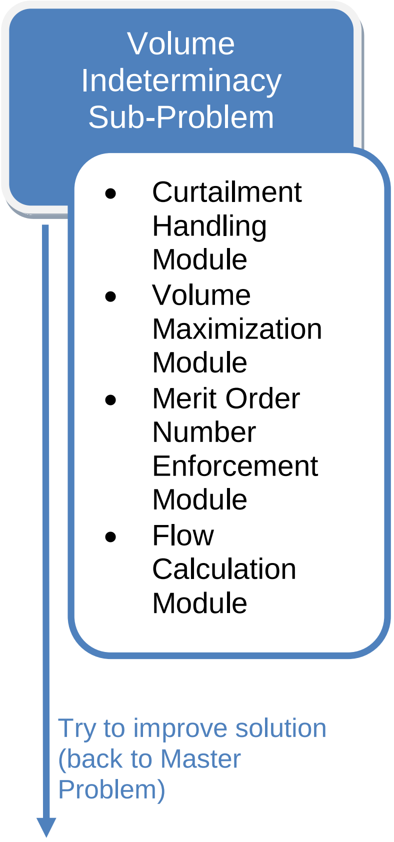
In order to solve this problem, EUPHEMIA runs a combinatorial optimization process based on the modeling of the market coupling problem. The reader can refer to the Annex B for a more detailed mathematical formulation of the problem. EUPHEMIA aims to solve a welfare maximization problem (also referred to as the master problem) and three interdependent sub-problems namely, the price determination sub-problem, the PUN search sub-problem and the volume indeterminacy sub-problem.



In the welfare maximization problem, Euphemia searches among the set of solutions (solution space) for a good selection of block and MIC orders that maximizes the *social welfare*. In this problem, the PUN and merit orders requirements are not enforced. Once an integer solution has been found for this problem, Euphemia moves on to determine the *market clearing prices*.

The objective of the price determination sub-problem is to determine, for each *bidding area*, the appropriate *market clearing price* while ensuring that no block and complex MIC orders are *paradoxically accepted* and that the flows meet their requirements. If a feasible solution could be found for the price determination sub-problem, Euphemia proceeds with the PUN search sub-problem. However, if the sub-problem does not have any solution, we can conclude that the block and complex orders selection is not acceptable, and the integer solution to the welfare maximization problem must be rejected. This is achieved by adding a cut to the welfare maximization problem that renders its current solution infeasible. Subsequently, Euphemia resumes the welfare maximization problem searching for a new integer solution for the problem.

The objective of the PUN search sub-problem is to find valid PUN volumes and prices for each period of the day while satisfying the PUN imbalance constraint and enforcing the strong consecutiveness of accepted PUN orders. When the PUN search sub-problem is completed, Euphemia verifies that the obtained PUN solution does not introduce any *paradoxically accepted block/complex orders*. If some orders become *paradoxically accepted*, a new cut is introduced to the welfare maximization problem that renders the current solution infeasible. Otherwise, Euphemia proceeds with the lifting of volume indeterminacies.



In the previous sub-problems, the algorithm has determined the *market clearing prices* for each *bidding area*, the *PUN price*s and volumes for the area with PUN orders, and a selection of block and complex MIC orders that are feasible all together. Though, there might exist several aggregated hourly volumes, *net position*s, and flows that are coherent with these prices and that yield the same welfare. Among all these possible solutions, EUPHEMIA pays special attention to the *price-taking orders*, enforces the merit order number, and maximizes the traded volume.

6.2. Welfare Maximization Problem (Master Problem)

As mentioned previously, the objective of this problem is to maximize the *social welfare*, *i.e.* the total market value of the Day-Ahead auction. The *social welfare* is computed as the sum of the *consumer surplus*, the *supplier surplus*, and the *congestion rent*. The latter takes into account the presence of tariff rates for the flows through defined interconnectors.

EUPHEMIA ensures that the returned results are coherent with the following constraints (see Chapters 4 and 5):

- The aggregated hourly demand and supply curves and merit orders
- The fill-or-kill requirement of block orders
- The scheduled stop, load gradient, and minimum income condition of complex orders
- The capacities and ramping constraints imposed on the ATC interconnectors while taking into account the losses and the tariff rates if applicable.
- The flow limitation through some critical elements of the network for *bidding areas* managed by the flow-based network model.

- The sum of the net export position of all the *bidding areas* must be equal to zero while respecting the hourly and daily ramping constraints applied to these net export positions.

It should be noted that the strict consecutiveness requirement of merit and PUN orders is not enforced in this problem. In other words, the merit orders are considered in this problem as aggregated hourly orders while, the PUN orders are just ignored. The main difficulty of the welfare maximization problem resides in selecting the block/MIC orders that are to be accepted and those to be rejected. The particularity of the block and MIC orders lies in the fact that they require the introduction of 0/1 variables in order to model their acceptance (0: rejected order, 1: accepted order). The discrete nature of these decision variables is referred to as the integrality constraint. The solution of this problem requires some decision variables to be integer (0/1) and the overall problem can be modeled as a Mixed-Integer Quadratic Program (MIQP).

A possible approach to solve such an MIQP problem is to use the branch-and-cut method. The branch-and-cut method is a very efficient technique for solving a wide variety of integer programming problems. It involves running a branch-and-bound algorithm and using cutting planes to tighten the QP relaxations. In the sequel, we will describe how the branch-and-cut method can be adapted to our particular welfare maximization problem and how cutting planes will be generated in the subsequent sub-problems in order to reduce the number and range of solutions to investigate.

6.2.1. Overview

EUPHEMIA starts by solving the initial MIQP problem where none of the variables is restricted to be integer. The resulting problem is called the QP relaxation of the original MIQP problem. For instance, relaxing the fill-or-kill constraint, *i.e.* the integrality constraint on the acceptance of the block orders, is equivalent to allowing all the block orders to be partially executed.

Because the QP relaxation is less constrained than the original problem, but still aims at maximizing *social welfare*, it always gives an upper bound on attainable *social welfare*. Moreover, it may happen that the solution of the relaxed problem satisfies all the integrality constraints even though these constraints were not explicitly imposed. The obtained result is thus feasible with respect to the initial problem and we can stop our computation: we got the best feasible solution of our MIQP problem. Note that this is rarely the case and the solution of the QP relaxation contains very often many fractional numbers assigned to variables that should be integer values.

6.2.2. Branching

In order to move towards a solution where all the constraints, including the integrality constraints, are met, EUPHEMIA will pick a variable that is violating its integrality constraint in the relaxed problem and will construct two new instances as following:

- The first instance is identical to the relaxed problem where the selected variable is forced to be smaller than the integer part of its current fractional value. In the case of 0/1 variables, the selected variable will be set to 0. This will correspond, for instance, to the case where the block order will be rejected in the final coupling solution.
- The second instance is identical to the relaxed problem where the selected variable is forced to be larger than the integer part of its current fractional value. In the case of 0/1 variables, the selected variable will be set to 1. This will correspond, for instance, to the case where the block order will be accepted in the final coupling solution.

Duplicating the initial problem into two new (more restricted) instances is referred to as branching. Exploring the solution space using the branching method will result in a tree structure where the created problem instances are referred to as the nodes of the tree. For each created node, the algorithm tries to solve the relaxed problem and branches again on other variables if necessary. It should be highlighted that by solving the relaxed problem at each of the nodes of the tree and taking the best result, we have also solved the initial problem (i.e. the problem in which none of the variables is restricted to be integer).

6.2.3. Fathoming

Expanding the search tree all the way till the end is termed as fathoming. During the fathoming operation, it is possible to identify some nodes that do not need to be investigated further. These nodes are either pruned or terminated in the tree which will considerably reduce the number of instances to be investigated. For instance, when solving the relaxed problem at a certain node of the search tree, it may happen that the solution at the current node satisfies all the integrality restrictions of the original MIQP problem. We can thus conclude that we have found an integer solution that still needs to be proved feasible. This can be achieved by verifying that there exist valid *market clearing prices* for each *bidding area* that are coherent with the market constraints. For this purpose, EUPHEMIA moves on to the price determination sub-problem (see section 6.3). If the latter sub-problem finds a valid solution for the current set of blocks/complex orders, we can conclude that the integer solution just found is feasible. Consequently, it is not required to branch anymore on this node as the subsequent nodes will not provide higher *social welfares*. Otherwise, if no valid solution could be found for the price determination sub-problem, we can conclude that the current block and complex order selection is unacceptable. Thus, a new instance of the welfare maximization problem is created where additional constraints are added to the welfare maximization problem that renders the previous integer solution infeasible (see section 6.2.4).

Let us denote the best feasible integer solution found at any point in the search as the incumbent. At the start of the search, we have no incumbent. If the integer feasible solution that we have just found has a better objective function value than the current incumbent (or if we have no incumbent), then we record this solution as the new incumbent, along with its objective function value. Otherwise, no incumbent update is necessary and we simply prune the node.

Alternatively, it may happen that the branch, that we just added and led to the current node, has added a restriction that made the QP relaxation infeasible. Obviously, if this node contains no feasible solution to the QP relaxation, then it contains no integer feasible solution for the original MIQP problem. Thus, it is not necessary to further branch on this node and the current node can be pruned.

Similarly, once we have found an incumbent, the objective value of this incumbent is a valid lower bound on the *social welfare* of our welfare maximization problem. In other words, we do not have to accept any integer solution that will yield a solution of a lower welfare. Consequently, if the solution of the relaxed problem at a given node of the search tree has a smaller welfare than that of the incumbent, it is not necessary to further branch on this node and the current node can be pruned.

6.2.4. Cutting

Introducing cutting planes is the other most important contributor of a branch-and-cut algorithm. The basic idea of cutting planes (also known as "cuts") is to progressively tighten the formulation by removing undesirable solutions. Unlike the branching method, introducing cutting planes creates a single new instance of the problem. Furthermore, adding such constraints (cuts) judiciously can have an important beneficial effect on the solution process.

As just stated, whenever EUPHEMIA finds a new integer solution with a better *social welfare* than the incumbent solution, it moves on to the price determination sub-problem and subsequent sub-problems. If in these sub-problems, we find out that the sub-problem is infeasible, we can conclude that the current block and complex order selection is unacceptable. Thus, the integer solution of the welfare maximization problem must be rejected. To do so, specific local cuts are added to the welfare maximization problem that renders the current selection of block and complex orders infeasible. Different types of cutting planes can be introduced according to the violated requirement that should be enforced in the final solution. For instance, if at the end of the price determination sub-problem, a block order is *paradoxically accepted*, the proposed cutting plane will force some block orders to be rejected so that the prices will change and will eventually make the block order no longer *paradoxically accepted*. Further types of cutting planes will be introduced in the subsequent sub-problems.

6.2.5. Stopping Criteria

The solution of the relaxed QP problem provides an upper bound on the achievable *social welfare* while the objective value of the incumbent is a valid lower bound on this *social welfare*. The difference between the current upper and lower bounds is known as the "*Gap*". Whenever the Gap is equal to zero, EUPHEMIA will abort the welfare maximization problem as no further improvement can be achieved. However, it may happen that the time limit would be reached before being able to reach a zero gap. In this latter case, EUPHEMIA will return the incumbent as the best solution found so far. Additional stopping criteria will be introduced later in this document.

6.3. Price Determination Sub-problem

In the master problem, EUPHEMIA has determined an integer solution with a given selection of block and complex orders. In addition, EUPHEMIA has also determined the matched volume of merit and aggregated hourly orders. In this sub-problem, EUPHEMIA must check whether there exist *market clearing prices* that are coherent with this solution while still satisfying the market requirements. More precisely, EUPHEMIA must ensure that the returned results satisfy the following constraints:

- The *market clearing price* of a given *bidding area* at a specific period of the day is coherent with the offered prices of the demand orders and the desired prices of the supply orders in this particular market.
- The *market clearing price* of a *bidding area* is compatible with the minimum and maximum price bounds fixed for this particular market.

However, the solution of this price determination sub-problem is not straightforward because of the constraints preventing the *paradoxical acceptance of block and MIC orders*, or preventing the presence of *adverse flows*. Indeed, whenever EUPHEMIA deems that the price determination sub-problem is infeasible, it will investigate the cause of infeasibility and a specific type of cutting plane will be added to the welfare maximization problem aiming at enforcing compliance with the corresponding requirement. This cutting plane will discard the current selection of block and complex orders.

- In order to prevent the *paradoxical acceptance of block orders*, the introduced cutting plane will reject some block orders that are *in-the-money*. Special attention will be paid when generating these cuts in order to prevent rejecting *deep-in-the money* orders.
- In order to prevent the acceptance of complex orders that do not satisfy their minimum income condition, the introduced cutting plane will reject the complex orders that will most likely not fulfill their minimum income condition.

- When the market coupling problem at hand features both block and complex orders, EUPHEMIA associates both cutting strategies in a combined cutting plane.
- Furthermore, if the bilateral intuitiveness mode is selected for the flow based model, the prices obtained at the end of the price determination sub-problem must satisfy an additional requirement. This requirement states that there cannot be *adverse flows*, *i.e.* flows exporting out of more expensive markets to cheaper ones. If the intuitiveness property is not satisfied, appropriate cutting planes are added as well to the welfare maximization problem.

Finally, if the price determination sub-problem is successful, the solution returned by EUPHEMIA should be free of any *false paradoxically rejected complex MIC order* (PRMIC). Thus, once the *market clearing prices* have been found, EUPHEMIA proceeds with an iterative procedure aiming to verify that all the rejected complex MIC orders, that are *in-the-money*, cannot be accepted in the final solution. For this purpose, EUPHEMIA first determines the list of *false PRMIC* candidates. Then, EUPHEMIA goes through the list, takes each complex MIC order from this list, activates it, and re-executes the price determination sub-problem. Two possible outcomes are expected:

- If the price computation succeeds and the *social welfare* was not degraded, we can conclude that the PRMIC reinsertion was successful. In this case, a new list of *false PRMIC* candidates is generated and the PRMIC reinsertion module is executed again.
- Conversely, if the price determination sub-problem is infeasible, or the *social welfare* is reduced, the complex MIC order candidate is simply considered as a true PRMIC, and the algorithm picks the next *false PRMIC* candidate. It should be noted that this case will not result to add a new cutting plane to the welfare maximization problem.

The PRMIC reinsertion module execution is repeated until no *false PRMIC* candidate remains. At this stage, we have obtained a feasible integer selection of block and complex orders along with coherent *market clearing prices* for all markets. Next, EUPHEMIA moves on to the PUN search sub-problem where it enforces the strong consecutiveness of the merit and PUN orders as well as the compliance with the PUN imbalance constraint.

It should be noted that the PRMIC reinsertion module will be executed again at the end of the PUN search sub-problem in order to ensure that the new solution found is still free of any *false paradoxically rejected complex MIC* order.

Branch-and-Cut Example

Here is a small example of the execution of the Branch-and-Cut algorithm (Figure 13).

At the start of the algorithm, we do not have an incumbent solution. EUPHEMIA first solves the relaxed welfare maximization problem where all the integrality constraints have been relaxed (Instance A). Let us assume that the solution of this problem has a *social welfare* equal to 3500 but has two fractional decision variables related to the acceptance of the block orders ID_23 and ID_54. At this stage, we can conclude that the upper bound on the attainable *social welfare* is equal to 3500.

Next, EUPHEMIA will pick a variable that is violating its integrality constraint (block order ID_23, for instance) and will branch on this variable. Thus, two new instances are constructed: Instance B where the block order ID_23 is rejected (associated variable set to 0) and Instance C where the block order ID_23 is accepted (associated variable set to 1). Then, EUPHEMIA will select one node that is not yet investigated and will solve the relaxed problem at that node. For example, let us assume that EUPHEMIA selects Instance B to solve and founds a solution where all the variables associated with the acceptance of block and complex orders are integral with a *social welfare* equal to 3050. Furthermore, we assume that the price determination sub-problem was successful and that a valid solution could be obtained. We can conclude that the solution of Instance B is thus feasible and can be marked as the incumbent solution of the problem. In addition, the obtained *social welfare* is a lower bound on any achievable welfare and it is not necessary to further branch on this node.

EUPHEMIA continues exploring the solution space and selects Instance C to solve. Let us assume that an integer solution was found with a *social welfare* equal to 3440. As the obtained *social welfare* is higher than that of the incumbent, EUPHEMIA moves on to the price determination sub-problem but let us assume that no valid *market clearing prices* could be found for this sub-problem. In this case, a local cut will be introduced to the welfare maximization problem. More precisely, an instance D is created identical to instance C where an additional constraint is added to render the current selection of block and complex orders infeasible. At this stage, we can conclude that the upper bound on the attainable *social welfare* is equal to 3440.

Now, let us assume that when solving the instance D of the problem, we get a solution with a *social welfare* equal to 3300 and a fractional decision variable related to the acceptance of the block order ID_30. As carried out previously, we need to branch on this variable. Thus, two new instances are constructed: Instance E where the block order ID_30 is rejected (associated variable set to 0) and Instance F where the block order ID_30 is accepted (associated variable set to 1). After solving the relaxed problem of Instance E, we assume that the obtained solution is integer with a *social welfare* equal to 3200. This *social welfare* is higher than that of the incumbent, so we try to solve the price determination sub-problem. We assume that the price determination sub-problem has a valid solution. Thus, the current solution for Instance E is feasible and is set as the new incumbent solution. We note that the lower bound on any achievable *social welfare* is now equal to 3200.

Similarly, after solving the relaxed problem of Instance F, we assume that the obtained solution has a *social welfare* equal to 3100 along with some fractional decision variables. As this solution has a lower *social welfare* than that of the incumbent, there is no need to further branch on this node and the current node can be pruned.

Figure 13 shows the search tree associated with our example.

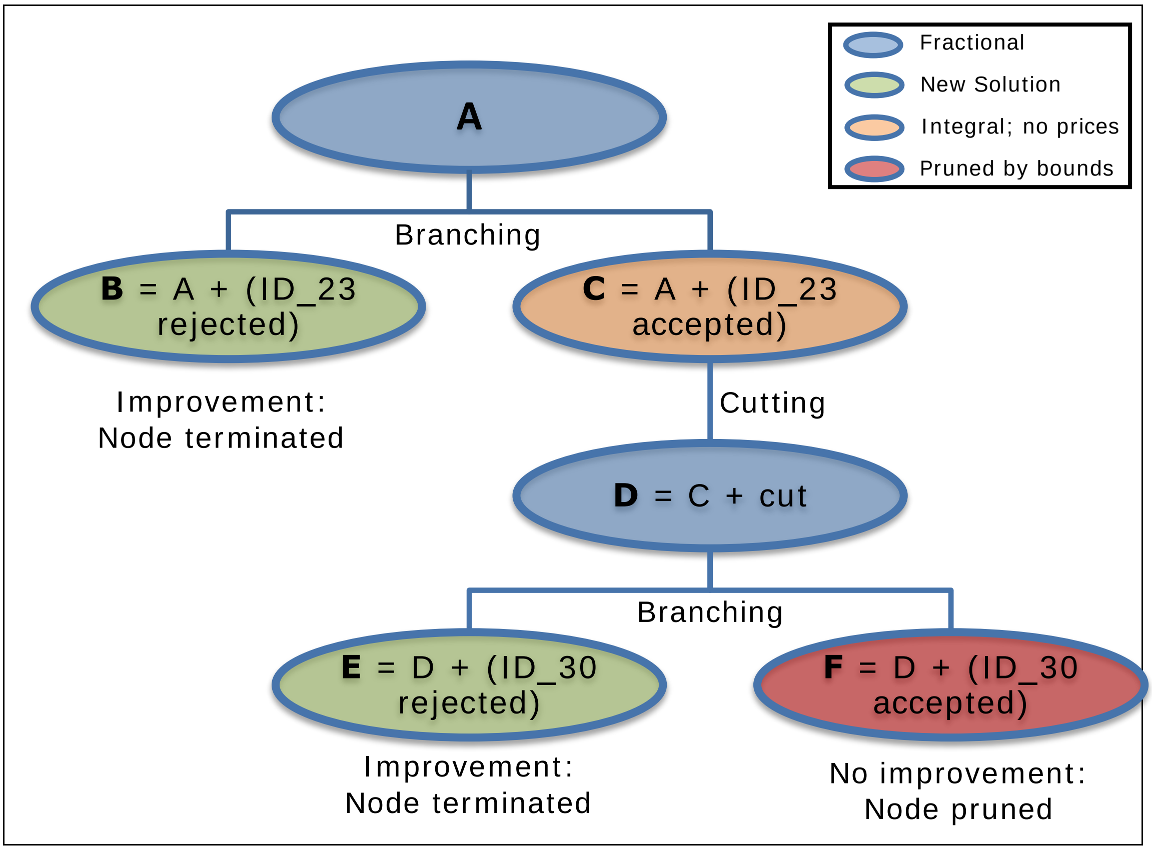


Figure 13 - Branch-and-Cut example

6.4. PUN Search Sub-problem

In order to avoid *paradoxically accepted* PUN orders, PUN (see Section 0) cannot be calculated as ex post weighted average of market price, but it must definitely be determined in an iterative process. Consider the following example:

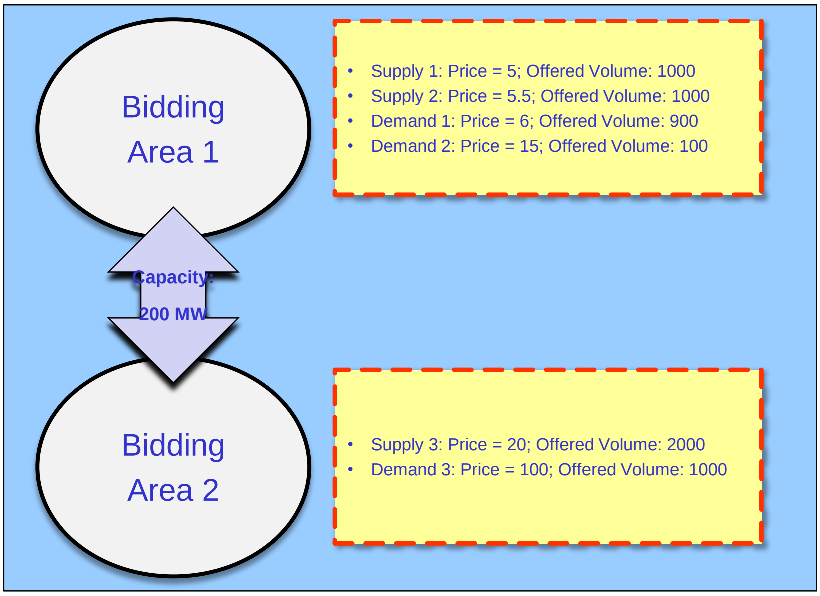


Figure 14 – PUN acceptance

If in Figure 15, Demand 1, Demand 2 and Demand 3 Orders were "simple" demand merit orders, then the market results would be:

- *Bidding area* 1:
  - *Market clearing price*: 5.5 €/MWh;
  - Executed Supply Volume: 1000 MWh;
  - Executed Demand Volume: 1000 MWh.
- *Bidding area* 2:
  - *Market clearing price*: 20 €/MWh;
  - Executed Supply Volume: 1000 MWh;
  - Executed Demand Volume: 1000 MWh.

If Demand 1, Demand 2 and Demand 3 Orders were "PUN" demand merit orders, then this solution is not acceptable. In fact, given a PUN imbalance tolerance=0, PUN calculated as weighted average will be:

[(1000 * 5.5) + (1000 * 20)] / 2000 = 12.75 €/MWh.

In this case, order Demand 1 would be *paradoxically accepted*.

Through an iterative process, the final solution will be the following:

- *Market clearing price* of *Bidding area* 1: 5 €/MWh;
- *Market clearing price* of *Bidding area* 2: 20 €/MWh;
- *PUN price*: 20 €/MWh;
- Supply order Supply 1: partially accepted (200 MWh);
- Supply order Supply 2: fully rejected;
- Supply order Supply 3: partially accepted (800 MWh)
- Demand orders Demand 1 and Demand 2: fully rejected;
- Demand order Demand 3: fully accepted;
- Flow from *Bidding area* 1 to *Bidding area* 2: 200 MWh;
- Imbalance: (1000 * 20) – (1000 * 20)= 0;
- Welfare: (1000 * 100) – [(200 * 5 + 800 * 20)] = 83000 €;

The PUN search is launched as soon as a first solution has been found at the end of the price determination sub-problem (activity 1 in Figure 15). This first solution respects all PCR requirements but PUN. The objective of the PUN search is to find, for each period, valid PUN volumes and prices (activity 2 in Figure 15) while satisfying the PUN imbalance constraint and enforcing the strong consecutiveness of accepted PUN orders.

If the solution found for all periods of the day, is compatible with the solution of the master problem (activity 3 in Figure 16), it means that a candidate solution is found. Otherwise, the process will resume calculating, for each period, new valid PUN volumes and prices to apply to PUN Merit orders.

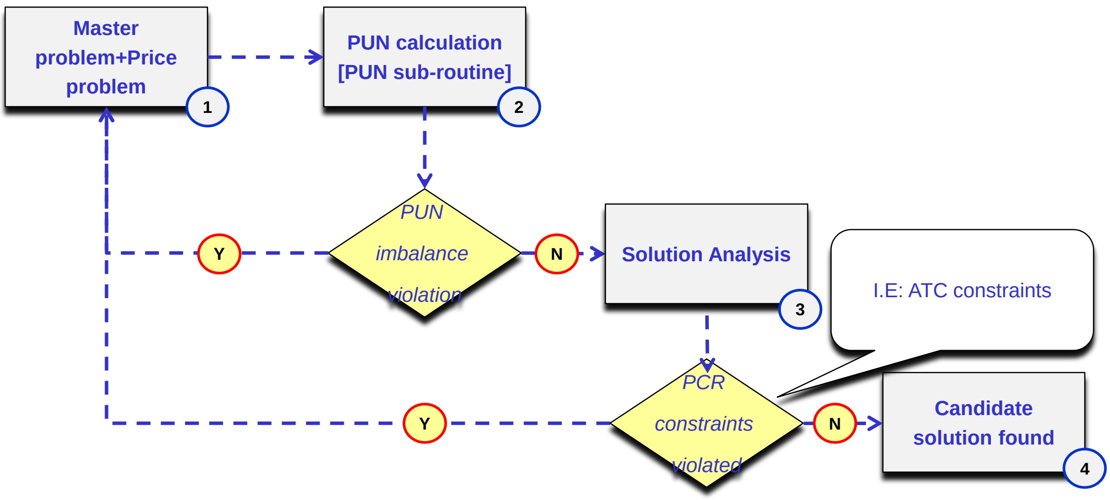


Figure 16 – PUN Search Sub-problem process

The PUN search is essentially an hourly sub-problem where the requirements are defined on an hourly basis, in which:

- Strong consecutiveness of PUN order acceptance is granted: a PUN order at a lower price cannot be satisfied until PUN orders at higher price are fully accepted
- PUN imbalance is within accepted tolerances.

For a given period, the selected strategy consists in selecting the maximum PUN volume (negative imbalance), and then trying to select smaller volumes until a feasible solution is found that minimizes the PUN imbalance.

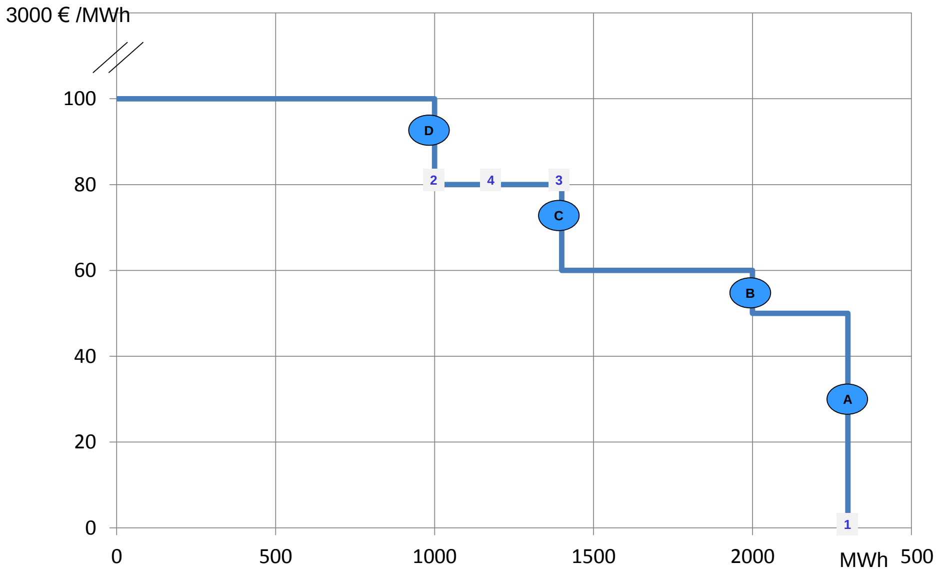


Figure 17 – PUN hourly curve

EUPHEMIA starts by calculating the PUN imbalance associated with the maximum accepted PUN volume (negative imbalance expected[3]; point 1 in Figure 17). If the PUN imbalance associated with the maximum PUN doesn't violate PUN imbalance tolerance, a candidate solution is found.

On the contrary, EUPHEMIA calculates the price which minimizes PUN imbalance (in Figure 17, analysis on vertical segment A) while the volume is fixed to the maximum accepted PUN volume. If the PUN imbalance calculated in this way is within the PUN imbalance tolerance interval, a candidate solution is found. If not, the next vertical segment (i.e. in Figure 17, vertical segment B), will be analyzed. This process is repeated until between 2 consecutive vertical segments, a change in sign of PUN imbalance is found (i.e. in Figure 17, positive PUN Imbalance in segment D; and negative PUN Imbalance in segment C). In this case, EUPHEMIA fixes the price (i.e. in Figure 17, the horizontal segment between point 2 and 3, to which corresponds a price of 80 €/MWh), and tries to minimize the PUN imbalance, using the volume as decision variable.

If the PUN imbalance calculated in this step is compatible with PUN imbalance tolerance, a candidate solution is found. If not, Euphemia continues the search on the horizontal segment (*i.e.* considering in Figure 17, let point 4 the one associated with PUN imbalance minimization at the price of 80 €/MWh. If in point 4, the imbalance is positive and greater than positive PUN imbalance tolerance, search will be continued in the interval between [4;3]; If in point 4, the imbalance is negative and less than negative PUN imbalance tolerance, the search will be continued in the interval between [2;4]).

[3] PUN consumers paid 0, producers receive market prices. Unless all market prices are equal to 0, imbalance will be negative

PUN SEARCH SUMMARY

1. *Calculation of PUN imbalance associated with maximum accepted PUN volume:*
    - *If minimum PUN imbalance tolerance ≤ calculated imbalance ≤ maximum PUN imbalance: candidate solution found*
    - *If imbalance < minimum PUN imbalance, next vertical segment is analyzed*
2. *Vertical segment analysis: Fixed the volume, minimization of the imbalance*
    - *If minimum PUN imbalance ≤ calculated imbalance ≤ maximum PUN imbalance: candidate solution found*
    - *If imbalance < minimum PUN imbalance, next vertical segment is analyzed*
    - *If imbalance > maximum PUN imbalance, next horizontal segment is analyzed*
3. *Horizontal segments analysis: Fixed the volume, minimization of the imbalance:*
    - *If minimum PUN imbalance ≤ calculated imbalance ≤ maximum PUN imbalance: candidate solution found*
    - *If imbalance < minimum PUN Imbalance, next horizontal segment is analyzed*
    - *If imbalance > maximum PUN Imbalance, next horizontal segment is analyzed*

The PUN search sequentially processes each period and fixes the PUN volume according to the strategy mentioned previously and then moves forward to the next period. Whenever no feasible solution could be found for a given period, PUN search backtracks to the previous period, looking for the next PUN solution.

As soon as PUN search is completed, EUPHEMIA verifies that the obtained PUN solution does not introduce any *paradoxically accepted block orders* or violates any other PCR constraints. If some block orders become *paradoxically accepted* or some other constrains are violated, a new cut is introduced to the welfare maximization problem that renders its current solution infeasible. Otherwise, EUPHEMIA proceeds with the lifting of volume indeterminacies.

6.5. Volume Indeterminacy Sub-problem

With calculated prices and a selection of accepted block, MIC and PUN orders that provide together a feasible solution to market coupling problem, there still might be several matched volumes, *net positions* and flows coherent with these prices. Among them, EUPHEMIA must select one according to the volume indeterminacy rules, the curtailment rules, the merit order rules and the flow indeterminacy rules. These rules are implemented by solving five closely related optimization problems:

- Curtailment minimization
- Curtailment sharing
- Volume maximization
- Merit order indeterminacy
- Flow indeterminacy

6.5.1. Curtailment minimization

A *bidding area* is said to be in curtailment when the *market clearing price* is at the maximum or the minimum allowed price of that *bidding area*. The curtailment ratio is the proportion of *price-taking orders* which are not accepted. All orders have to be submitted within a (technical) price range set in the respective *bidding area*. Hourly supply orders at the minimum price of this range and hourly demand orders at the maximum price of this range are interpreted as *price-taking orders*, indicating that the member is willing to sell/buy the quantity irrespective of the *market clearing price*.

The first step aims at minimizing the curtailment of these *price-taking* limit orders, *i.e.* minimizing the rejected quantity of *price-taking orders*. More precisely, EUPHEMIA enforces local matching of *price-taking hourly orders* with hourly orders from the opposite sense in the same *bidding area* as a counterpart. Hence, whenever curtailment of *price-taking orders* can be avoided locally on an hourly basis – *i.e.* the curves cross each other - then it is also avoided in the final results. This can be interpreted as an additional constraint setting a lower bound on the accepted *price-taking quantity* (see Figure 18 where the dotted line indicates the minimum of *price-taking supply quantity* to be accepted).

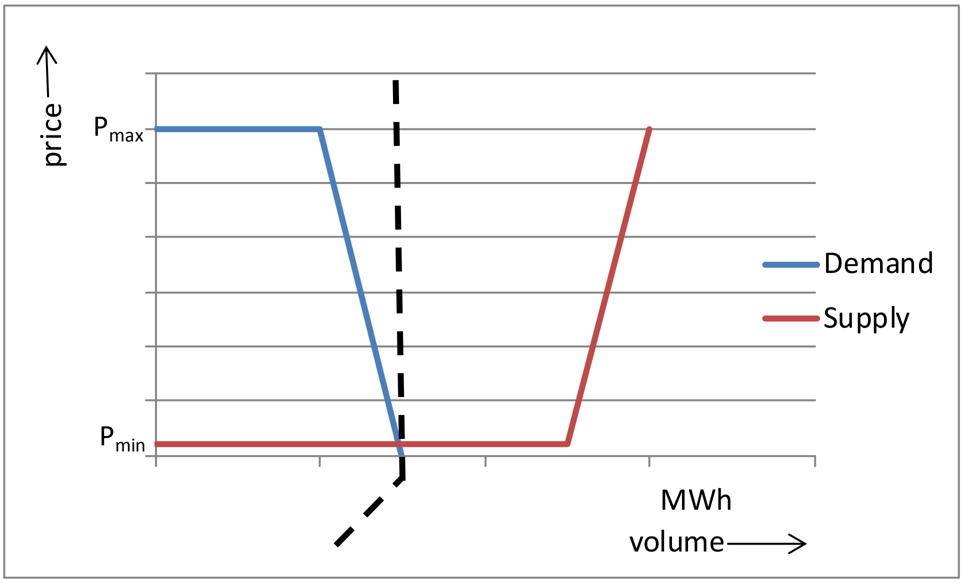


Figure 18 – Dotted line indicates the minimum of (*price-taking*) supply volume to be accepted

6.5.2. Curtailment sharing

This step guarantees that the curtailment is distributed in respect to identical curtailment ratio among *bidding areas* initially in curtailment, except for those biding areas that are not willing to share curtailment. The supply or demand orders within a *bidding area* being in curtailment at maximum (minimum) price are shared with other *bidding areas* in curtailment at maximum (minimum) price. For those markets that share curtailment, if they are curtailed to a different degree, the markets with the least severe curtailment (by comparison) would help the others reducing their curtailment, so that all the *bidding areas* in curtailment will end up with identical curtailment ratios in line with all network constraints.

6.5.3. Maximizing Accepted Volumes

In this step, the algorithm maximizes the accepted volume.

All hourly orders, complex hourly sub-orders, merit orders and PUN orders are taken into account for maximizing the accepted volumes. The acceptance of most orders is already fixed at this point. Either because it is completely below or above the *market clearing price*, or it is a *price-taking order* fixed at the first or second volume indeterminacy sub-problem (curtailment minimization or curtailment sharing). Block orders are not considered in this optimization because a feasible solution has been found prior to this step in the master problem.

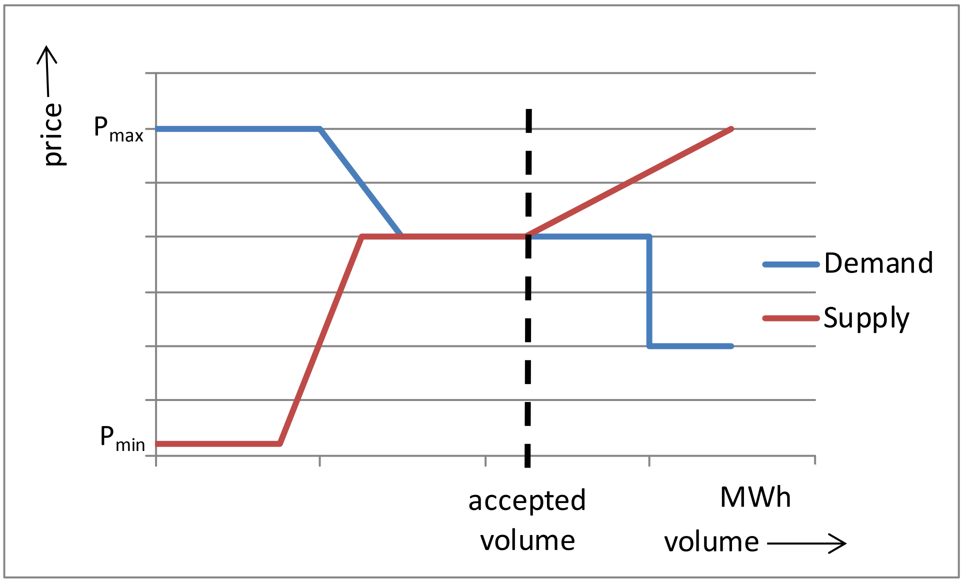


Figure 19 – The accepted volume is maximized

6.5.4. Merit order enforcement

This step enforces merit order numbers of the hourly orders if applicable. The acceptance of hourly orders with merit order numbers *at-the-money* is relaxed and re-distributed according to their acceptance priority. This problem is solved only if the solution found satisfies the PUN requirements (after the PUN search) or if there are no PUN orders but there exist some merit orders.

6.5.5. Flow indeterminacy

The last sub-problem re-attributes flows on the ATC lines based on the linear and quadratic cost coefficients of these lines. Apart from the flows, all other variables are fixed to their predetermined value. This step can only affect the results in situations where there is full price convergence within a meshed network, allowing multiple flow assignments to result in identical *net position*s. By using specific values for the cost coefficients, certain routes will be chosen and unique flows will be determined.

7. Additional Requirements

7.1. Precision and Rounding

- EUPHEMIA provides results (unrounded) which satisfy all constraints with a target tolerance (currently set at 10^{-5}). These prices and volumes (flows and *net positions*) are rounded by applying the commercial rounding (round-half-up) convention before being published.

7.2. Properties of the solution

During the execution of EUPHEMIA, several feasible solutions can be found. However, only the solution with the largest welfare value (complying to all network and market requirements) found before the stopping criterion of the algorithm is met is reported as the final solution.

It should be noted that for difficult instances some heuristics[4] are used by EUPHEMIA in its execution. Thus, it cannot be expected that the "optimal" solution is found in all cases.

[4] In mathematical optimization, a **heuristic** is a technique designed for solving a problem more quickly when classic methods are too slow, or for finding an approximate solution when classic methods fail to find any exact solution. This is achieved by trading optimality, completeness, accuracy, and/or precision for speed (Ref-: http://en.wikipedia.org/wiki/Heuristic_(computer_science)).

7.3. Stopping Criteria

As an optimization algorithm, EUPHEMIA searches the solution space for the best feasible solution until some stopping criterion is met. The solution space is defined as the set of solutions that satisfy all the constraints of the problem.

EUPHEMIA is tuned to provide a first feasible solution as fast as possible. However, after finding the first solution, EUPHEMIA continues searching ,the solution space for a better solution until a stopping criterion for example the maximum time limit of 10 minutes, is reached or until no more feasible selection of blocks and MIC orders exists.

Additional stopping criteria have also been implemented in the algorithm and can be used. The calculation will stop when one of these criteria is reached:

- **TARGET GAP**
  The target gap is the maximum gap allowed between the final solution and the optimal solution. More precisely, using a value different from 0 allows EUPHEMIA to stop as soon as it has proven that any other valid solution would be no more than TARGET GAP monetary units better than its current best valid solution.

- **TIME LIMIT**
  This parameter sets a limit to the total running time of EUPHEMIA. However, since the time taken by operations after calculation (e.g. writing of the solution in the database) can be variable, this is an approximate value.

- **ITERATION LIMIT**
  EUPHEMIA can stop after it has processed a given number of nodes.

- **SOLUTION LIMIT**
  EUPHEMIA can stop after it has found a given number of solutions (regardless of their quality).

7.4. Transparency

EUPHEMIA produces feasible solutions and chooses the best one according to the agreed criterion (welfare-maximization). Therefore the chosen results are well explainable to the market participants: published solution is the one for which the market value is the largest while respecting all the market rules.

7.5. Reproducibility

The reproducibility of an algorithm is defined as the capability of the algorithm to reproduce the same results upon request. On the same machine, two subsequent runs with the same input data should find the same solutions, meaning that the intermediate/final solutions found at iteration 'X' are the same. In other words, when the stopping criterion is the number of investigated solutions, a reproducible algorithm can guarantee to obtain the same final result. However, when the stopping criterion is a time limit, a faster computer will allow the algorithm to investigate more solutions than a slower one. In this case, the reproducibility consists in investigating on the faster computer at least the same set of solutions as the ones investigated on the slower computer.

Annex A. Glossary

- Bidding area: A bidding area represents a hub, that is, a virtual place where power is injected and/or withdrawn, and can be connected to other hubs through a network.
- Net position (net export position): The difference between accepted local supply and demand for a bidding area.
- Market Clearing Price (MCP): A common reference price for the whole Market area, when not considering transmission constraints.
- PUN price: PUN is the average (weighted by purchased quantity of PUN orders) of GME Zonal Market Prices (Italian "physical" zones). PUN is the price to consider accepting/rejecting purchase hourly orders made by PUN orders ("consumption purchase hourly orders").
- Adverse Flow: In market coupling, it is expected that the flow between two bidding areas goes from the market with a lower price towards the market with a higher prices. However, it may happen that, due to some constraints such as the ramping constraint imposed on some interconnectors, the cross-border flow end up being, at some particular periods, in the direction from a higher price bidding area towards a lower price bidding area. These flows are commonly known as "Adverse flows" and force the *Congestion Rent* to be negative.
- Consumer Surplus: The Consumer Surplus measures for the buyers whose orders are executed the difference between the maximum amount of money they are offering (limit price of their order × the executed volume of their order) and the amount of money they will effectively pay (market clearing price × the executed volume of their order).
- Producer Surplus: The Producer Surplus measures for the sellers whose orders are executed the difference between the minimum amount of money they are requesting (limit price of their order × executed volume of their order) and the amount of money they will

effectively receive (market clearing price × executed volume of their order).

- Congestion Rent: In an ATC model, the Congestion Rent measures for each interconnector traversed by a flow the difference between the total amount of money to be paid to the supplier of this flow at one end of the interconnector (market clearing price of the supplying bidding area × the volume of the energy flow through the interconnector) and the total amount of money to be received from the consumer of this flow at the other end of the interconnector (market clearing price of the consuming bidding area × the volume of the energy flow through the interconnector). It is equal to the product of the cross-border price spread and the implicit flow obtained by EUPHEMIA. The presence of losses on the interconnector will not impact the congestion rent. However, if the interconnector implements tariffs, the congestion rent will be reduced by the product of the tariff rates and the implicit flow obtained by EUPHEMIA.
- Social welfare: The Social Welfare is defined as the sum of the Consumer Surplus, the Producer Surplus, and the Congestion Rent.
- In-the-money: A supply (demand) order is considered in-the-money if its price is smaller (greater) than the market clearing price.
- At the money: A supply (demand) order is considered at-the-money if its price is equal to the market clearing price.
- Out of the money: A supply (demand) order is considered out-of-the-money if its price is greater (smaller) than the market clearing price.
- Deep in the money: A supply (demand) order is considered In-the-money if its price is smaller (greater) than the market clearing price plus a specified parameter (Max Delta P).
- Paradoxical acceptance of block orders: A block which is accepted while being out-of-the-money.
- False paradoxically deactivated complex MIC orders: A false paradoxically deactivated MIC order (false PR MIC) is a deactivated MIC whose economic condition seems to be fulfilled with the MCPs obtained in the final solution (so it seems that it should be activated) but, after acceptance its economic condition is not fulfilled anymore.
- Price-taking orders: Price taking orders (PTOs) are hourly buy (resp. sell) orders at the maximum (resp. minimum) price. PTOs are not block orders.

Annex B. Mathematical Approach

Purpose of EUPHEMIA algorithm is to grant the maximization of welfare, under a set of given constraints:

- network constraints
- clearing constraints
- hourly order acceptance rules
- price network properties
- kill – or – fill conditions
- no PAB constraints
- MIC constraints
- PUN consecutiveness constraints
- PUN imbalance constraints

In order to pursue this issue, EUPHEMIA relies on the concept of duality[5] to calculate prices and volumes on which welfare calculation is based on.

In the case of EUPHEMIA, the primal and dual problem can be synthesized as follows:

Problem	Unit	Variables	Constraints
Primal	MWh	Acceptance of Order Flow between bidding areas	Precedence between orders Network load limitations
Dual	€/MWh	Market Clearing Prices Congestion Rent	Constraints on price differences

[5] Duality is a relationship between two problems, called respectively the primal and dual. Each constraint in the primal problem corresponds to a variable in the dual problem (called its dual variable), and each variable in the primal problem has a corresponding constraint in the dual problem. The coefficients of the objective in the dual problem correspond to the right-hand side of the constraints in the primal problem. When the primal problem is a maximization problem, the dual is a minimization problem and vice-versa. Linear optimization problem is the dual of its dual. In the case of a convex problem, duality theory states that if both primal and dual problems are feasible, the optimal solutions of the primal and dual problems share the same objective value and exhibit a special relationship, called complementary slackness conditions. Specifically, whenever a constraint is not binding in the optimal primal (resp. dual) solution, then the corresponding dual (resp. primal) variable has a value of zero in the optimal dual (resp. primal) solution. Conversely, when a variable has a non-zero value in the primal (resp. dual), the corresponding constraint must be binding in the dual (resp. primal).

Strictly speaking, there are some reasons why the primal and dual problems in EUPHEMIA do not fit exactly in the above duality context.

1. The objective of the primal problem (the social welfare) is quadratic in terms of the acceptance variables. This is due to the interpolated orders: their marginal contribution to the welfare varies with the proportion matched. Fortunately, the Lagrangian duality principle still applies in the context of problems with quadratic objectives.
2. The primal problem contains integer variables. This is due to the presence of binary variables to represent the activation of blocks and complex orders. The linear duality theory unfortunately does not extend immediately to problems with integral variables. However, as soon as all integer variables have been fixed to certain values (that is, for a given selection of blocks and complex orders), then we are back into the regular duality theory context.
3. The dual problem in EUPHEMIA contains additional constraints which do not emerge naturally from the primal problem[6].
4. The coupling problem involves so called primal-dual constraints, i.e. constraints involving both primal and dual variables in their expression[7].
5. Not all dual variables are created. In particular, each order acceptance variable is bound to 1. This constraint should normally have a dual surplus variable, which would then play a role on the admissible prices. Almost all of those constraints would be redundant, so in the dual model of EUPHEMIA the price bounds are computed explicitly, and the surplus variables are not created.
6. The objective of the dual problem used by EUPHEMIA does not correspond to the primal one. Indeed, the objective value is already known from the primal problem and the goal of the dual problem will be to tackle other requirements, e.g. price indeterminacy rules.

Annex B.1. Welfare Maximization Problem

The purpose of the Master Problem is to find a good selection of blocks and complex orders (i.e. all binary variables) satisfying all of their respective requirements. The objective function of this problem is to maximize the global welfare:

[6] For example: the condition of accepted blocks to be not paradoxically accepted is not naturally met by an optimal primal-dual solution. Intuitively, this is related to the integer nature of the primal problem: by imposing the selection of blocks, we are exposed to the fact that some are losing money individually for the benefit of the social welfare.

[7] For example, the Minimum Income Condition for complex orders involves both the volumes matched (i.e. primal variables) and the market clearing prices (i.e. dual variables). Those constraints can only be formulated in the dual problem by substituting the corresponding primal variables by their optimal value in the primal problem, and reciprocally in dual one.

$$- \sum_{\substack{m,h,s,o: \\ Step\ Orders}} ACCEPT_{m,h,s,o} q_{m,h,s,o}\ p^{O}_{m,h,s,o} \quad (1)$$

$$- \sum_{\substack{m,h,s,o: \\ Interpolated\ Orders}} ACCEPT_{m,h,s,o} q_{m,h,s,o} \left(p^{O}_{m,h,s,o} + ACCEPT_{m,h,s,o} \frac{p^{1}_{m,h,s,o} - p^{O}_{m,h,s,o}}{2} \right) \quad (2)$$

$$- \sum_{bo,h} ACCEPT_{bo} q_{bo,h}\, p_{bo} \quad (3)$$

$$- \sum_{m,co,h} ACCEPT_{m,co,h,o}\ q_{m,co,h,o} p_{m,co,h,o} \quad (4)$$

$$- \sum_{mo} ACCEPT_{mo}\ \ q_{mo} p_{mo} (5)$$

$$- \sum_{l,u,h} Tariff_{l,h}\ FLOW_{l,u,h} (6)$$

where (bearing in mind that q_o is positive for a supply order and negative for demand orders):

- IS THE CONTRIBUTION OF HOURLY STEP ORDERS
- IS THE CONTRIBUTION OF HOURLY INTERPOLATED ORDERS
- IS THE CONTRIBUTION OF BLOCK ORDERS
- IS THE CONTRIBUTION OF COMPLEX ORDERS
- IS THE CONTRIBUTION OF MERIT ORDERS
- IS THE IMPACT OF TARIFFS

Subject to:

- Market constraints
  - Balance/clearing constraints
  - Block order acceptance constraint
  - Complex suborders acceptance constraints
  - Load Gradient constraint
  - Merit order acceptance constraints
- Network constraints
  - ATC constraints
  - PDTF constraints
  - Various ramping constraints

Annex B.2. Price Determination Sub-problem

For each feasible solution of the primal problem, EUPHEMIA solves the following price problem:

$$\min_{prices} distance\ to\ mid\ point$$

i.e.:

$$min \sum_{m,h} \left(MCP_{m,h} - \frac{UpperBound_{m,h} + LowerBound_{m,h}}{2} \right)^2$$

Subject to:

- complementarity slackness conditions
- price bounds
- no PAB constraints
- Minimum Income Condition
- PUN imbalance

Annex B.3. Indexes and Annotations

m	Bidding area
h	Period
s	Supply/Demand
c	Curve identified by *m,h,s*
o	Hourly Order identified by *m,h,s,o*
bo	Block Order
mo	Merit order
po	PUN order
co	Complex Order , where • complex curve is identified by *m,co,h* • complex suborder by *m,co,h,o*
l	(DC/ATC) Line
uu(convention: up=0 and down=1)	Up/Down direction

ACCEPT [0;1]	Acceptance variables
p	Offered Price
q	Offered Volume
MCP	*Market clearing price*