

# TERRE

## *Trans European Replacement Reserves Exchange*

### TERRE Stakeholder Workshop

26<sup>th</sup> June 2019



# Agenda

## TERRE Stakeholder Workshop

#	TOPICS	Timing (CET)
	Registration & Coffee	9:30
0	Welcome message	10:00
1	Introduction from TSOs and NRAs	10:15
2	RR Implementation Framework	10:30
3	RR pricing and TSO-TSO settlement rules applied for the RR Platform Go-Live	11:00
4	RR inputs format description	11:30
5	Algorithm Optimisation Function description-part 1	12:00
	Lunch break	12:30
5	Algorithm Optimisation Function description-part 2	13:15
6	RR process timing	13:45
7	Parallel Run phase, BSP involvement and planning for Go-Live(s)	14:00
	Break	14:45
8	Transparency and publication	15:00
9	Questions & Answers	15:30
	End of the meeting	16:00



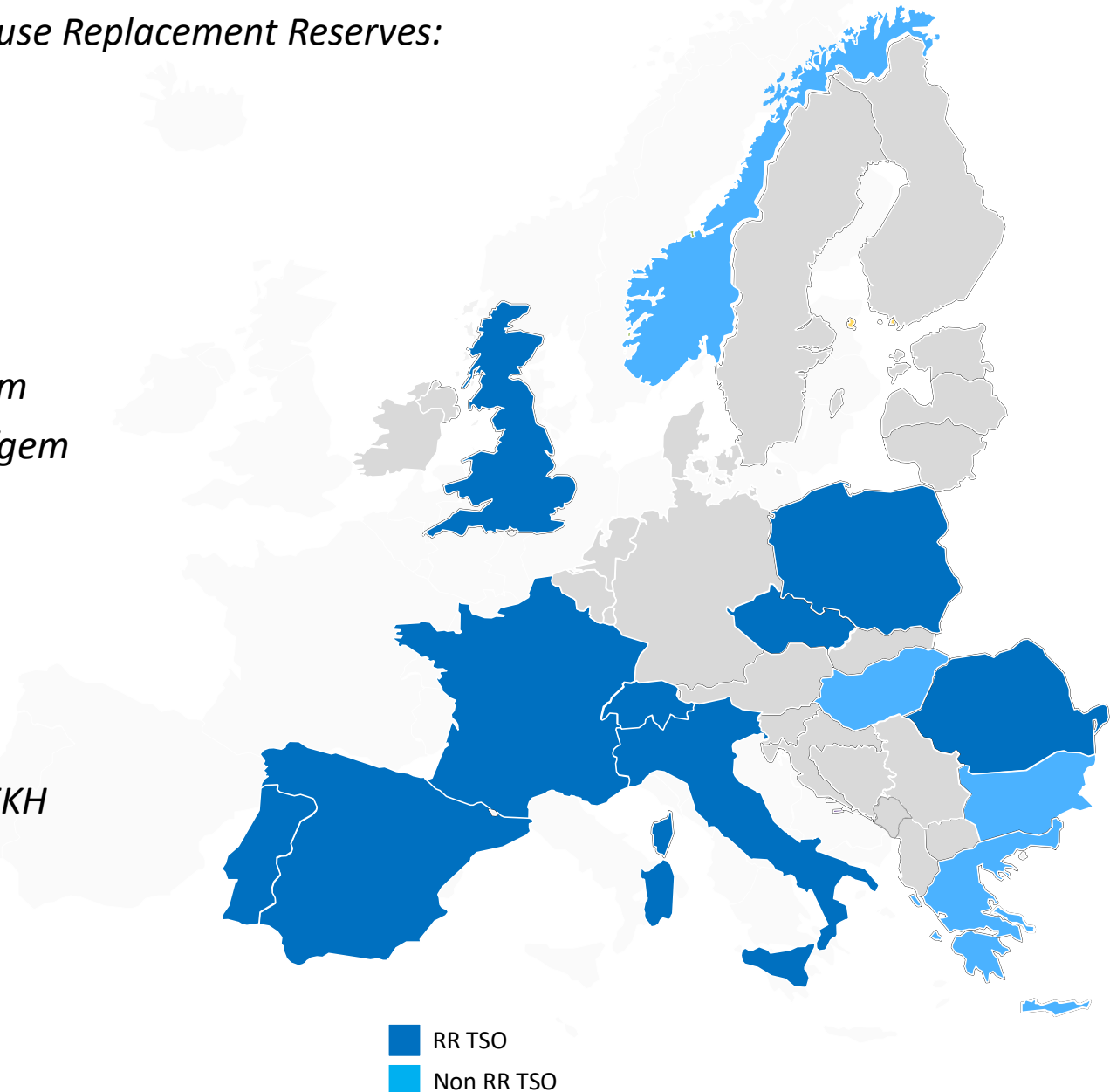
# **Topic 1: Introduction from TSOs and NRAs**

# 1. Introduction

## TERRE project - Participating TSOs & NRAs

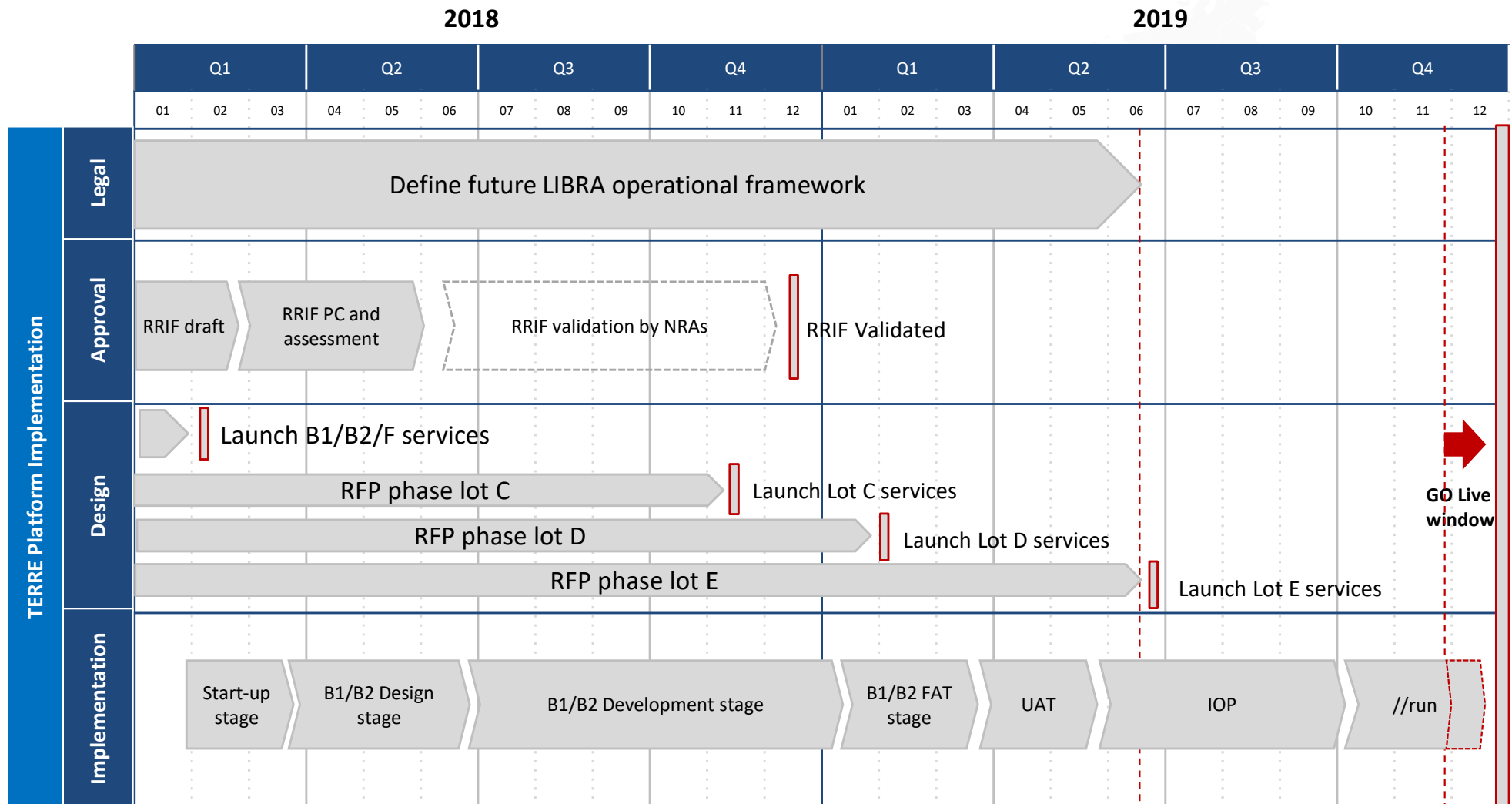
The following countries plan to use Replacement Reserves:

- **Region 1**
  - France (RTE) - CRE
  - Italy (TERNA) - ARERA
  - Portugal (REN) - ERSE
  - Spain (REE) - CNMC
  - Switzerland (SG) - Elcom
  - Great Britain (NG) - Ofgem
- **Region 2**
  - Poland (PSE) - URE
  - CZ (CEPS) - ERU
- **Region 3**
  - Romania (TEL) - ANRE
  - Hungary (MAVIR) - MEKH
  - Bulgaria (ESO) - DKER
- **Observers:**
  - Greece (ADMIE)
  - Nordic area



# 1. Introduction

## High level planning

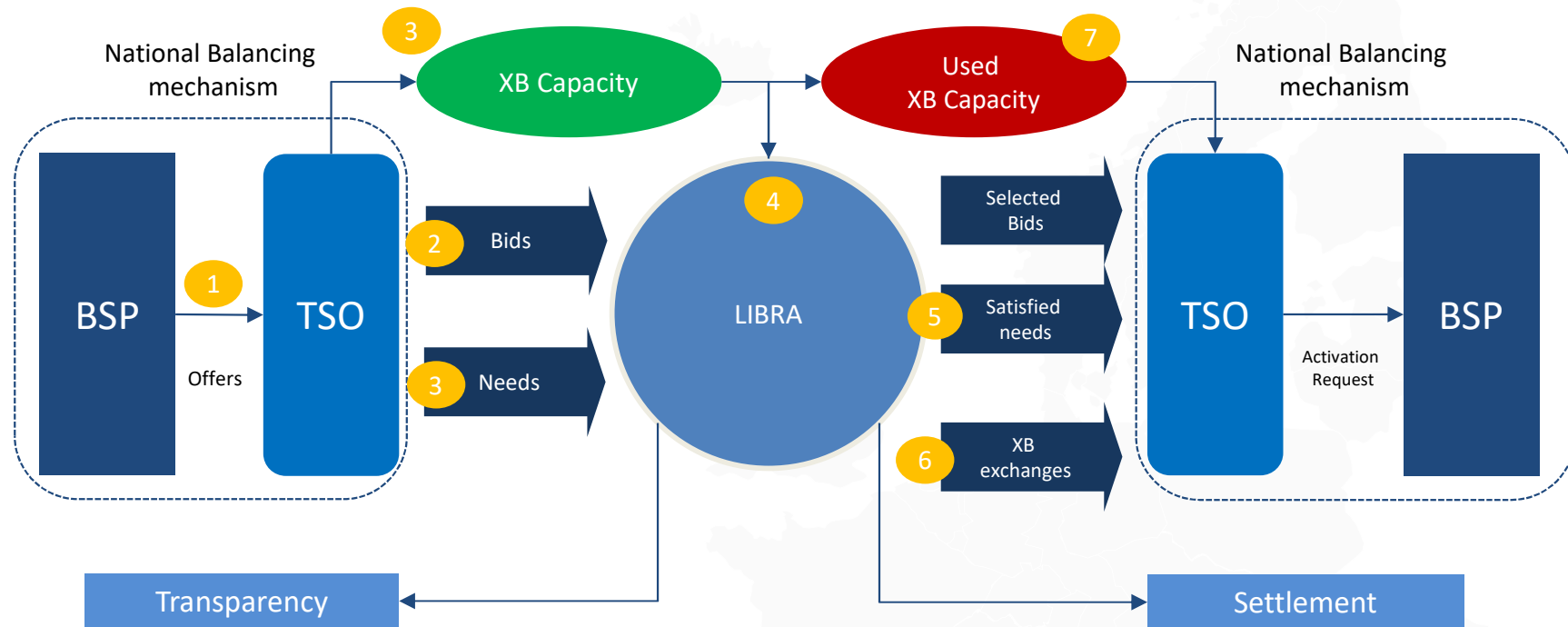


B1: Optimization Module; B2: Data Management Module; F: Testing; C: Hosting; D: IT monitoring; E: Financial service

Today

# 1. Introduction

## RR process overview



1. TSO receive bids from BSPs from their local balancing area/bidding zone.
2. TSOs put the valid RR bids on the LIBRA platform
3. TSOs send their needs and ATC values to the platform.
4. Platform runs the algorithm with offers and needs.
5. Communication of accepted offers, satisfied needs and marginal prices
6. Calculation of the bilateral exchanges between balancing areas and TSO-TSO settlement.
7. Residual ATC and net positions are communicated to TSOs



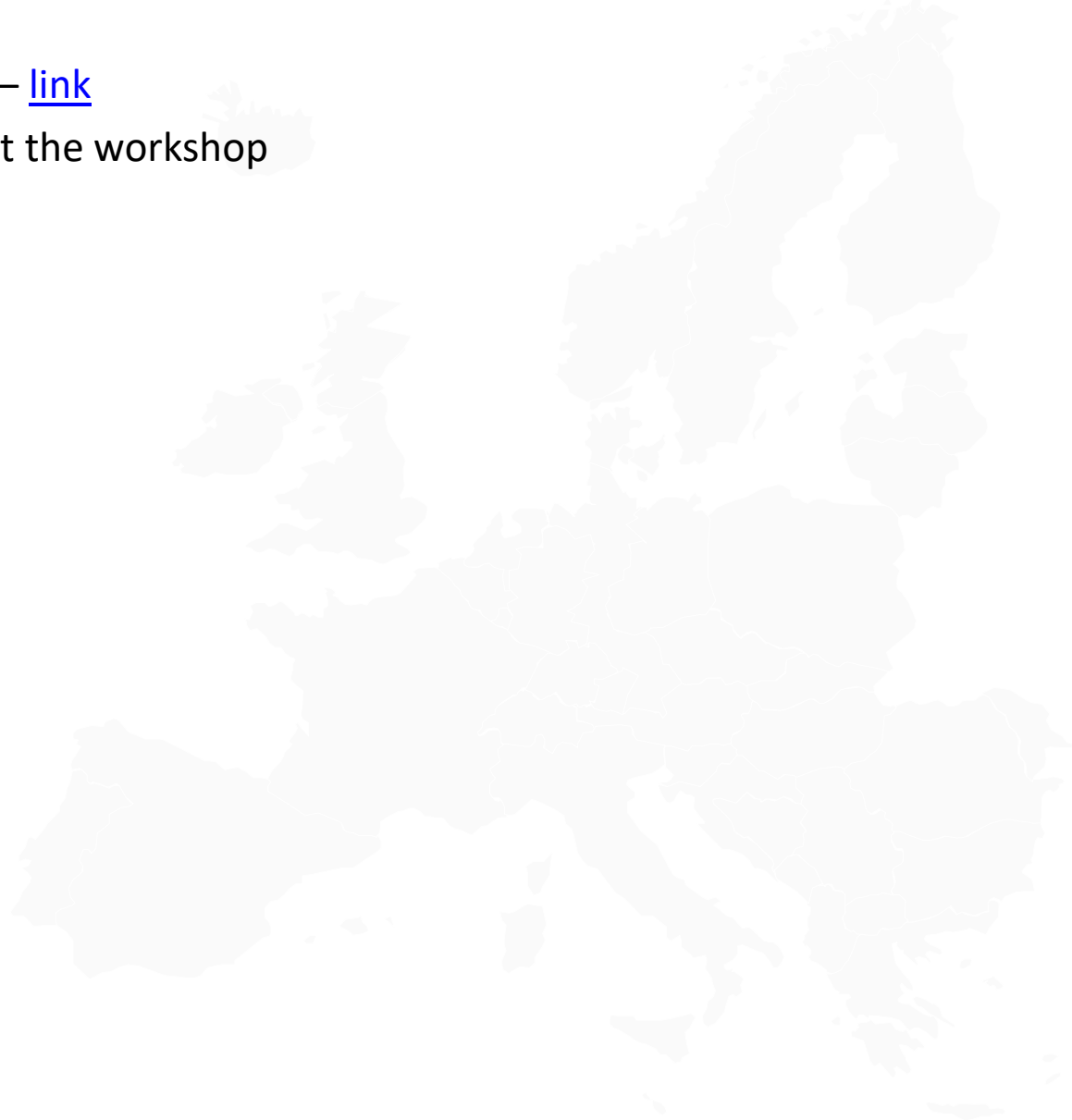
## **Topic 2: RR Implementation Framework**

## 2. RR Implementation Framework


### Highlights

---

- RRIF validated by the NRAs – [link](#)
- Feedback to be presented at the workshop







## **Topic 3: RR pricing and TSO-TSO settlement rules applied for the RR Platform Go-Live**

## 3. RR pricing and TSO-TSO settlement rules applied for the RR Platform Go-Live

### Highlights

---

#### Pricing proposal ([link](#))

---

- Article 3: General principles
- Article 4: Additional Provisions for the Pricing of Standard RR Balancing Energy Product Bids

#### TSO-TSO settlement proposal ([link](#))

---

- Article 6: Settlement of the intended balancing energy activated for system constraints purposes

When TERRE goes live, to simplify the calculation, the extra cost due to constraint will be equally paid by the TSOs who evoke the request. In a later stage it will be divided proportionally to the sum of the differences between the flow from CC run and UC run on each interconnector

These rules will be implemented for TERRE go live but may change in the future because the proposals are still not validated by ACER

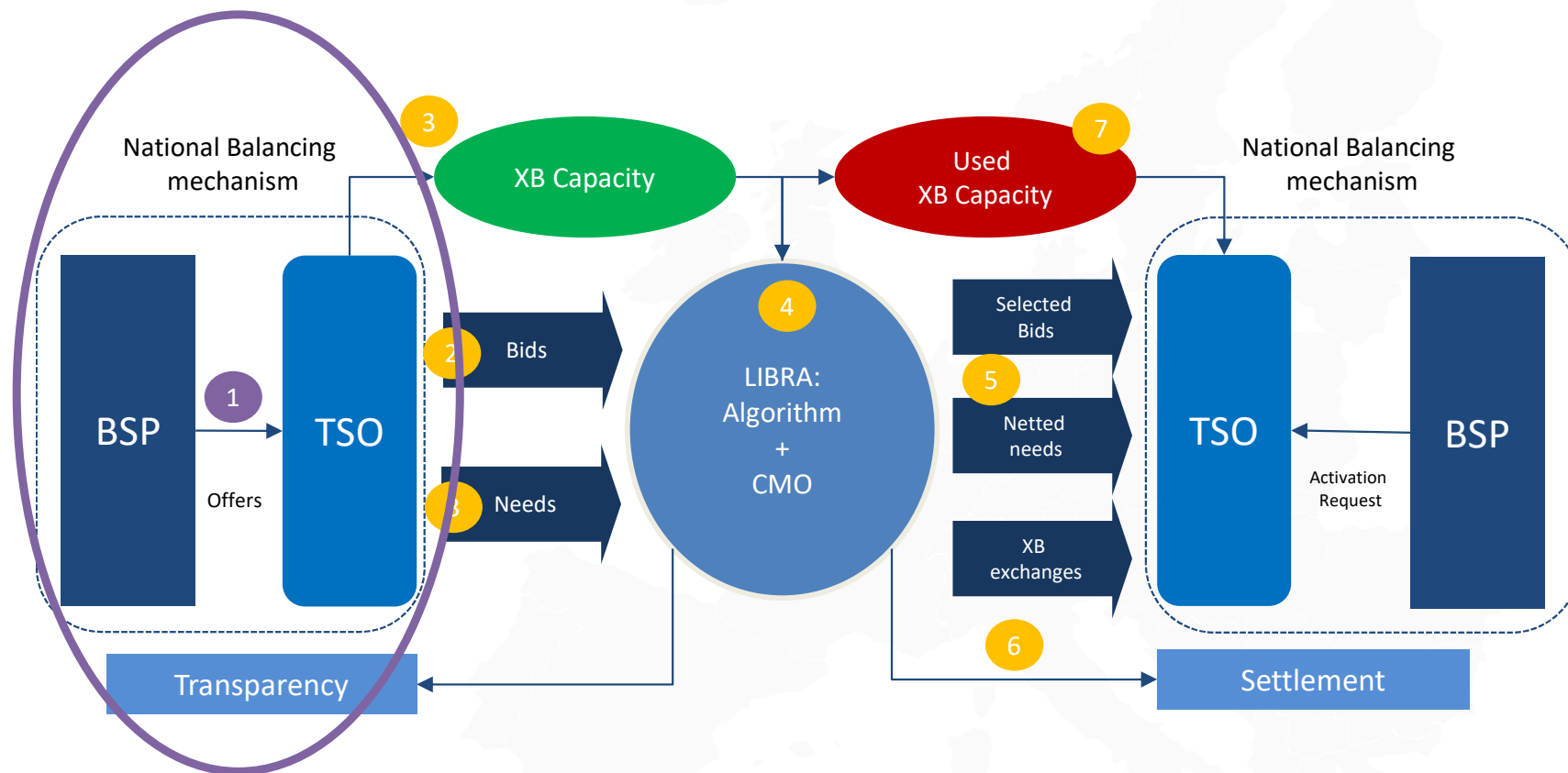


## **Topic 4: RR inputs format description**

## 4. RR inputs format description

### Optimization Algorithm – Inputs

What kind of bids can be submitted?



*BSP submitting balancing energy bids to their TSO*

## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What kind of bids can be submitted?

- For each of the format below, please indicate if it is allowed in the LIBRA platform or not.
- Then, indicates its characteristics

1. Full divisible bids



2. Divisible bids



3. Indivisible bids



4. Linked bids in time

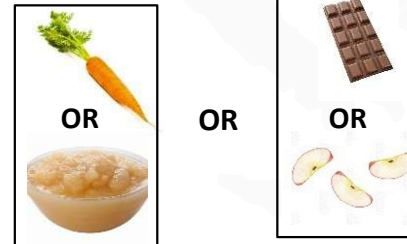
5. Exclusive bids in volume



6. Exclusive bids in time



7. Exclusive bids in time and volume



8. Multi part bids

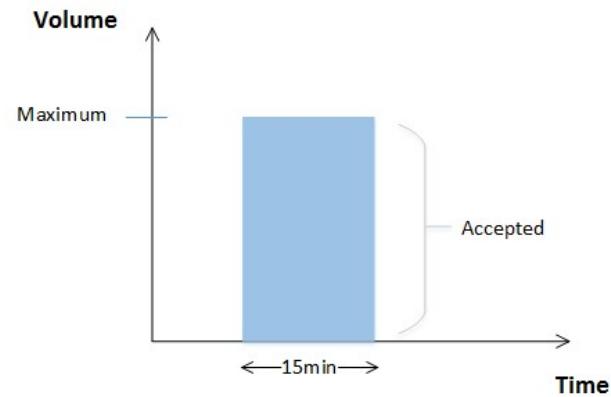


## 4. RR inputs format description

### Optimization Algorithm – Inputs

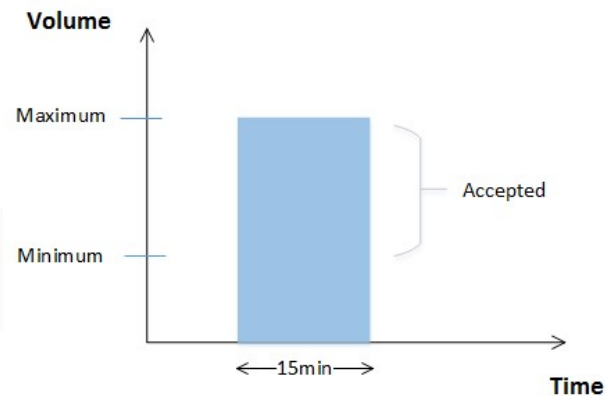
#### What kind of bids can be submitted?

- **Full divisible bids:**



- The accepted volume is less or equal to the maximum volume and greater or equal to 0
- The bid is rejected when the accepted volume is equal to 0
- They correspond to single time steps, i.e. to 15 minute time steps

- **Divisible bids:**



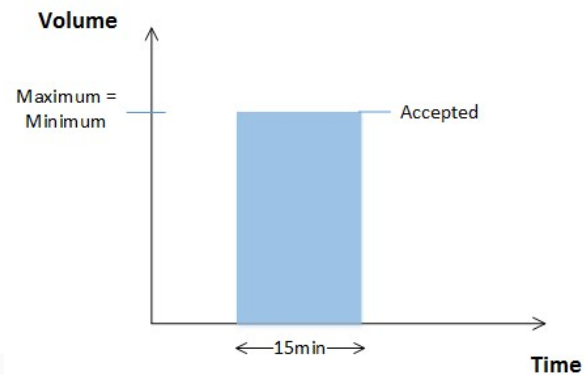
- The accepted volume is less or equal to the maximum volume and greater or equal to the minimum quantity
- The bid is rejected when the accepted quantity is equal to zero
- They correspond to single time steps, i.e. to 15 minute time steps

## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What kind of bids can be submitted?

- **Indivisible bids:**



- Either the whole bid or nothing is accepted
- They correspond to single time steps, i.e. to 15 minute time steps

## 4. RR inputs format description

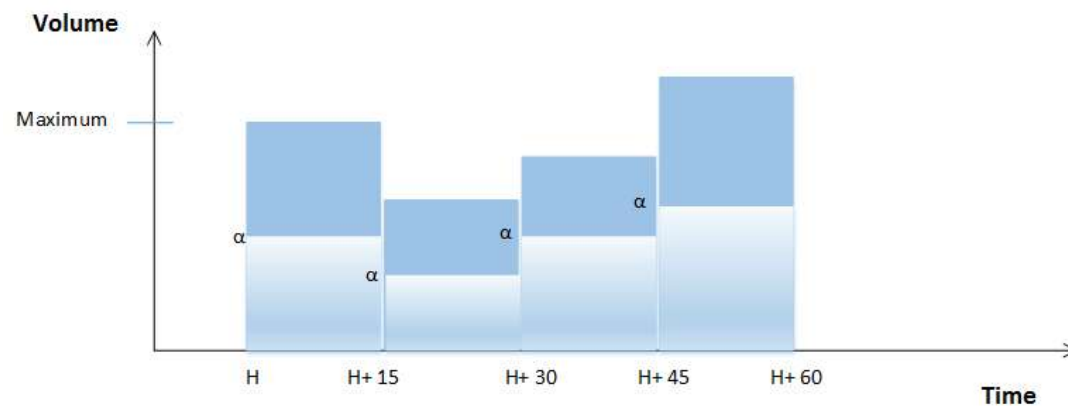
### Optimization Algorithm – Inputs

#### What kind of bids can be submitted?

##### ▪ Linked bids in time

BSPs can submit linked bids in time:

- Either with a curve  $\rightarrow$  single bid (direction upward or downward)
- Or with explicit links  $\rightarrow$  two or more bids with an explicit link in time
- Linking in time and in volume simultaneously is not allowed



- They can be either fully divisible, or divisible or indivisible bids
- They can offer different quantities/prices per time step
- The explicit links can only link bids (including curve) corresponding to different time steps
- The same percentage ratio  $\alpha$  will be accepted:  $\alpha = \text{Accepted volume of each bid} / \text{Maximum volume of each bid}$

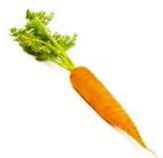


## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What kind of bids can be submitted?

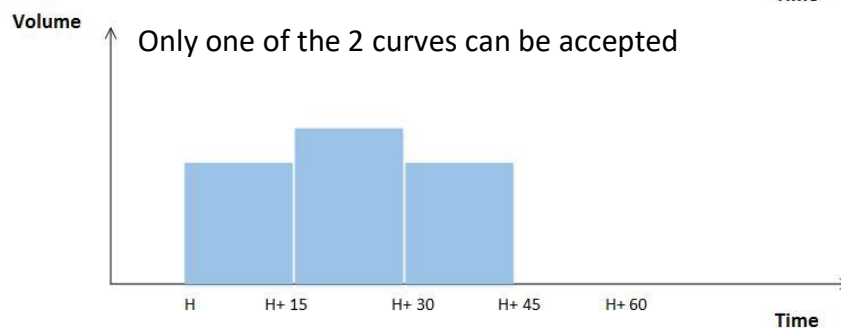
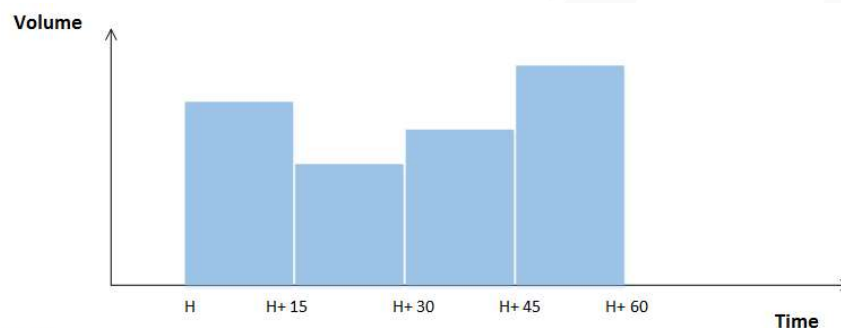
- **Exclusive bids in volume**



OR



Example with 2 curves exclusive in volume



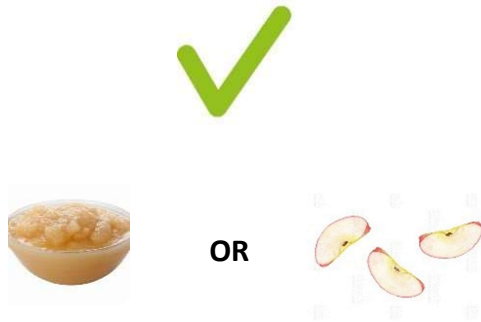
- They can be either fully divisible, or divisible or indivisible bids
- Bids corresponding to single time steps can be exclusive in volume
- Curves (linking bids in time) can also be exclusive in volume
- Maximum one exclusive bid can be accepted

## 4. RR inputs format description

### Optimization Algorithm – Inputs

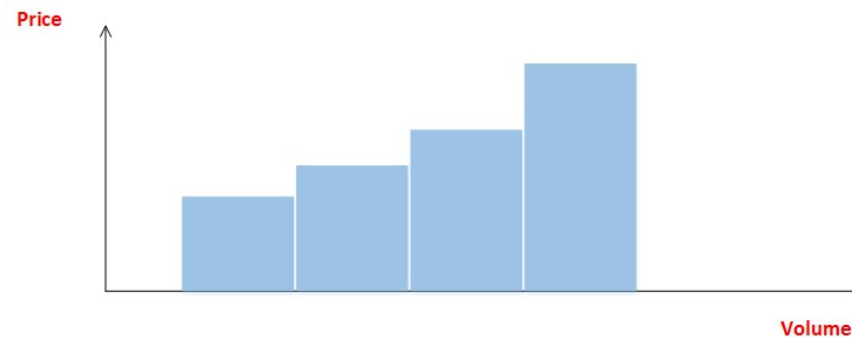
#### What kind of bids can be submitted?

##### ■ Exclusive bids in time



- They can be either fully divisible, or divisible or indivisible bids
- Bids corresponding to single time steps only can be exclusive in volume
- Maximum one exclusive bid can be accepted

##### ■ Multi-part bids

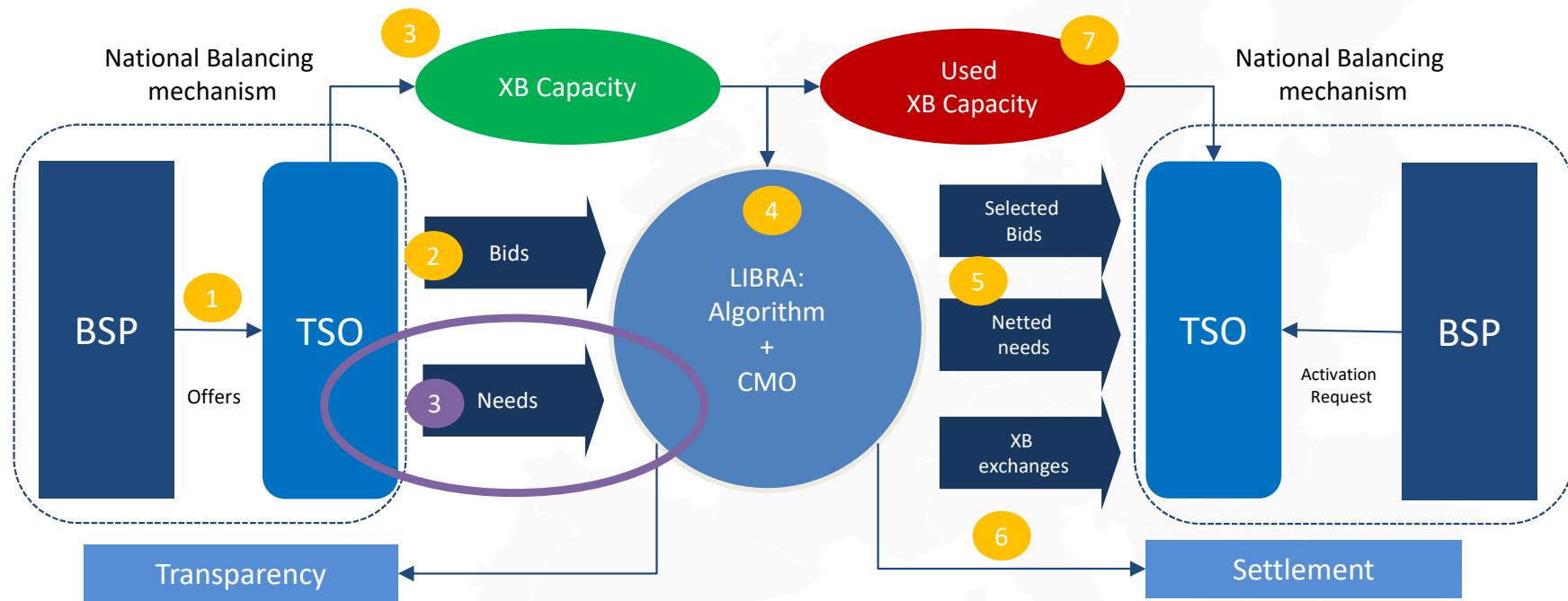


- Increasing or decreasing only prices
- They can be either fully divisible, or divisible or indivisible bids
- A multi-part bid can be defined with a starting and an ending time and can last from 15 to 60 minutes.. The same volume will be accepted for the whole defined delivery period

## 4. RR inputs format description

### Optimization Algorithm – Inputs

What kind of needs can be submitted?

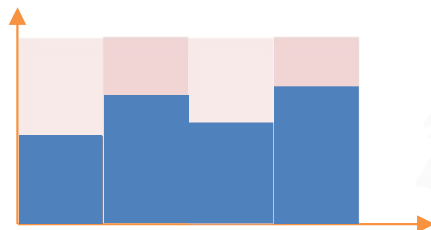
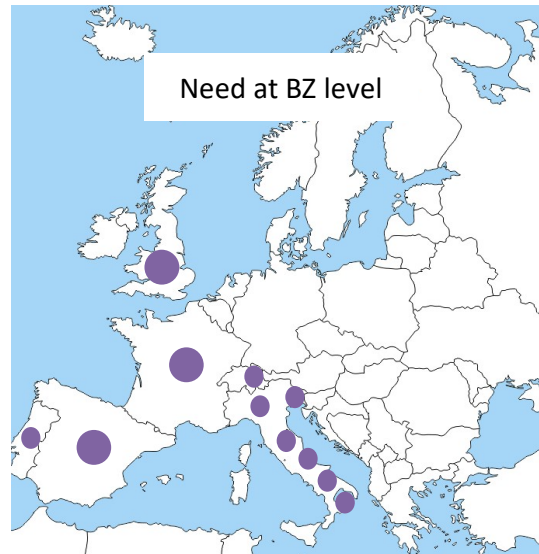


*TSO submitting balancing needs to the platform*

## 4. RR inputs format description

### Optimization Algorithm – Inputs

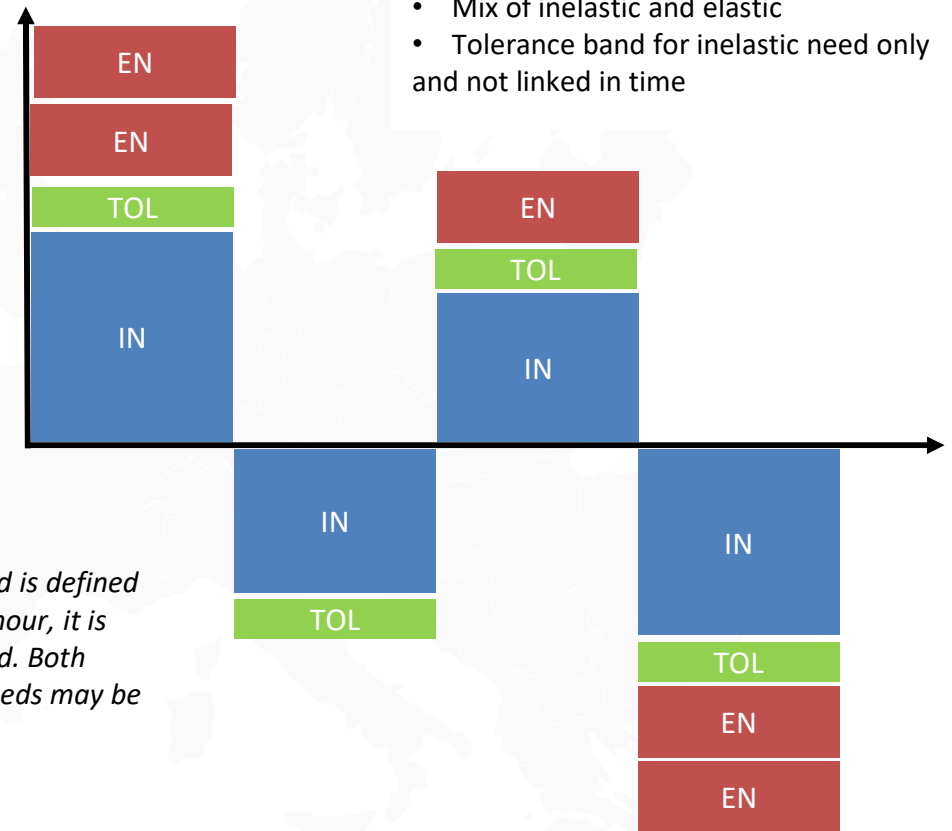
#### What kind of needs can be submitted?



*When a balancing need is defined over multiple quarter hour, it is referred as linking need. Both elastic and inelastic needs may be linked.*

#### BASIC NEED FORMAT

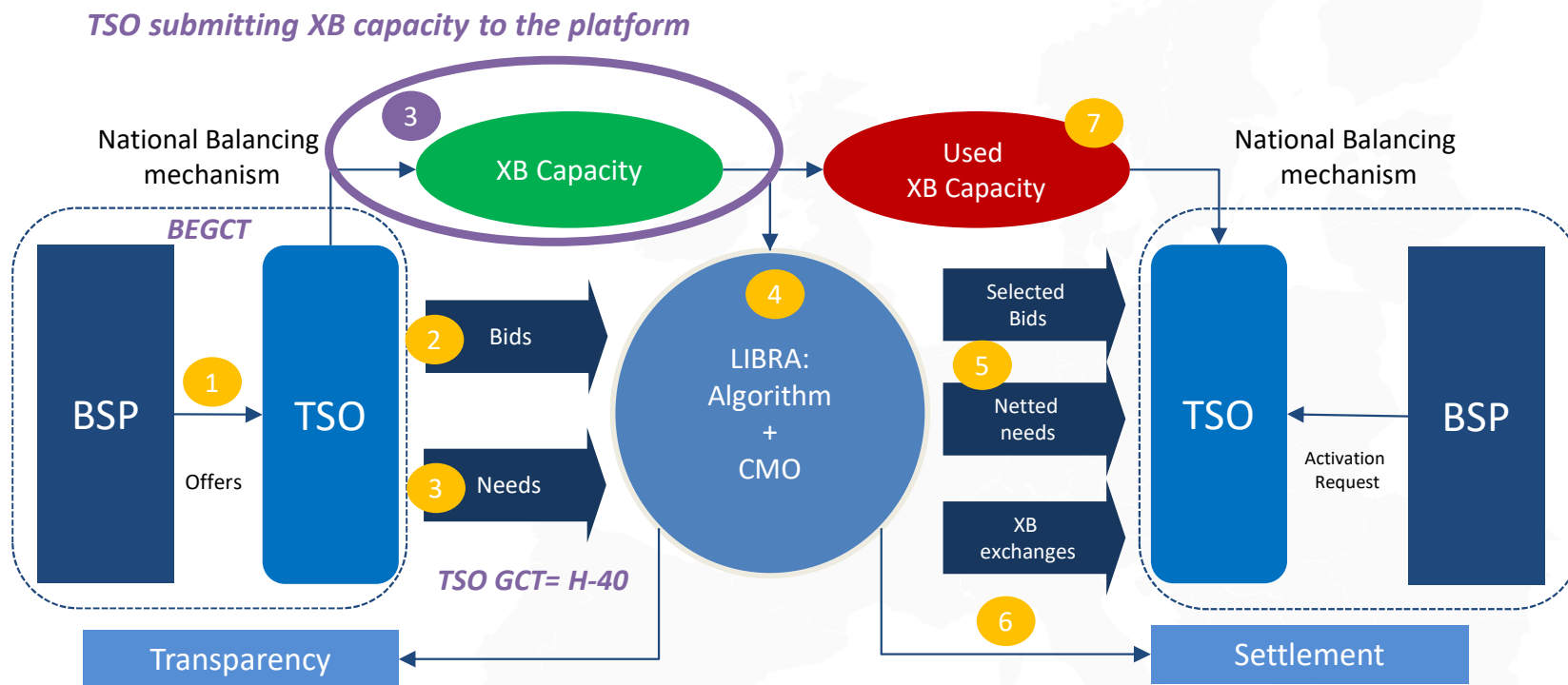
- 15 min resolution
- Upward or downward direction
- Mix of inelastic and elastic
- Tolerance band for inelastic need only and not linked in time



## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What XB capacity format?

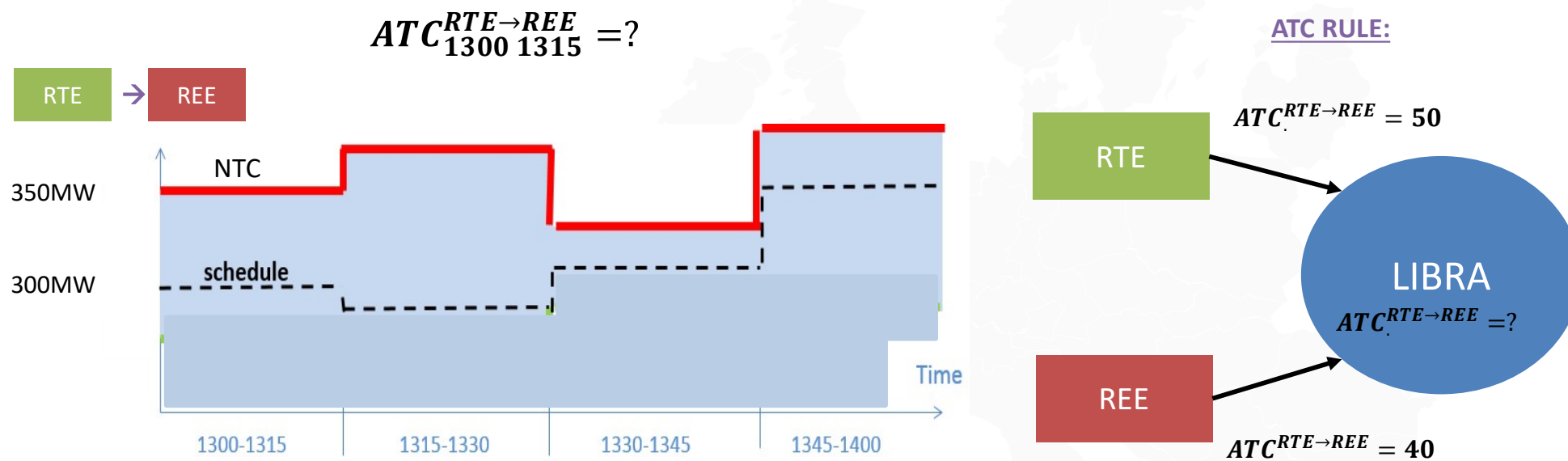


## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What XB capacity format?

- ATC document for the ATC in both directions

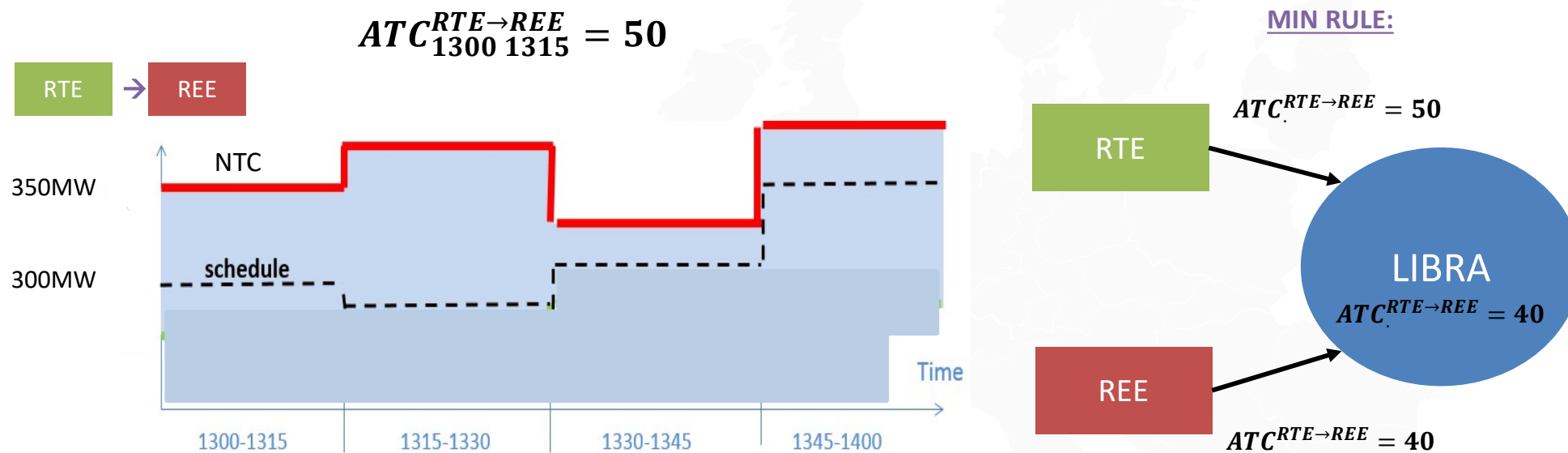


## 4. RR inputs format description

### Optimization Algorithm – Inputs

#### What XB capacity format?

- ATC document for the ATC in both directions and will cover the entire delivery period

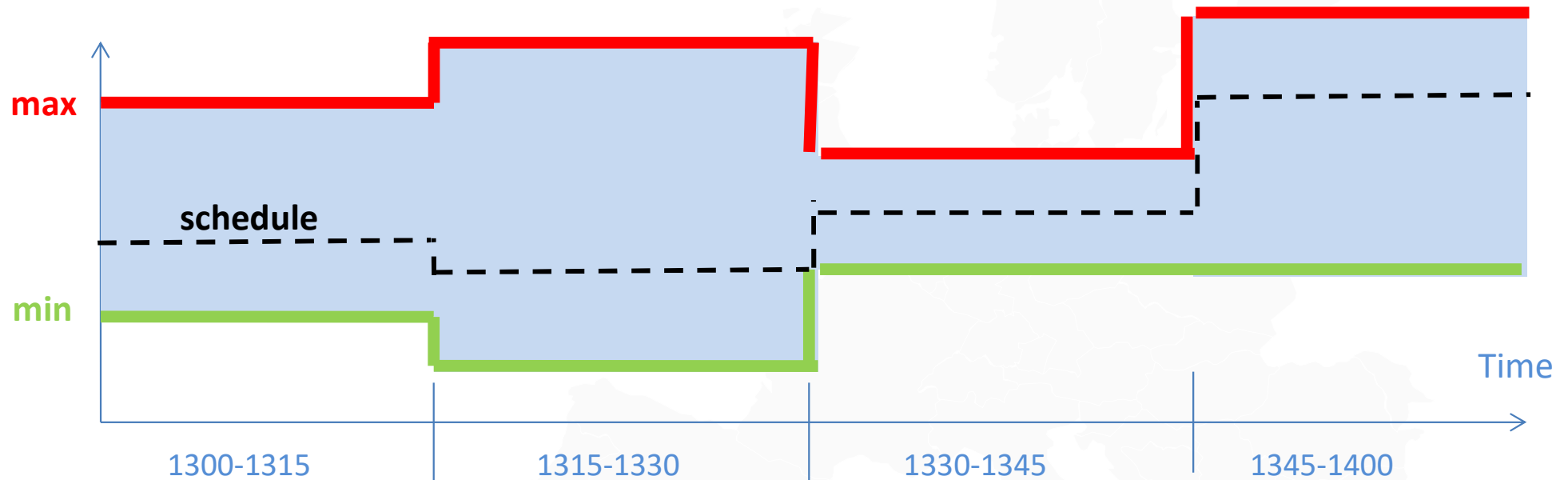


## 4. RR inputs format description

### Optimization Algorithm – Inputs

- HVDC document, two documents transmitted:
  - 1) Schedule
  - 2) Min and max permitted values

#### Schedule



FR>UK time step 1300-1315: Schedule = 300MW, min=100MW, max=700MW

Means algorithm must allocate flow in direction FR>UK from -200MW up to +400MW.

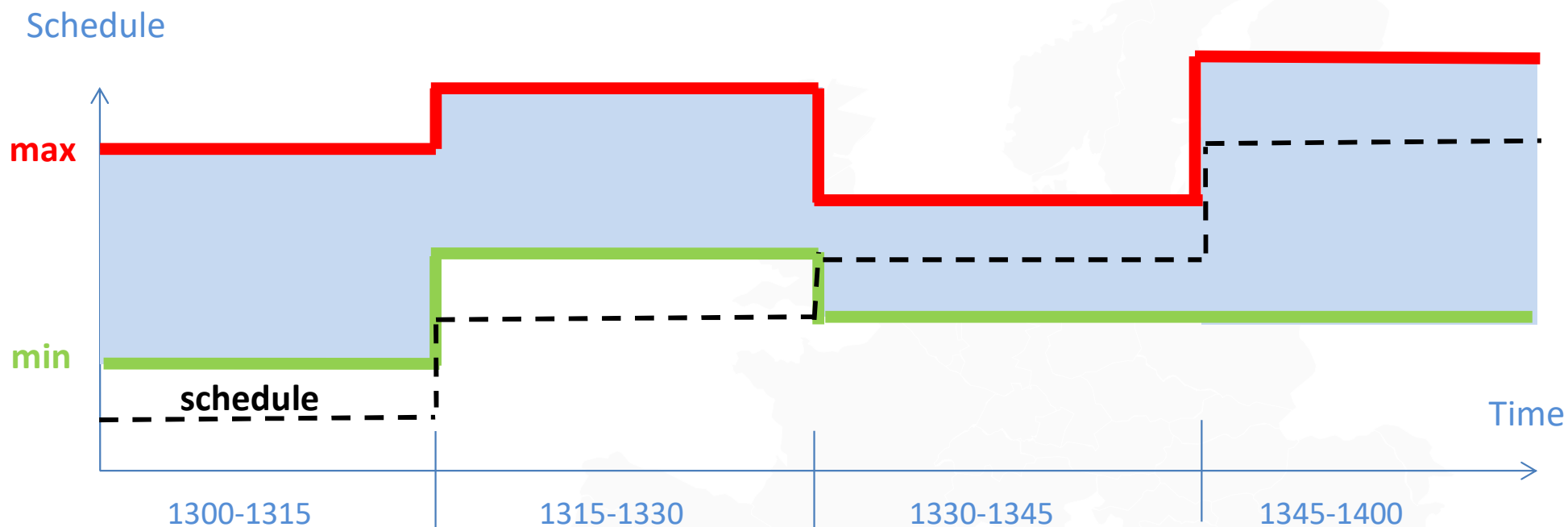
Negative number implies a counter-flow in the opposite direction UK>FR. Hence, algorithm has freedom to allocate anything between 0 and 400MW in direction FR>UK or anything between 0 and 200 MW in direction UK>FR.



## 4. RR inputs format description

### Optimization Algorithm – Inputs

- HVDC document: both min and max may sometimes situated above schedule



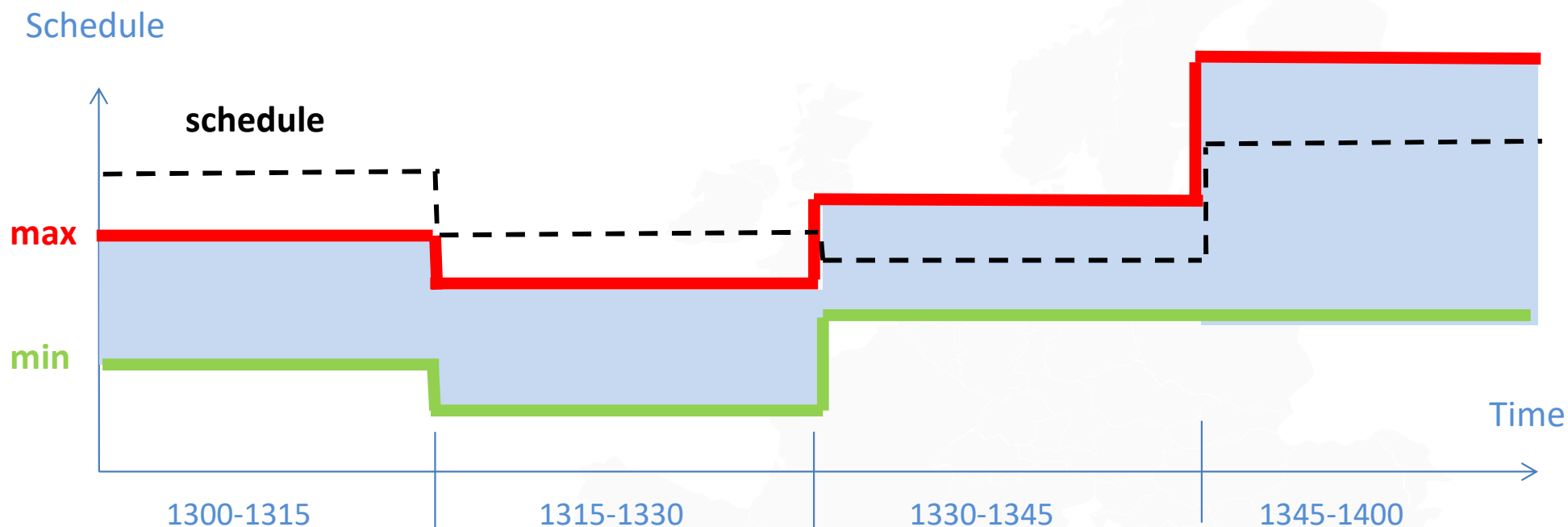
FR>UK step 1300-1315: Schedule = 100MW, min=300MW, max=700MW

Means that algorithm must allocate flow in direction FR>UK in the range from 200MW up to 600MW, i.e. algorithm must at least allocate 200MW and not more than 600MW in direction FR>UK.

## 4. RR inputs format description

### Optimization Algorithm – Inputs

- HVDC document: both min and max may sometimes situated below schedule



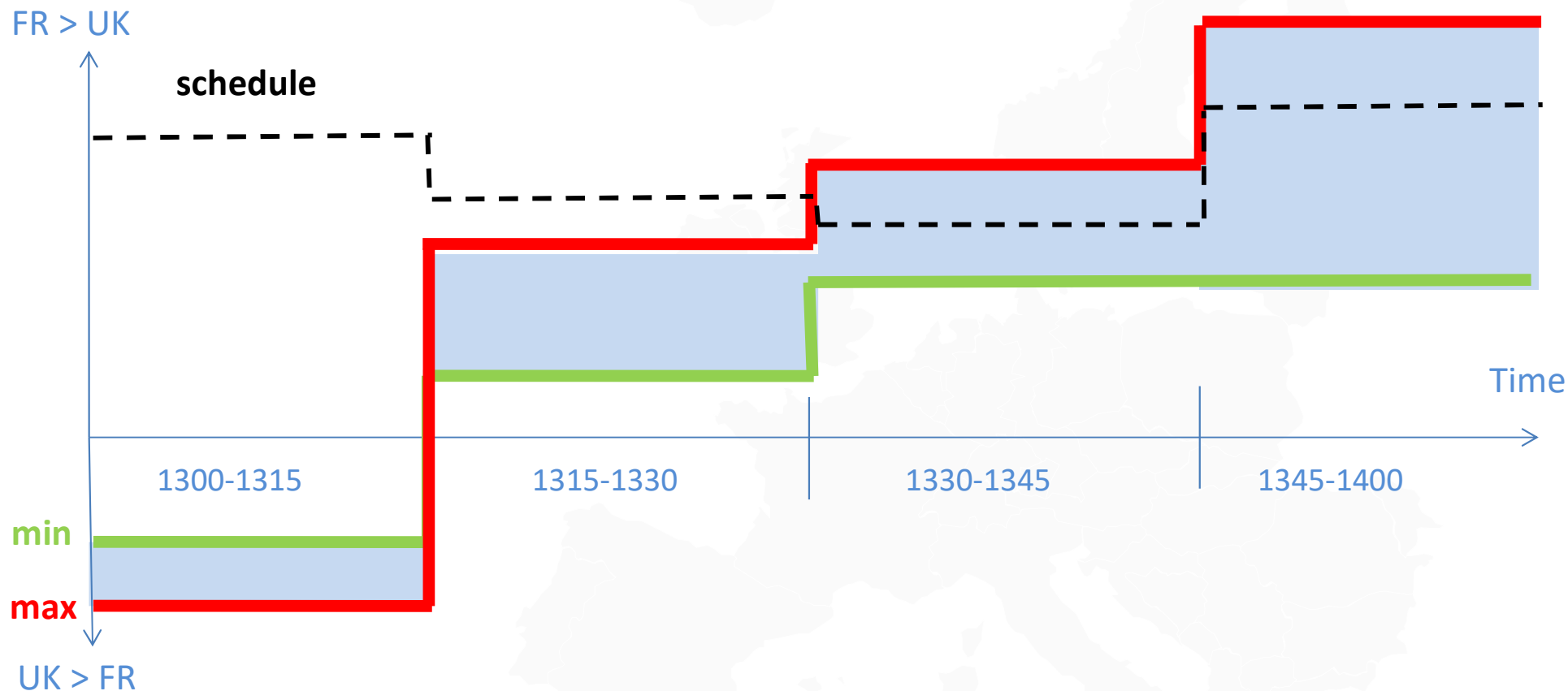
FR>UK step 1300-1315: Schedule = 700MW, min=200MW, max=600MW

Means that algorithm must allocate flow in opposite direction UK>FR in the range 100MW up to 500MW, i.e. algorithm must at least allocate 100MW and not more than 500MW in direction UK>FR.

## 4. RR inputs format description

### Optimization Algorithm – Inputs

- HVDC document: min and max may be situated in opposite direction



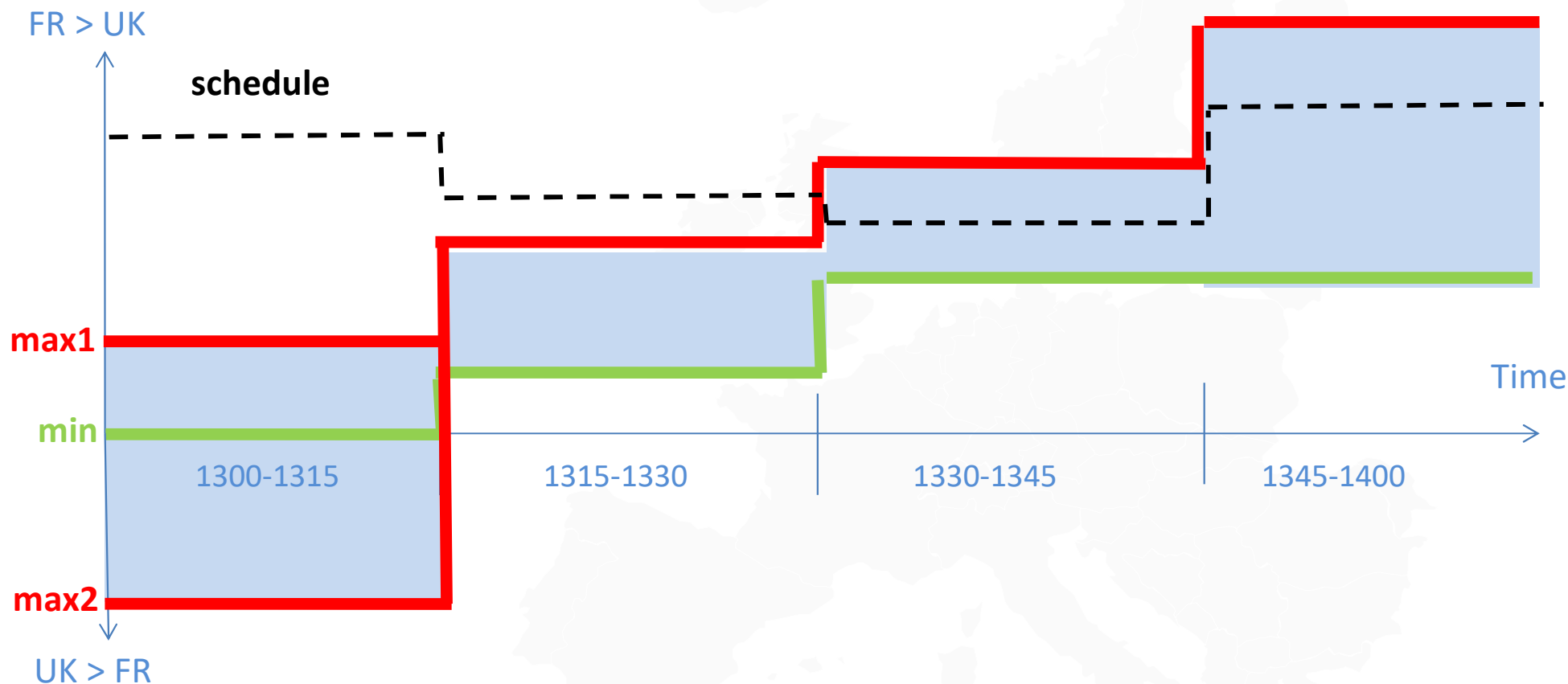
Step 1300-1315: Schedule = 700MW FR>UK, min=400MW & max=600MW UK > FR

Means that algorithm must allocate at least 1100MW and not more than 1300MW in opposite direction UK>FR.

## 4. RR inputs format description

### Optimization Algorithm – Inputs

- HVDC document: constraints may be provided for both directions (then min value must be zero)



Step 1300-1315: Schedule = 700MW FR>UK, max1=200MW FR > UK & max2=400MW UK > FR  
Means that algorithm must allocate at least 500MW and not more than 1100MW in direction UK>FR.

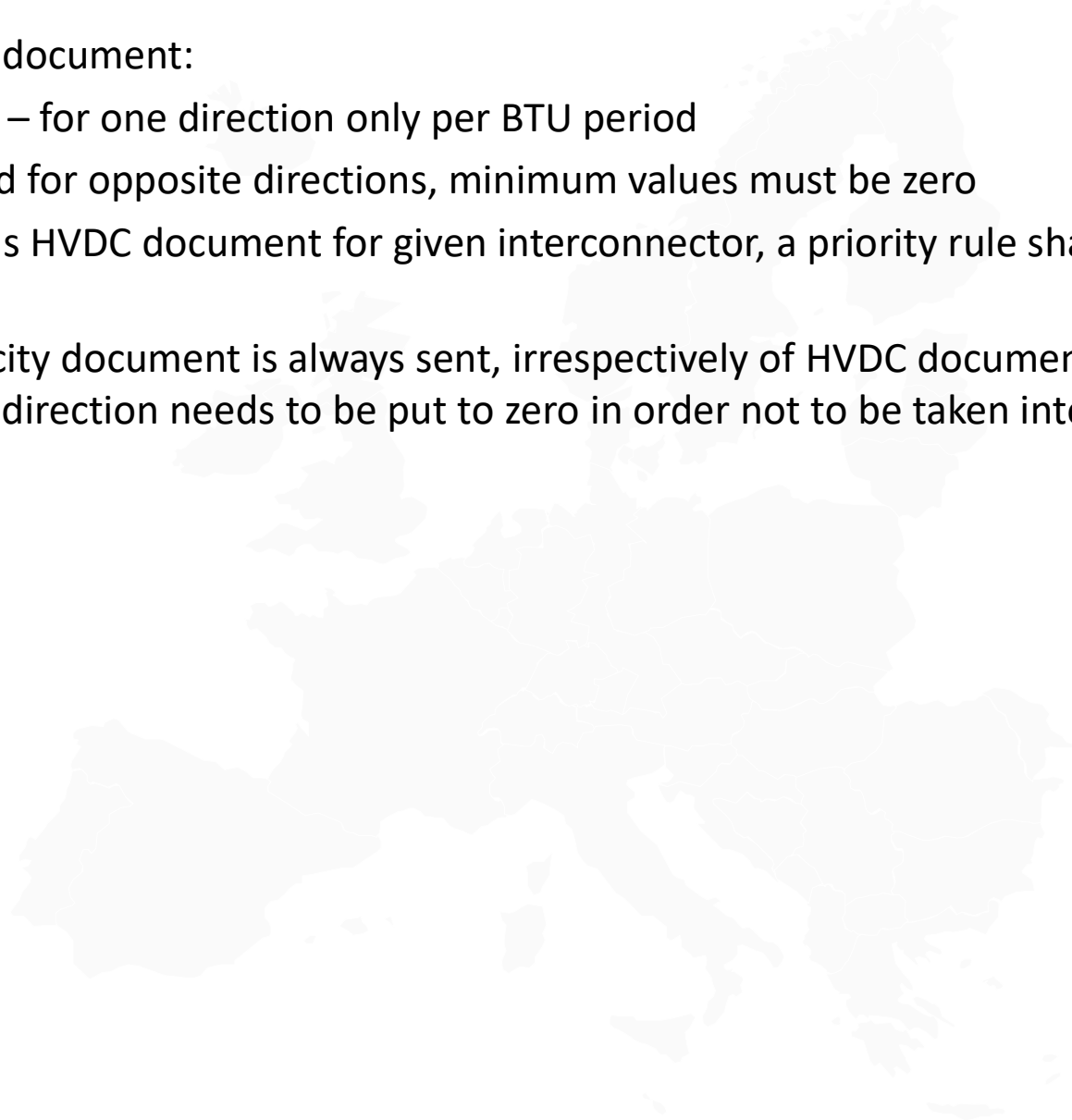
## 4. RR inputs format description

### Optimization Algorithm – Inputs

---

Further clarifications on HVDC document:

- HVDC schedule always net – for one direction only per BTU period
- If constraints are submitted for opposite directions, minimum values must be zero
- If more than one TSO sends HVDC document for given interconnector, a priority rule shall be applied
- Be aware that ATC in capacity document is always sent, irrespectively of HVDC documents (the ATC of the concerned direction needs to be put to zero in order not to be taken into account in the settlement)





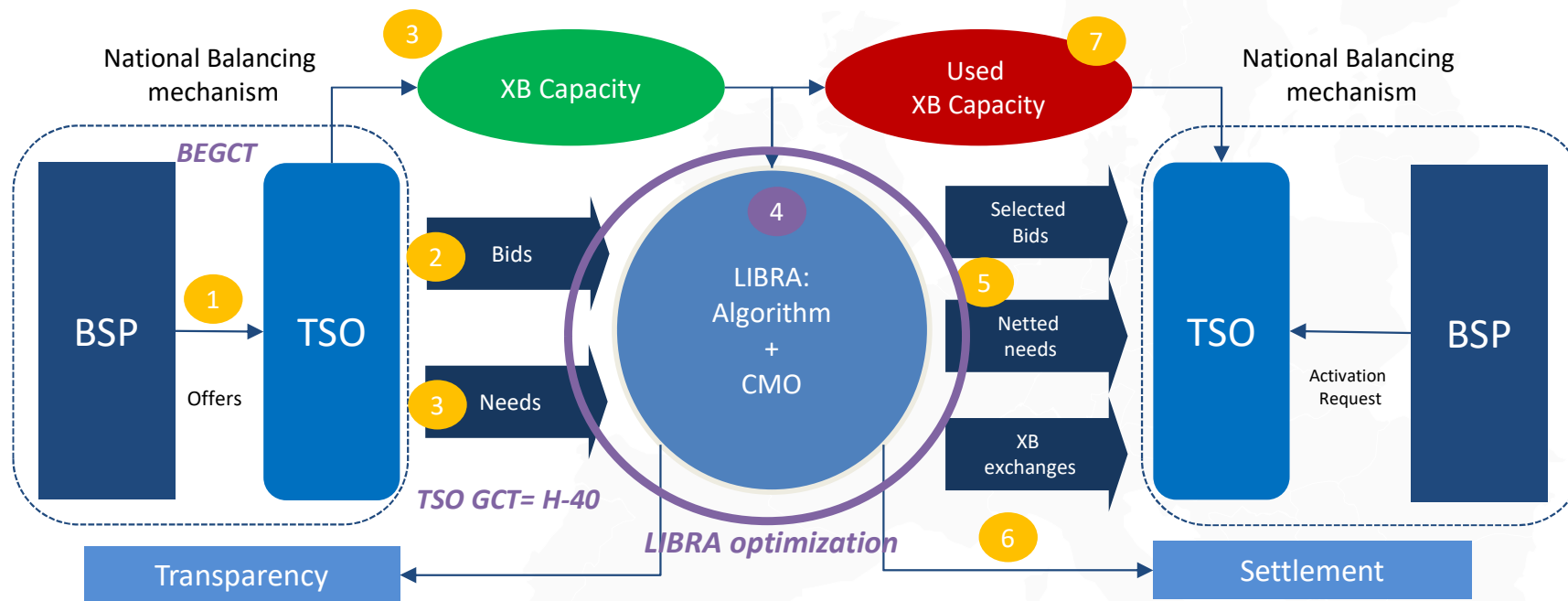
# **Topic 5: Algorithm Optimisation**

## **Function description-part 1**

# 5. Algorithm Optimisation Function description-part 1

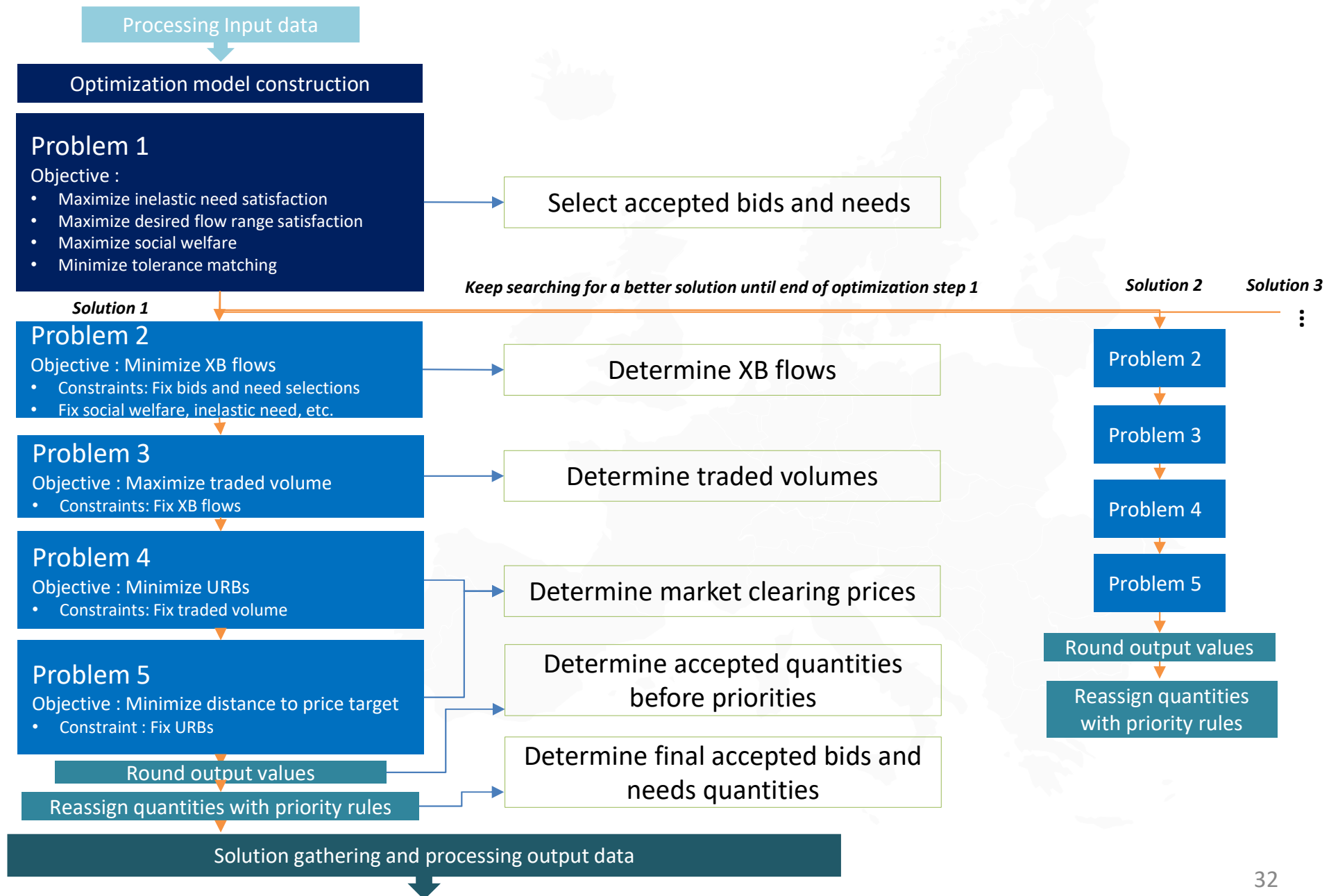
## Optimization Algorithm – Algorithm steps

How the activated bids, satisfied needs and clearing prices are determined?



# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps





## 5. Algorithm Optimisation Function description-part 1

### Optimization Algorithm – Algorithm steps

---

#### Problem 1 – basic definitions

- **Market Clearing Price (MCP)** - price determined for each Balancing Time Unit (BTU) and Bidding Zone (BZ) used for the settlement of accepted bids and satisfied needs related to the given BTU and BZ
- **Bid/need surplus** – surplus involved with the bid/need calculated under the assumption that the bid is fully accepted and settled at the MCP, e.g.:
  - for single-BTU upward bid:  $\text{surplus} = (\text{MCP} - \text{bid price}) \times \text{maximum quantity}$
  - for single-BTU downward bid:  $\text{surplus} = (\text{bid price} - \text{MCP}) \times \text{maximum quantity}$
- **Interconnector surplus** – difference between the product of imported balancing energy and MCP in importing BZ and the product of exported balancing energy and MCP in exporting BZ, e.g.:
  - for single-BTU interconnector without losses that is between importing BZ A and exporting BZ B:  $\text{surplus} = (\text{MCP in A} - \text{MCP in B}) \times \text{flow}$
- **Delta-P** – bid/need surplus divided by the bid/need maximum quantity, e.g.:
  - for single-BTU upward bid:  $\text{Delta-P} = \text{MCP} - \text{bid price}$
  - for single-BTU downward bid:  $\text{Delta-P} = \text{bid price} - \text{MCP}$

# 5. Algorithm Optimisation Function description-part 1

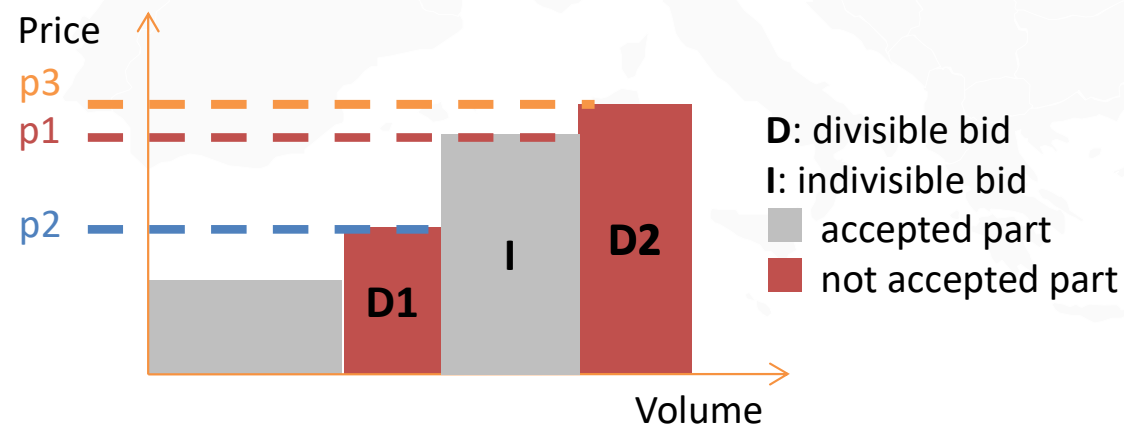
## Optimization Algorithm – Algorithm steps

### Problem 1 – basic definitions

- UAB and URB definition:
  - Depending on Delta-P value the bid/need could be:
    - in-the-money ( $\Delta P > 0$ )
    - out-of-the-money ( $\Delta P < 0$ )
    - at-the-money ( $\Delta P = 0$ )
  - **Unforeseeably accepted bid/need (UAB)** – accepted bid/need that is out-of-the-money
  - **Unforeseeably rejected bid/need (URB)** – rejected bid/need that is in-the-money

If final price is set to  $p_2$ , offer I will be unforeseeably **accepted**

If final price is set to  $p_1$ , offer D1, will be unforeseeably **rejected**



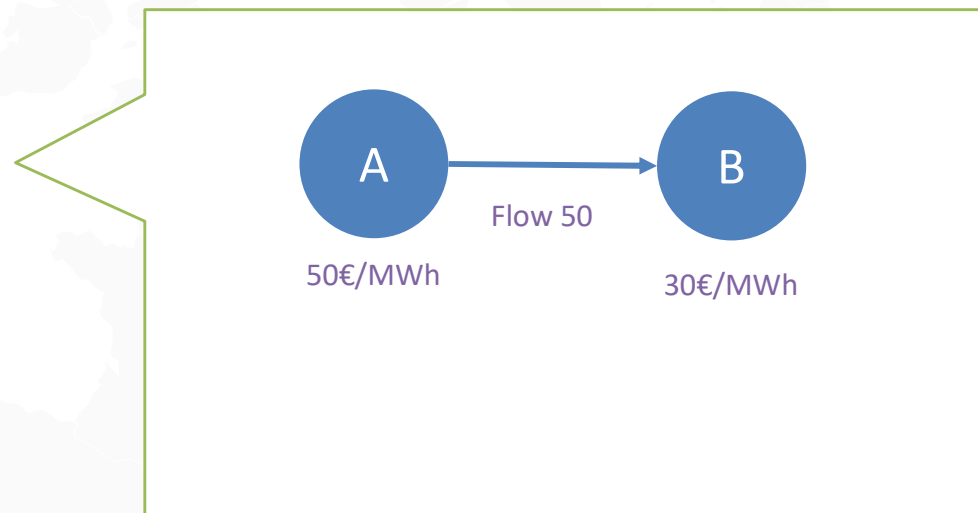
## 5. Algorithm Optimisation Function description-part 1

### Optimization Algorithm – Algorithm steps

#### Problem 1 – main constraints

- **Prevent UAB rule** – no UABs are allowed
- **Prevent adverse flows rule** – the surplus of each interconnector calculated for the **scheduling step** in the direction of the flow is non-negative

**Adverse flows:** the commercial exchange goes from the more expensive area to the less expensive one, inducing negative congestion rent.



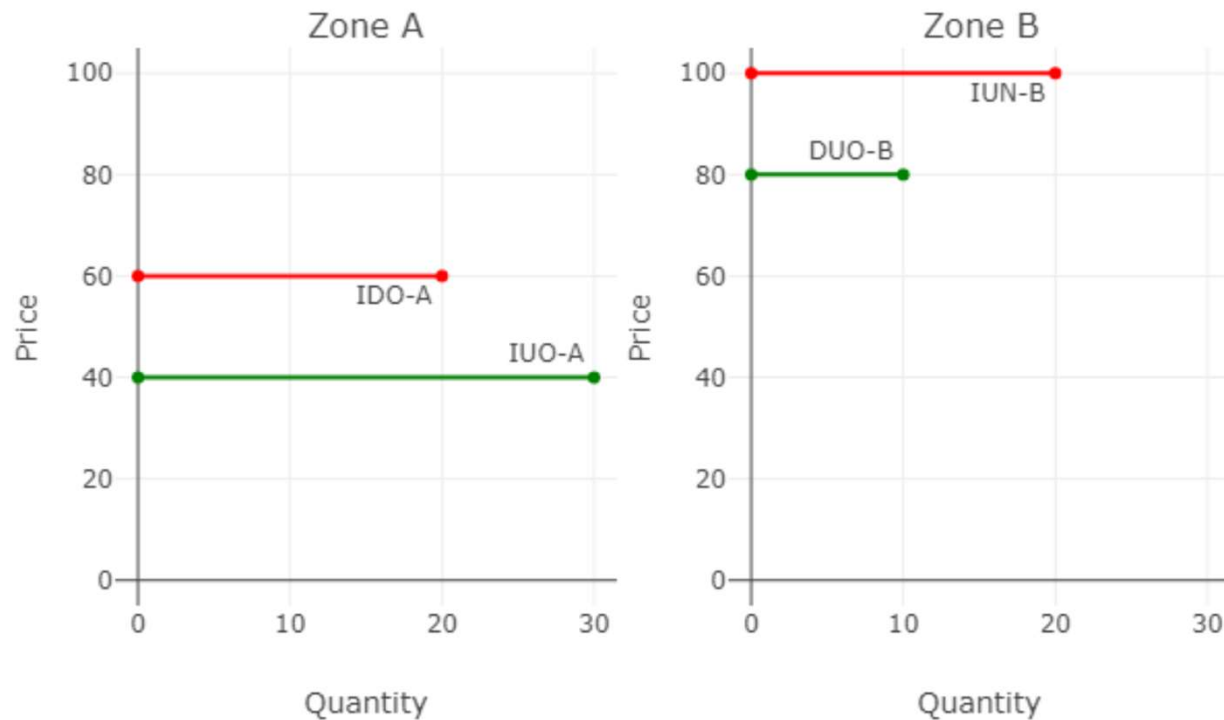
- **Enforce price convergence rule** – the surplus of each **uncongested** interconnector calculated for the **scheduling step** is zero

## 5. Algorithm Optimisation Function description-part 1

### Optimization Algorithm – Algorithm steps

#### Problem 1 – enforce price convergence example

- Two bidding zones A and B connected by single-BTU interconnector



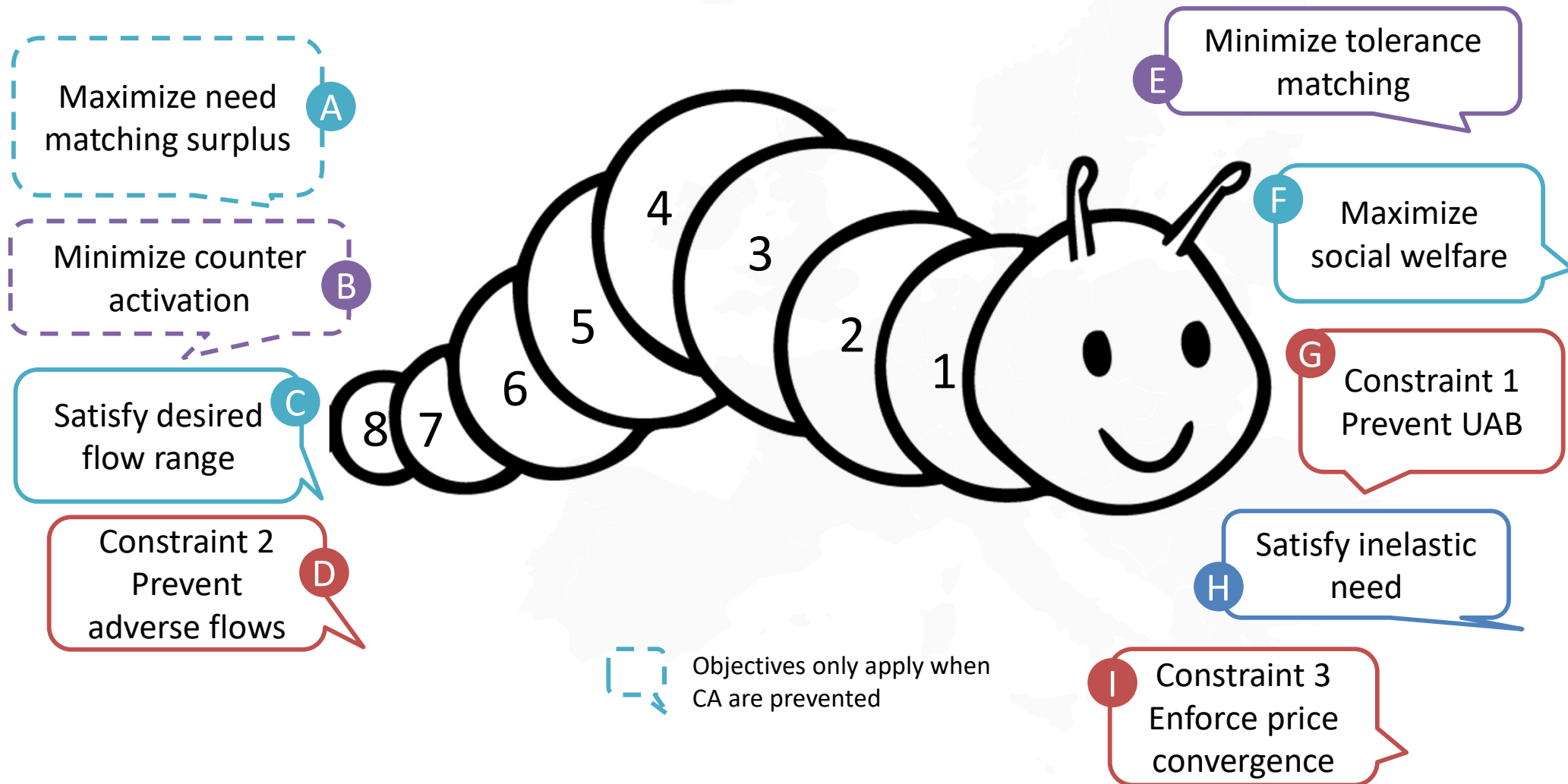
- Results:
  - DUO-B is accepted, the bids in BZ A are rejected
  - The half of IUN-B is satisfied
  - The MCP in both zones is 80€/MWh

If zones are connected only by single-BTU interconnectors without losses and there is no DFR requests, then there is no price divergence within the uncongested area

# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps

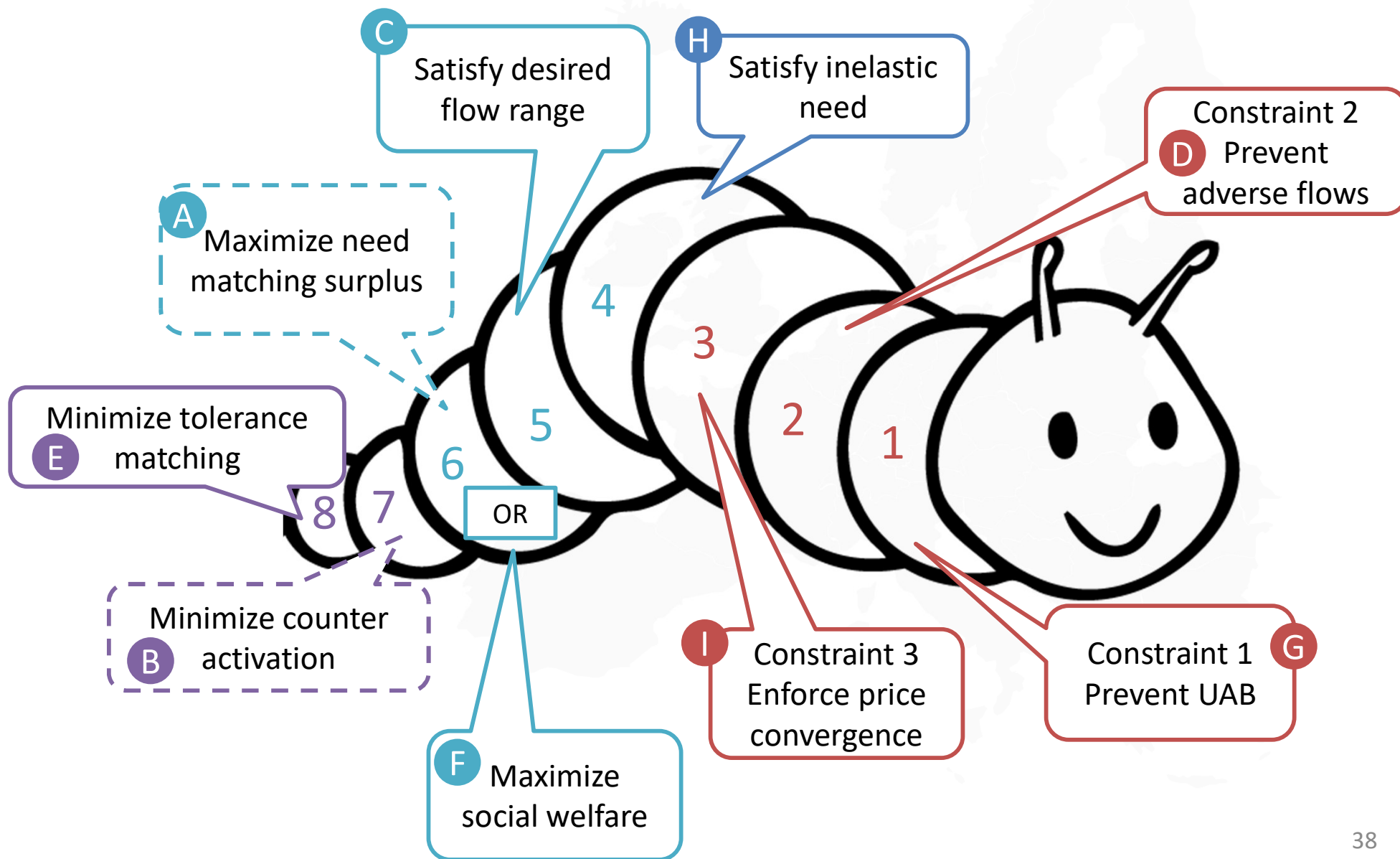
### Problem 1 objective - considered market rules



## 5. Algorithm Optimisation Function description-part 1

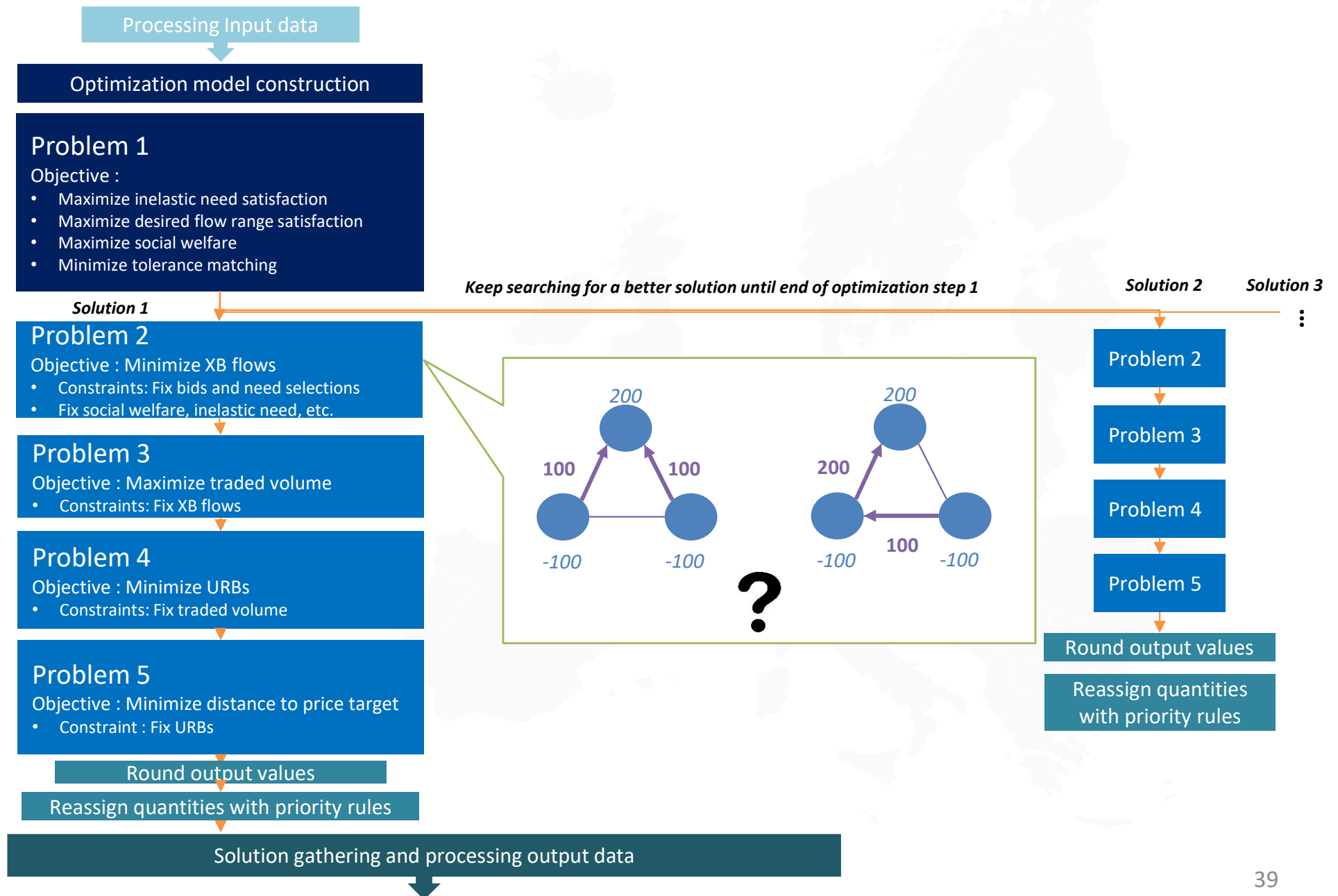
### Optimization Algorithm – Algorithm steps

Problem 1 selects the bids by applying rules in an specific order



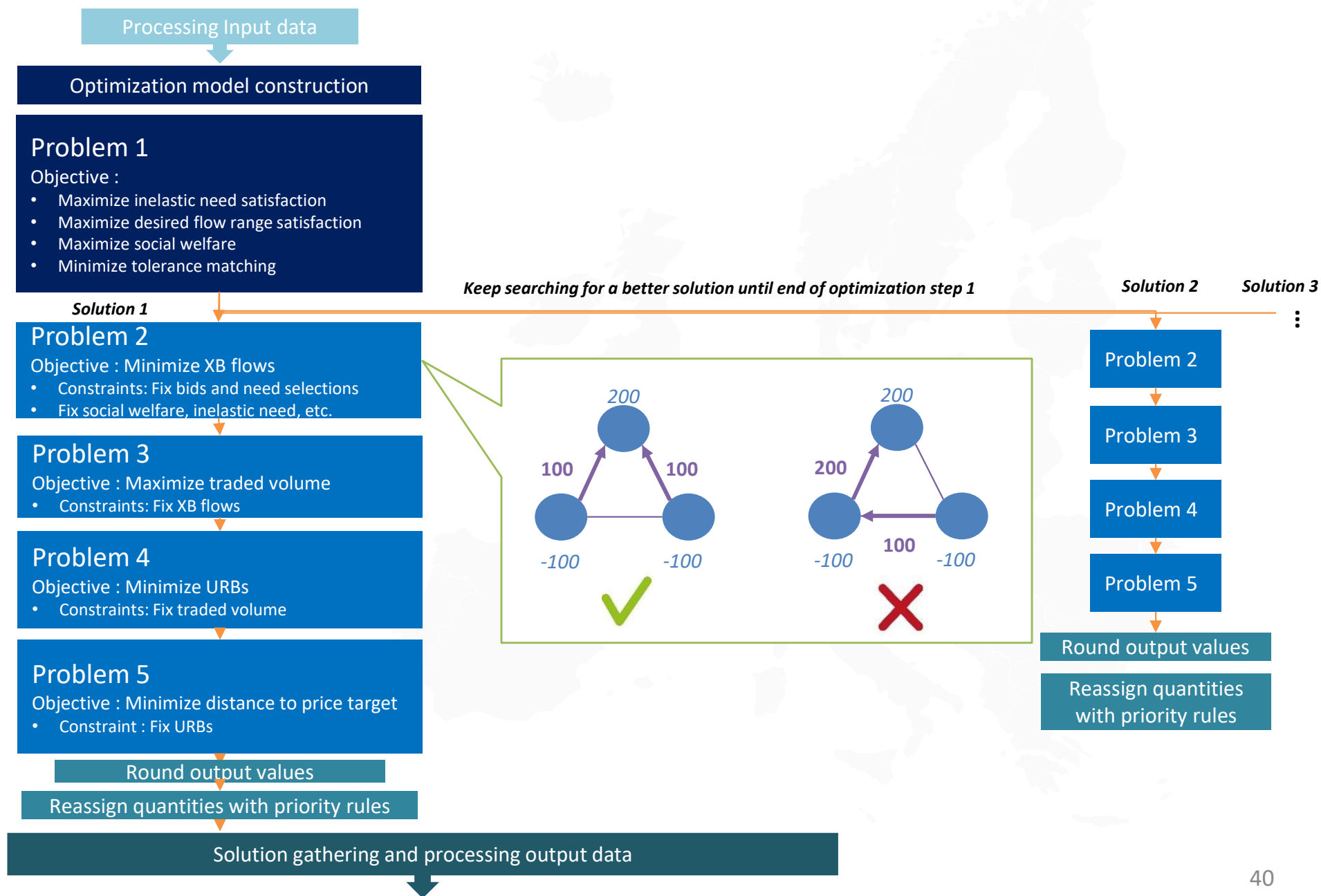
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



# 5. Algorithm Optimisation Function description-part 1

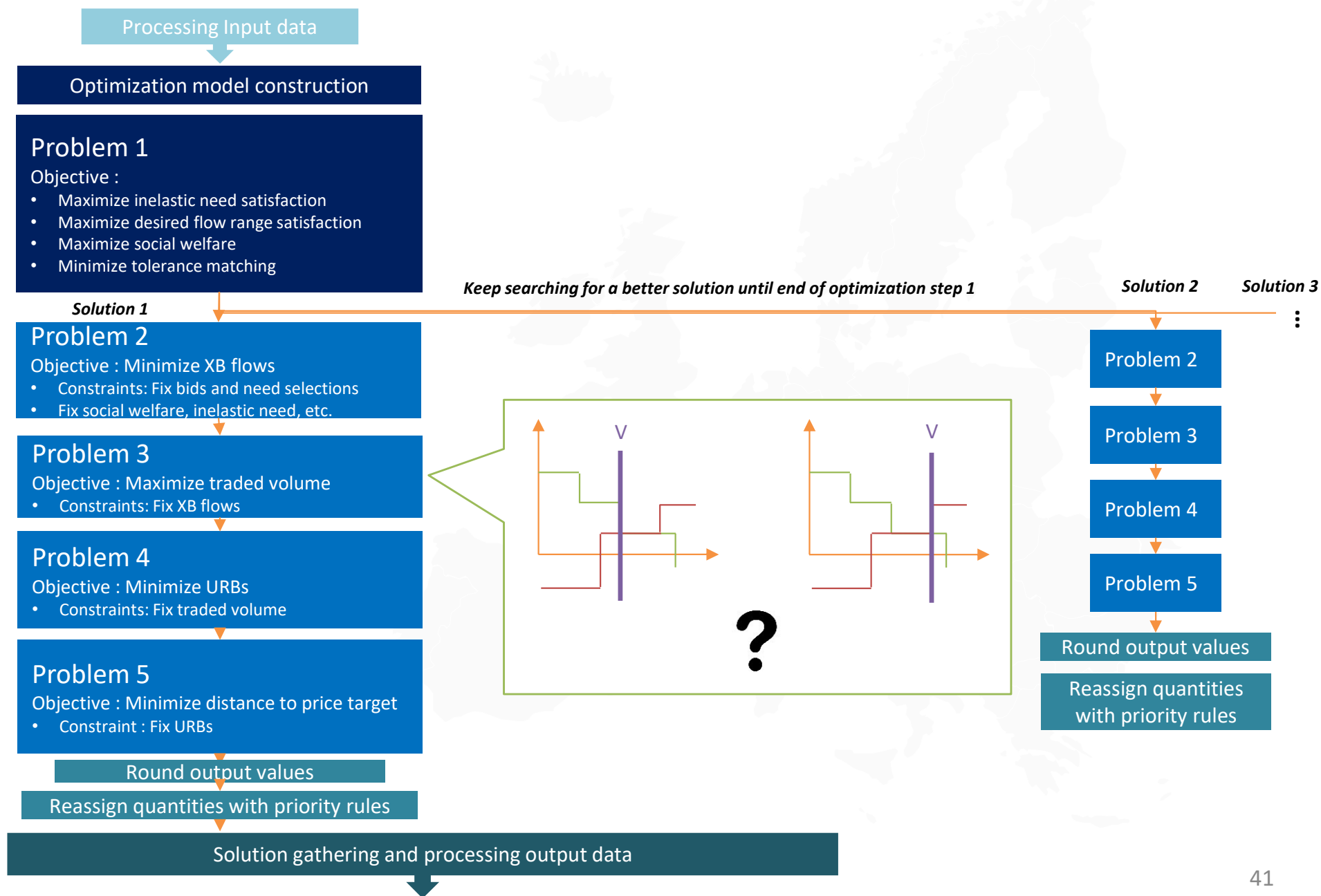
## Optimization Algorithm – Algorithm steps





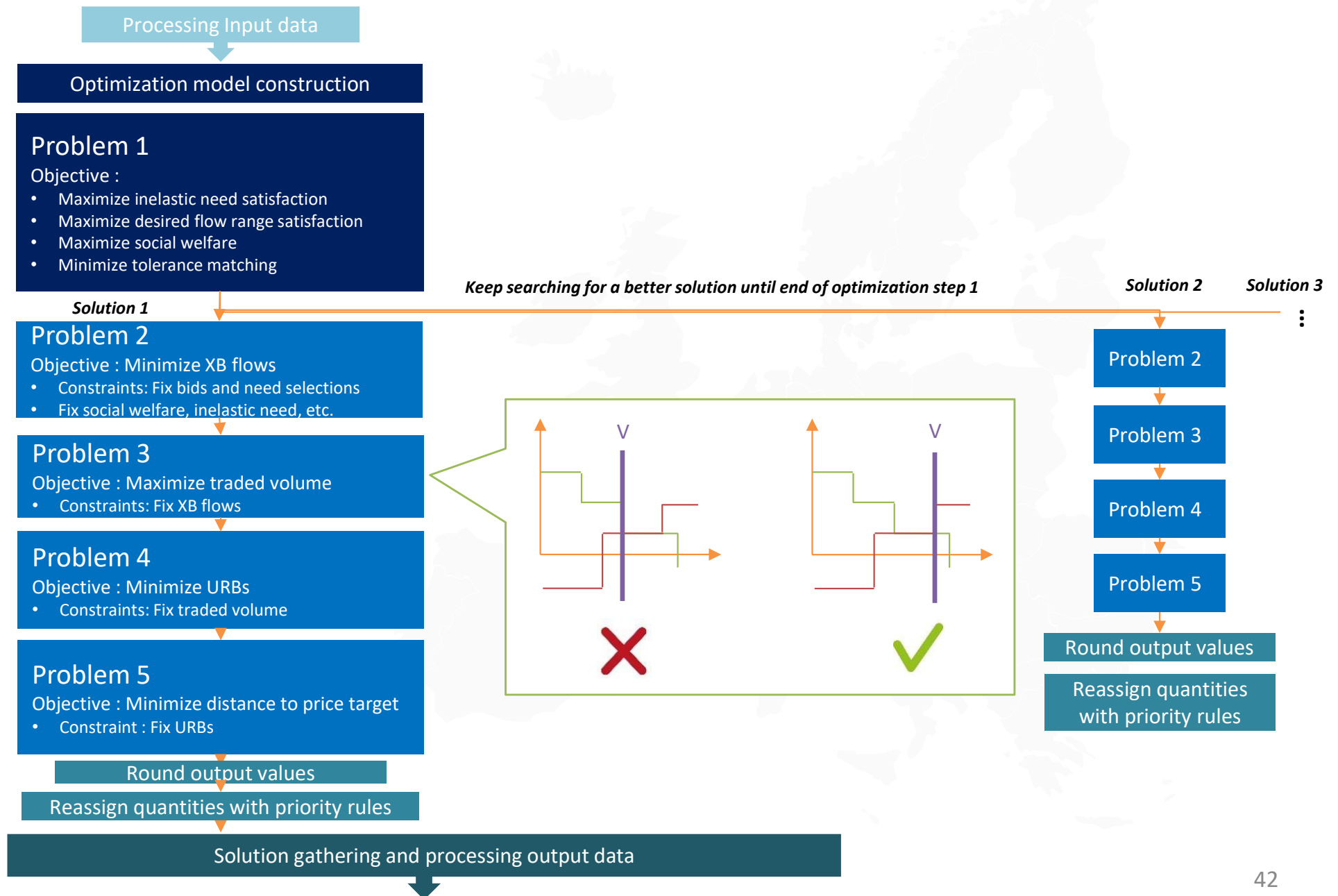
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



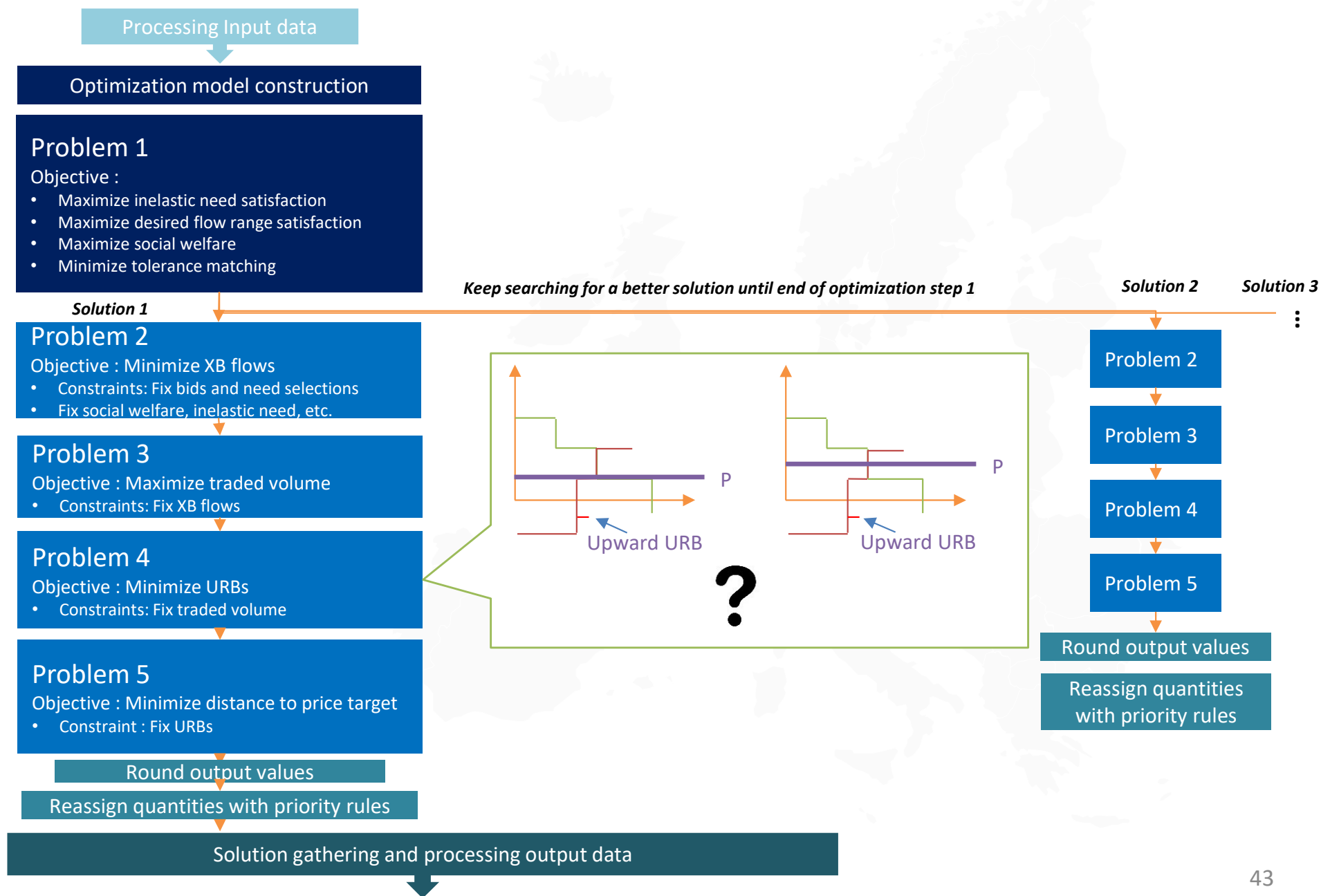
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



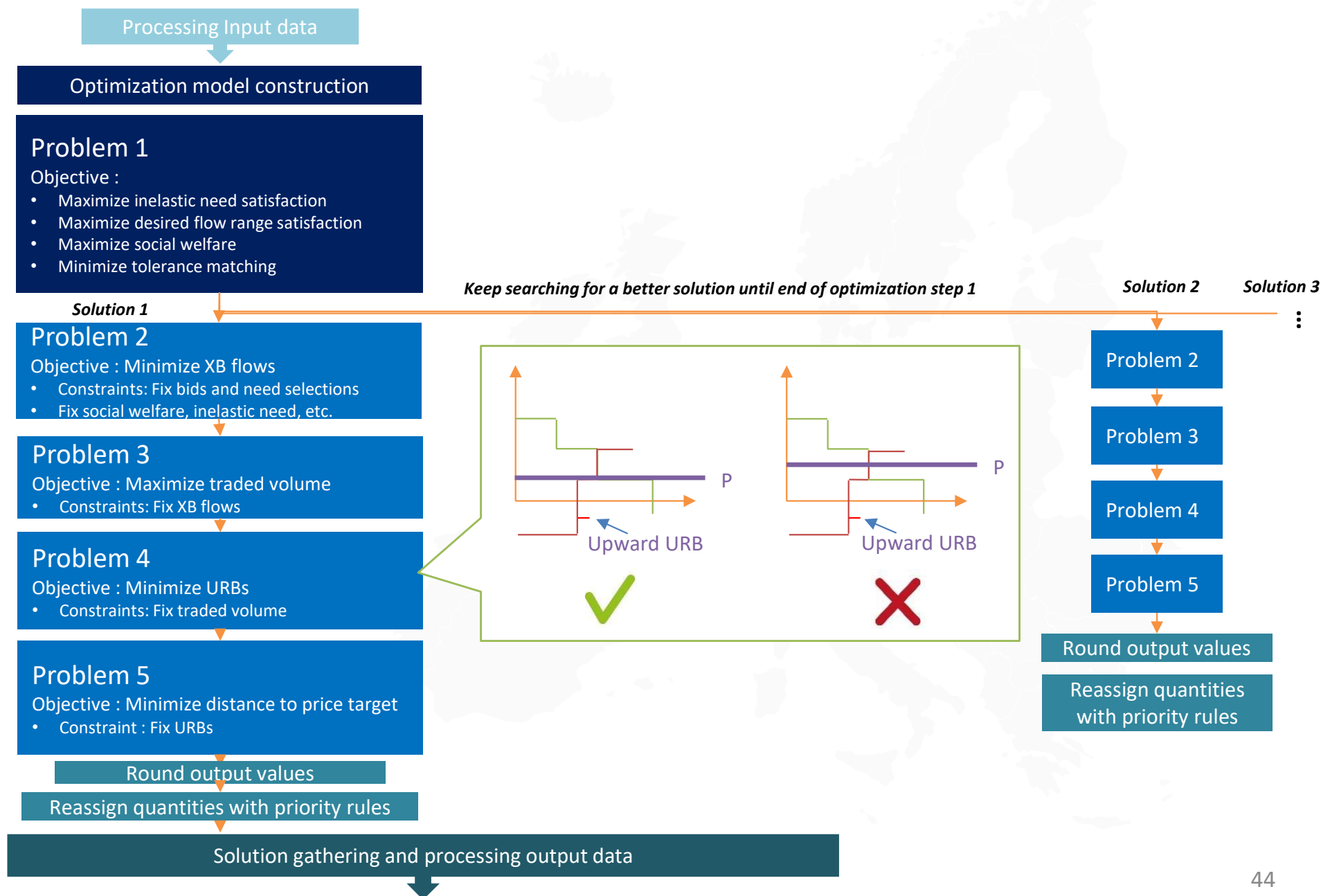
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



## 5. Algorithm Optimisation Function description-part 1

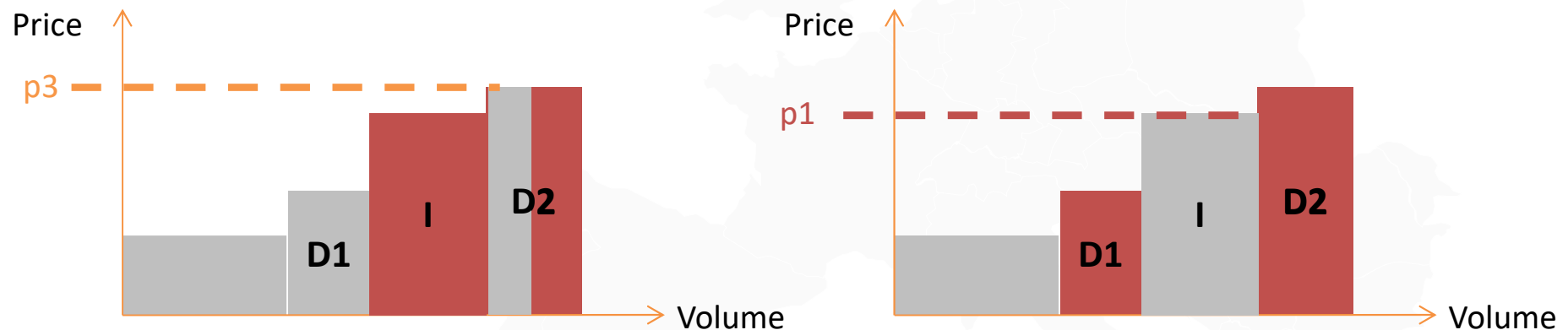
### Optimization Algorithm – Algorithm steps

#### Minimize URBs rule

- **Minimize URB rule** – the sum of positive Delta-P over the rejected (or partially accepted) bids/needs is to be minimized after the determination of satisfied needs and activated bids (this rule affects only the price determination)

Depending on which solution gives the higher social welfare LIBRA will

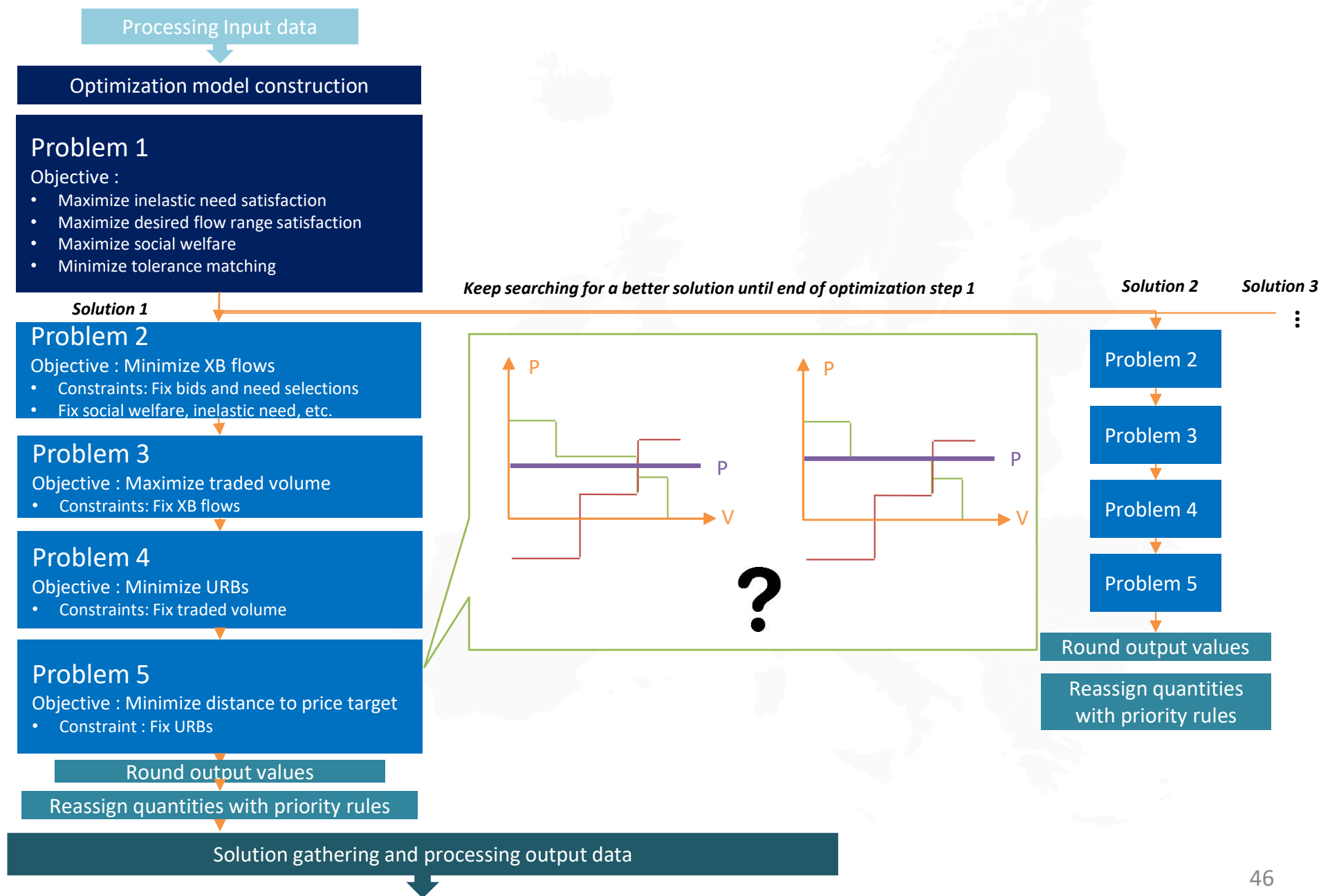
- reject offer **I**, accept offers **D1** and **D2**, and put the price at **p3**, or
- reject offers **D1** and **D2**, accept offer **I**, and put the price at **p1**.



**D:** divisible bid  
**I:** indivisible bid  
■ accepted part  
■ not accepted part

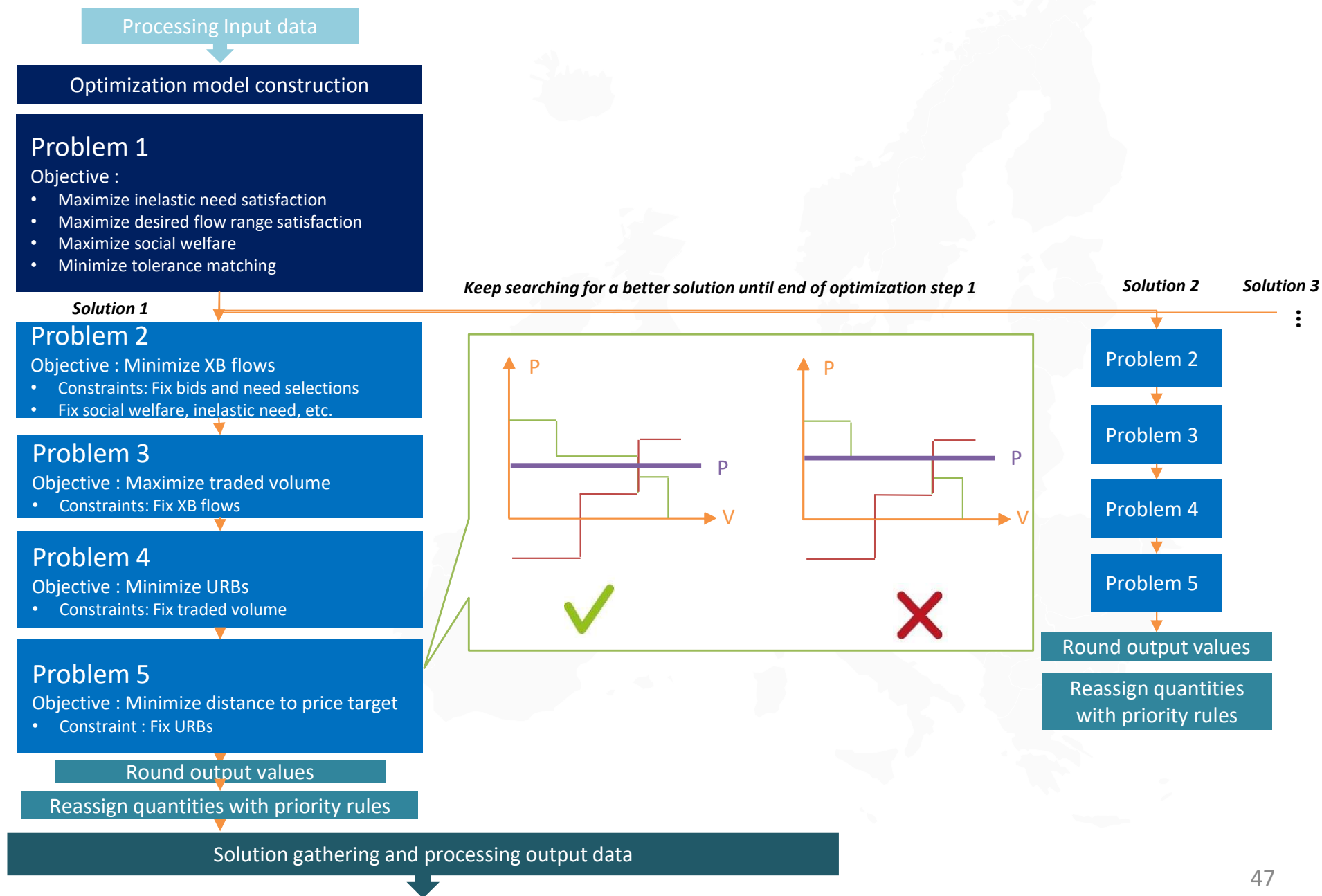
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



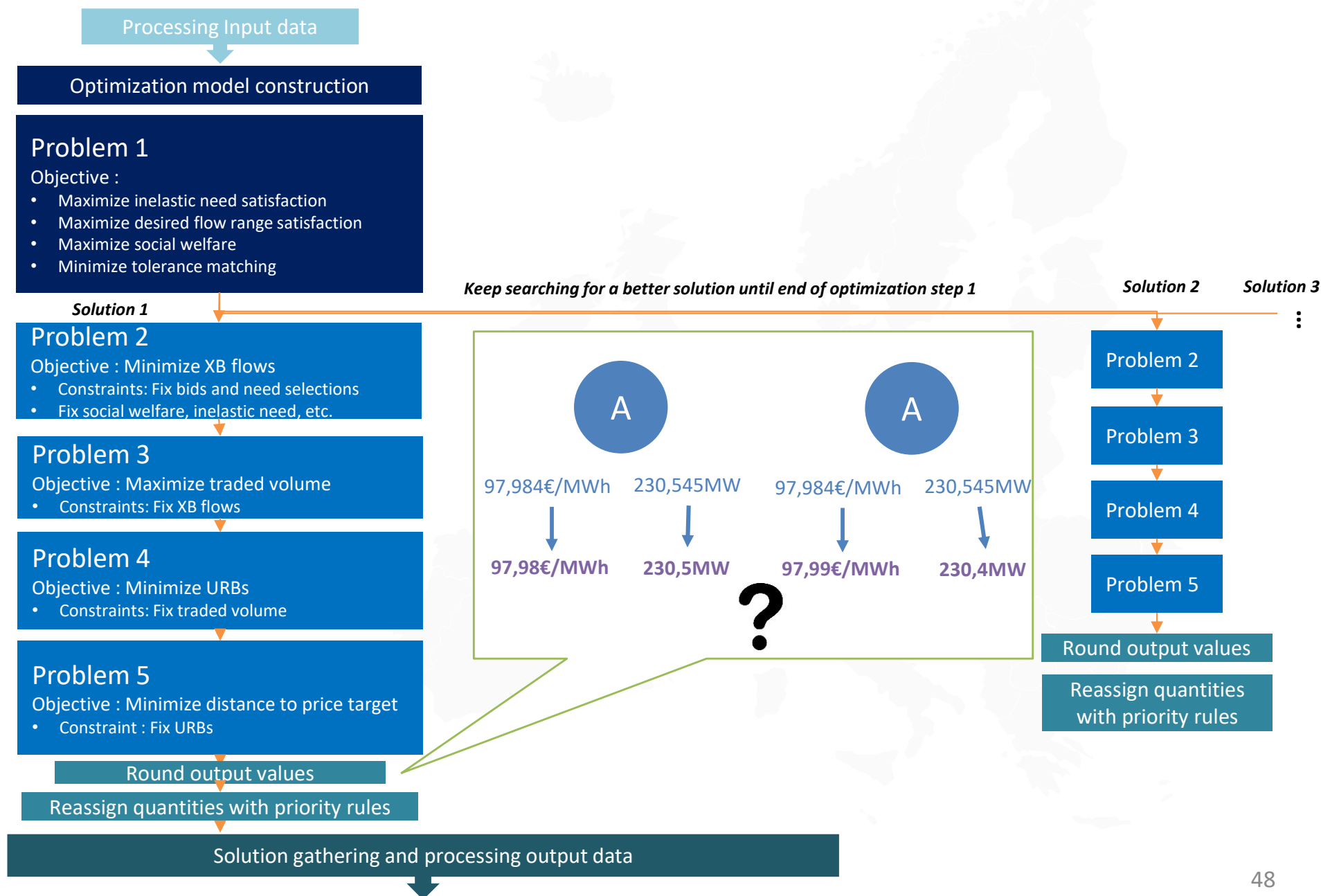
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



# 5. Algorithm Optimisation Function description-part 1

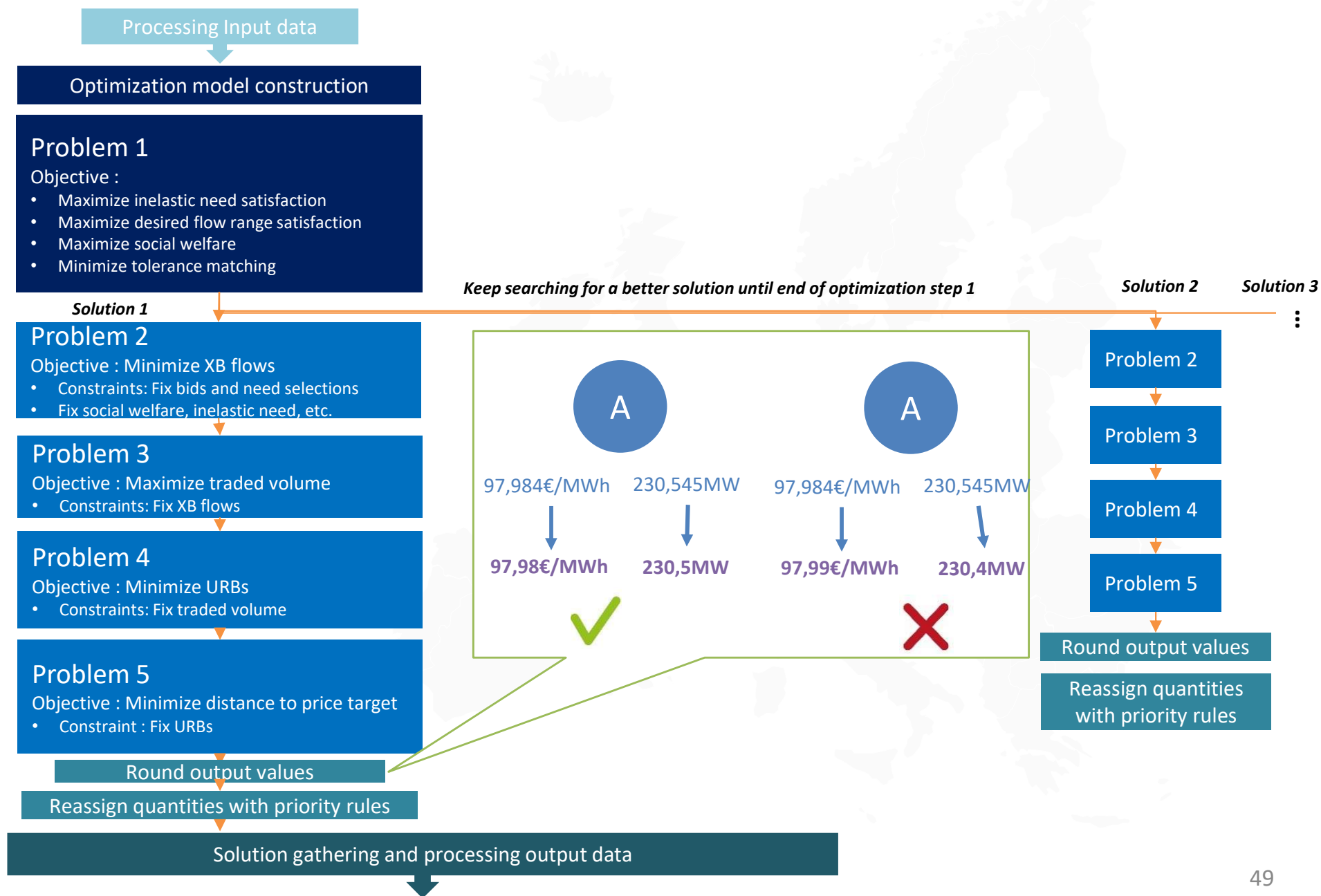
## Optimization Algorithm – Algorithm steps





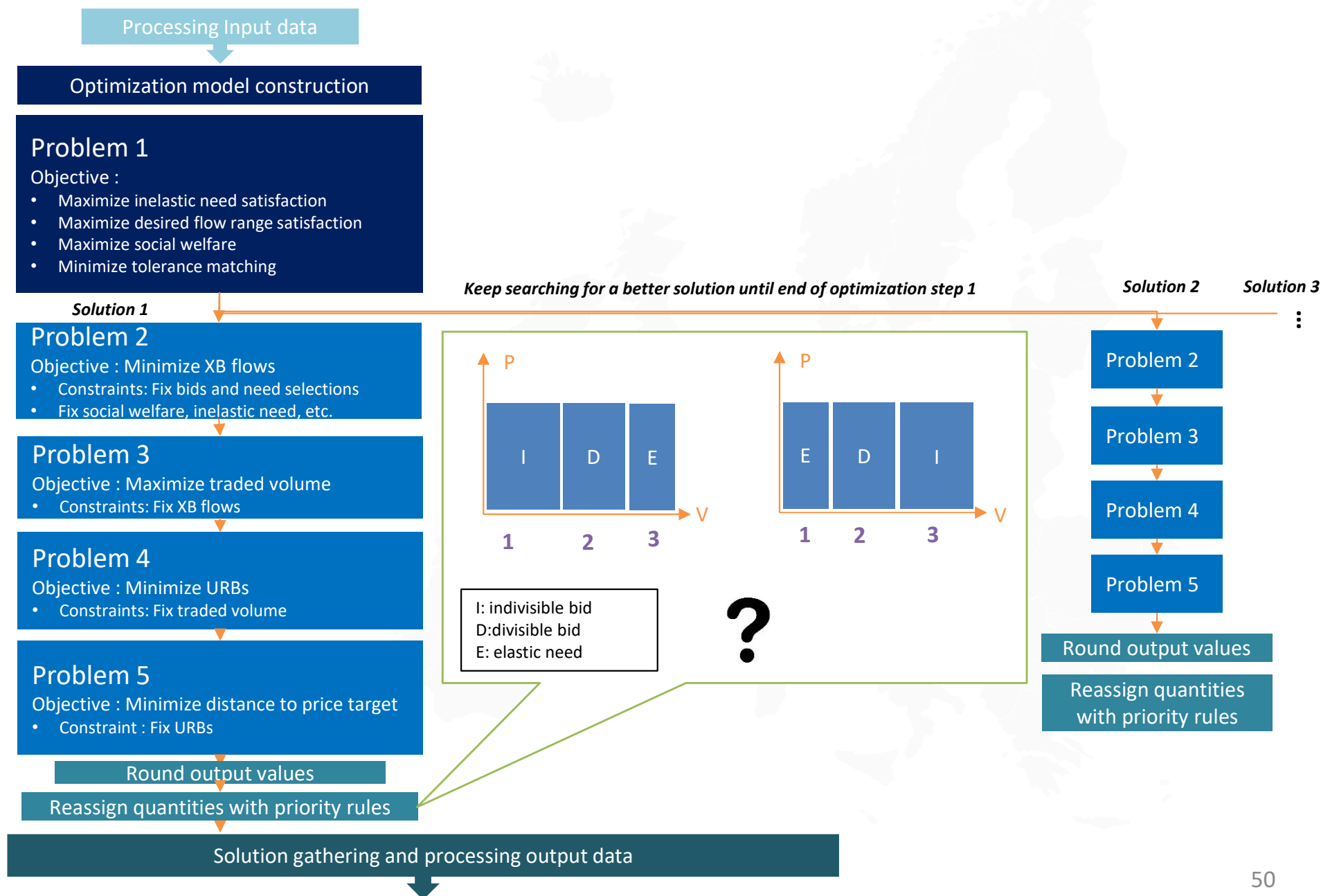
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



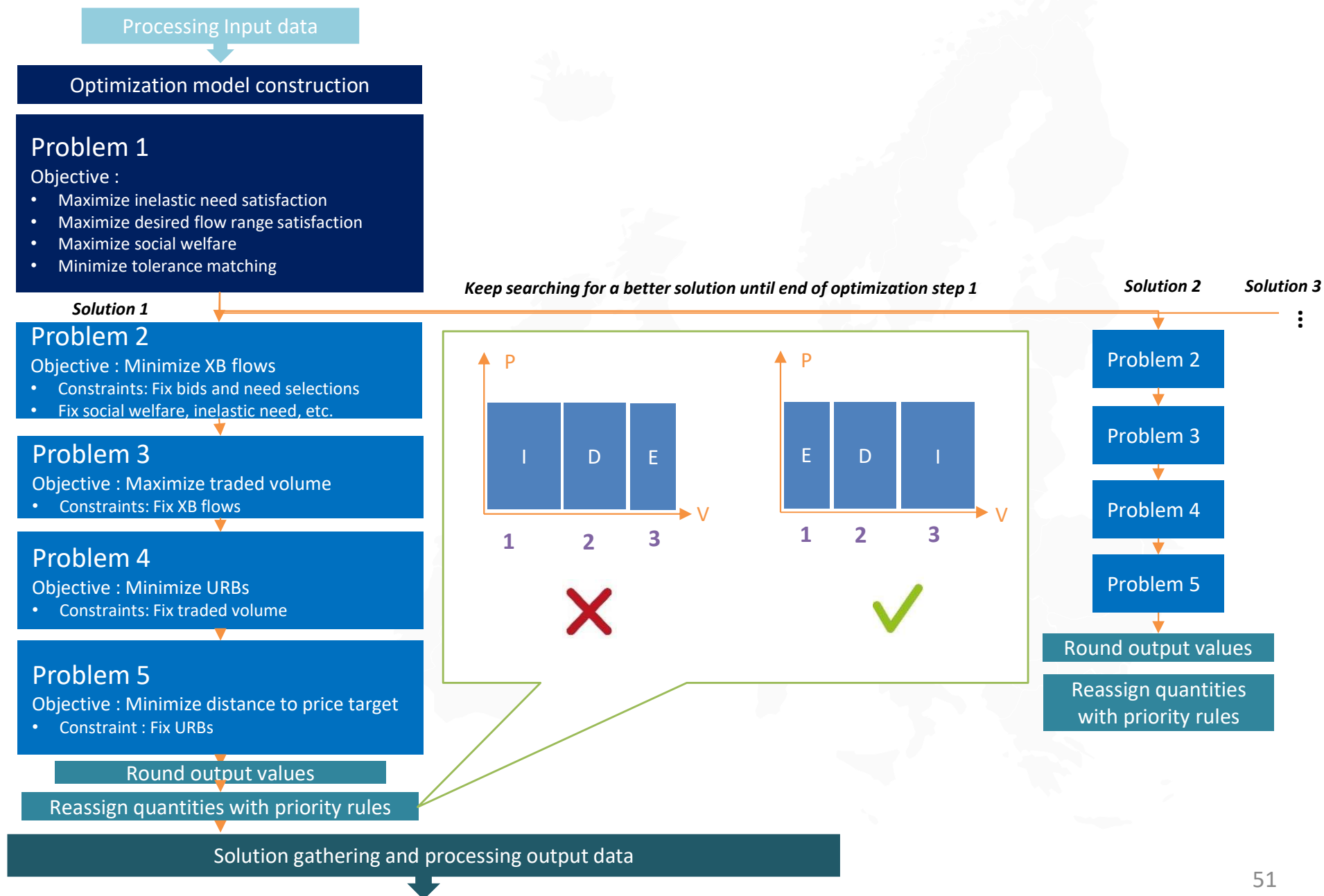
# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps



# 5. Algorithm Optimisation Function description-part 1

## Optimization Algorithm – Algorithm steps





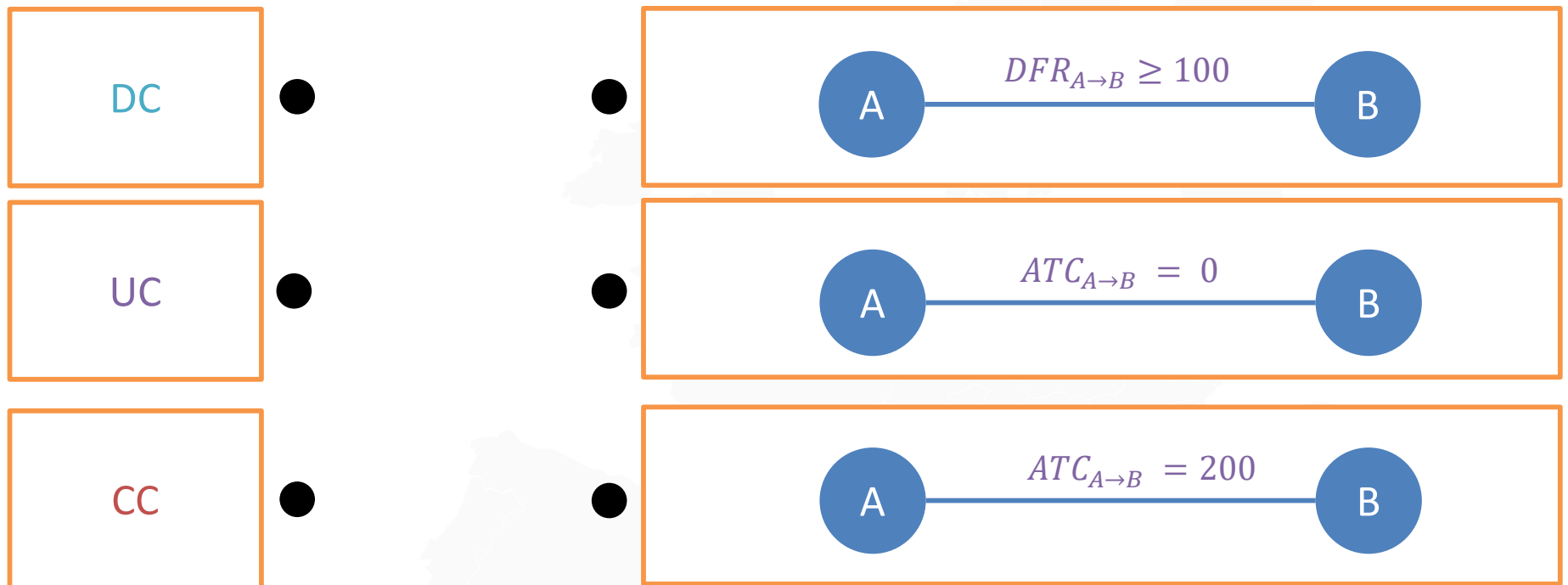
# **Topic 5: Algorithm Optimisation**

## **Function description-part 2**

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

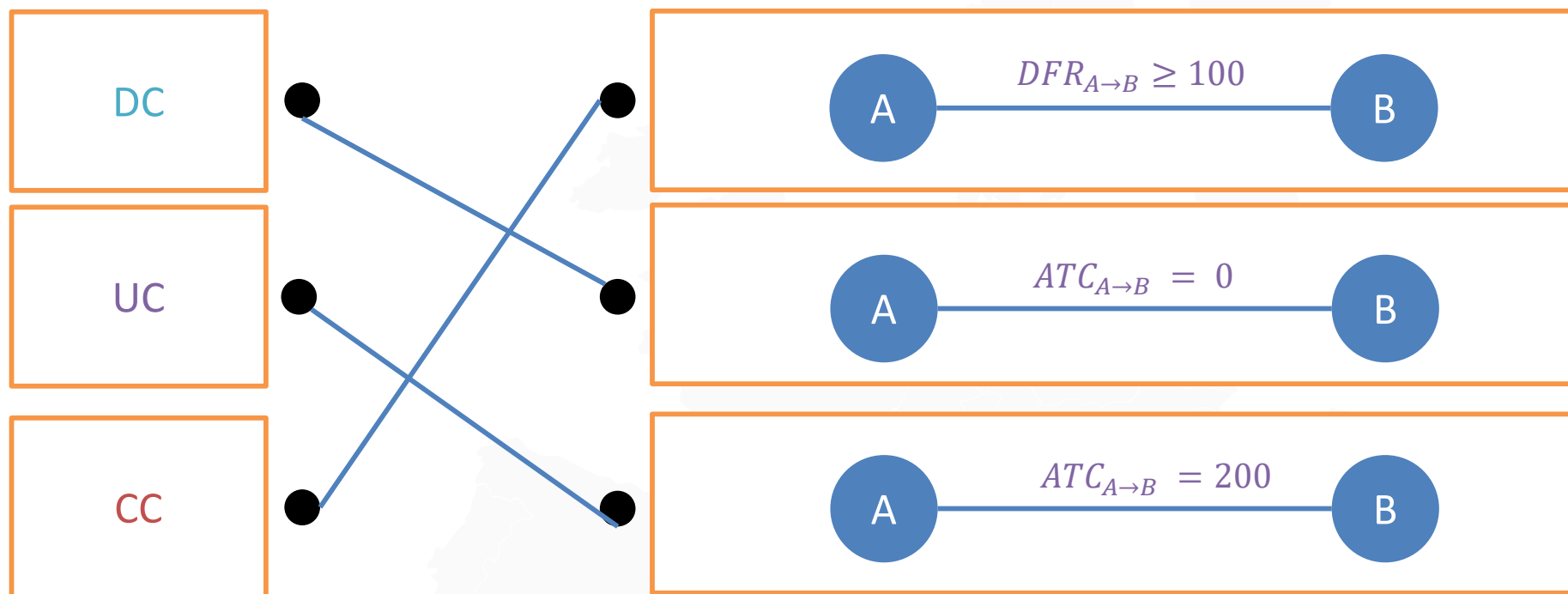
What are the 3 optimization modes?



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

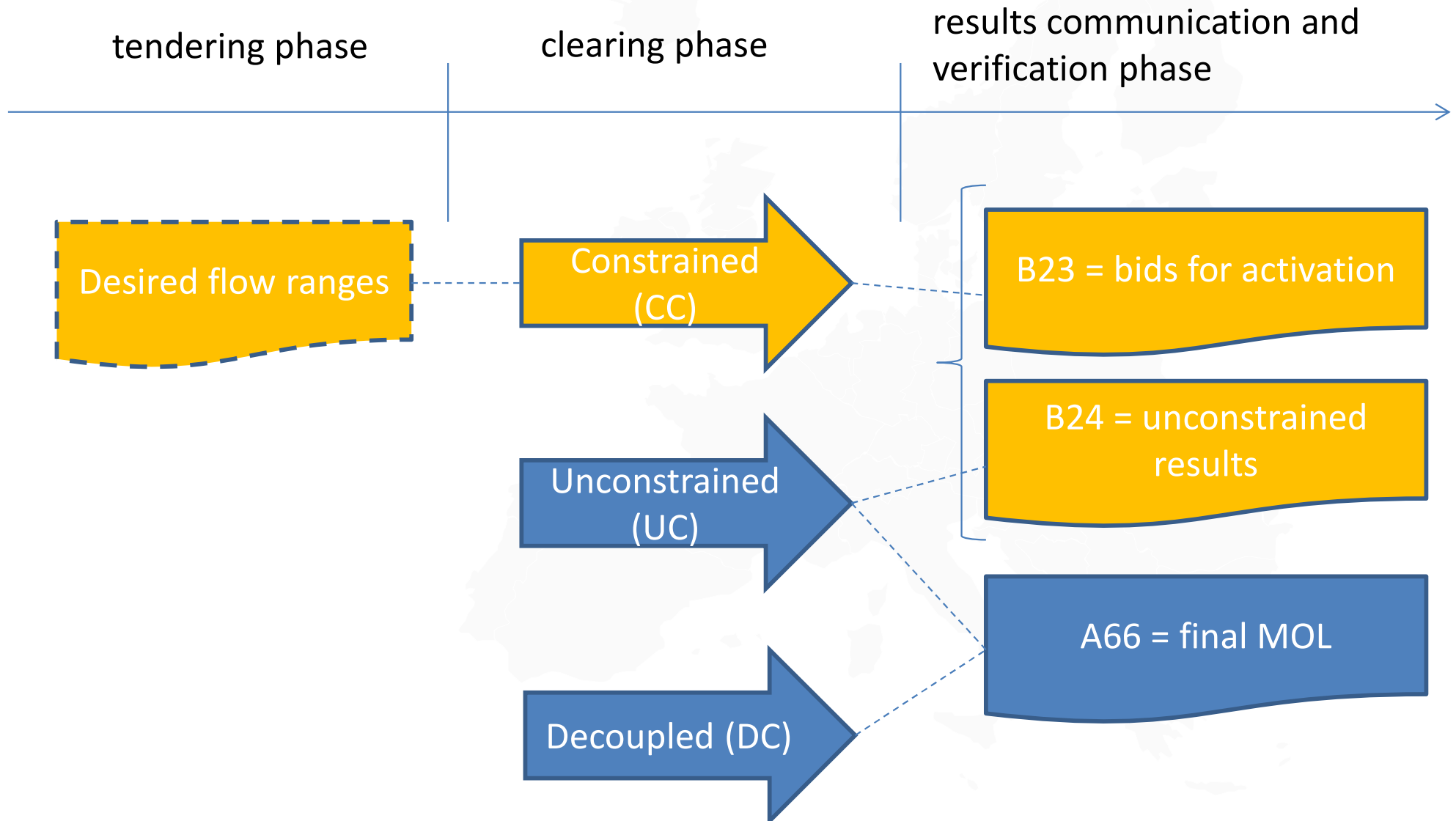
What are the 3 optimization modes?



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

Three runs at the same time: UC, DC, CC



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Possible solutions

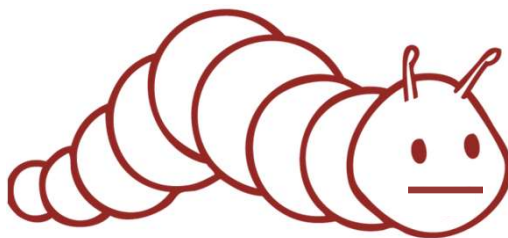
---

Three types of output are possible



#### Optimal solution found for UC

*Meaning feasible solution and no better solution exist in terms of welfare*



#### Not optimal solution found for UC

*Meaning feasible solution exists  
Max calculation time has been reached*



#### No solution found for UC

*Meaning no feasible solution:*

- Timed out
  - Process stopped manually
  - Some part of the platform failed
- > Use decoupled (DC) fallback solution*



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability

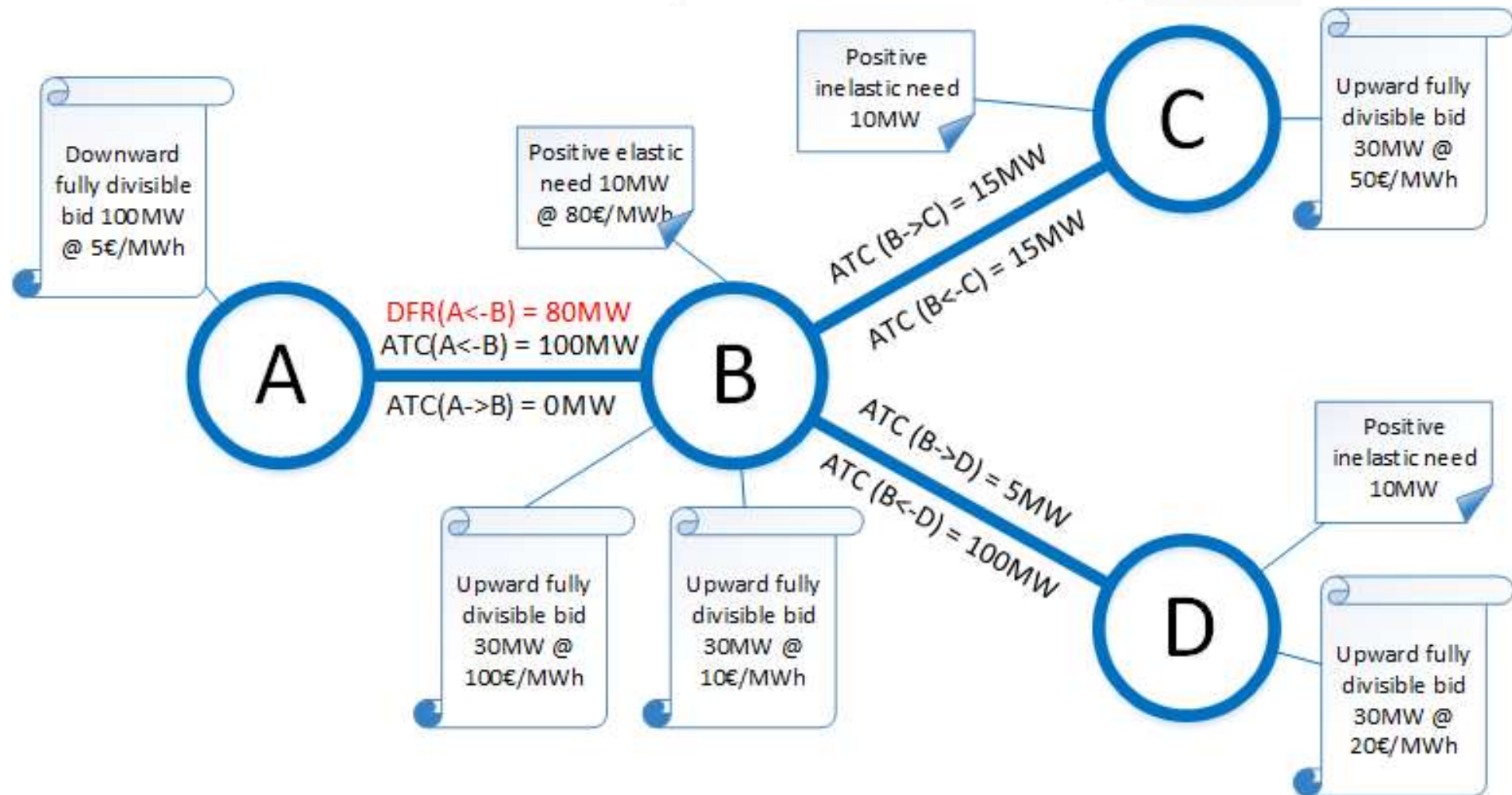
---

- CC run takes into account desired flow ranges at interconnectors:
  - Prevent adverse flow rule and enforce price convergence rule do not apply to the borders on which the desired flow is defined
- UC run does not consider the desired flow ranges
- Both runs determine the accepted volumes of bids/needs and MCPs, but only accepted volumes of bids/needs from CC and the MCPs from the UC run are binding:
  - The bids/needs accepted in CC but not accepted in UC are considered as to be activated because of system constraint purpose and are settled according to pay-as-bid rule
  - The bids/needs rejected in CC but accepted in UC are considered as to be rejected because of the system constraints
  - The interconnectors congested in CC but not congested in UC will not generate congestion rent
  - The interconnectors uncongested in CC but congested in UC could generate (positive or negative) congestion rent

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability

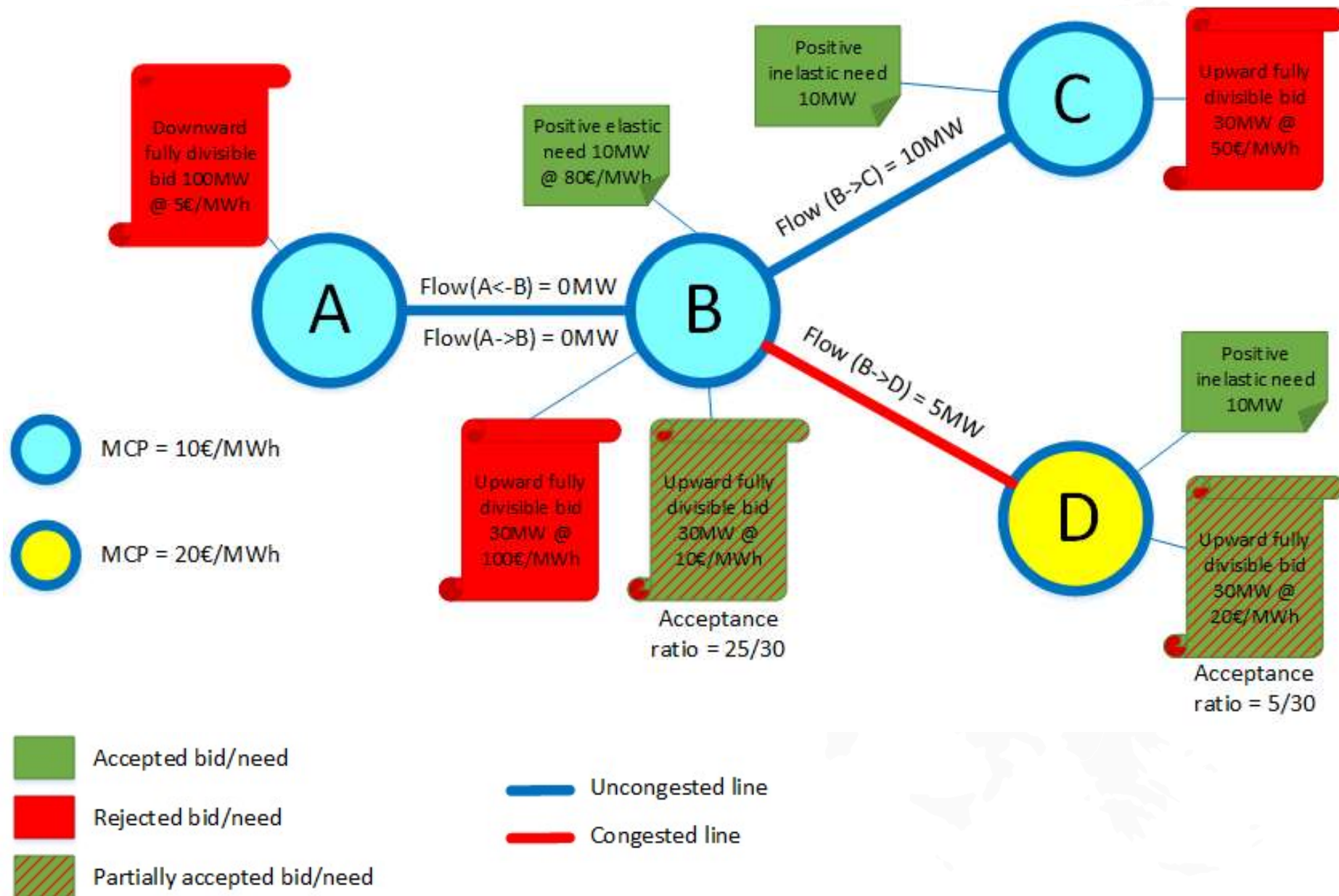
#### Example – input data



\* Positive need = Upward need

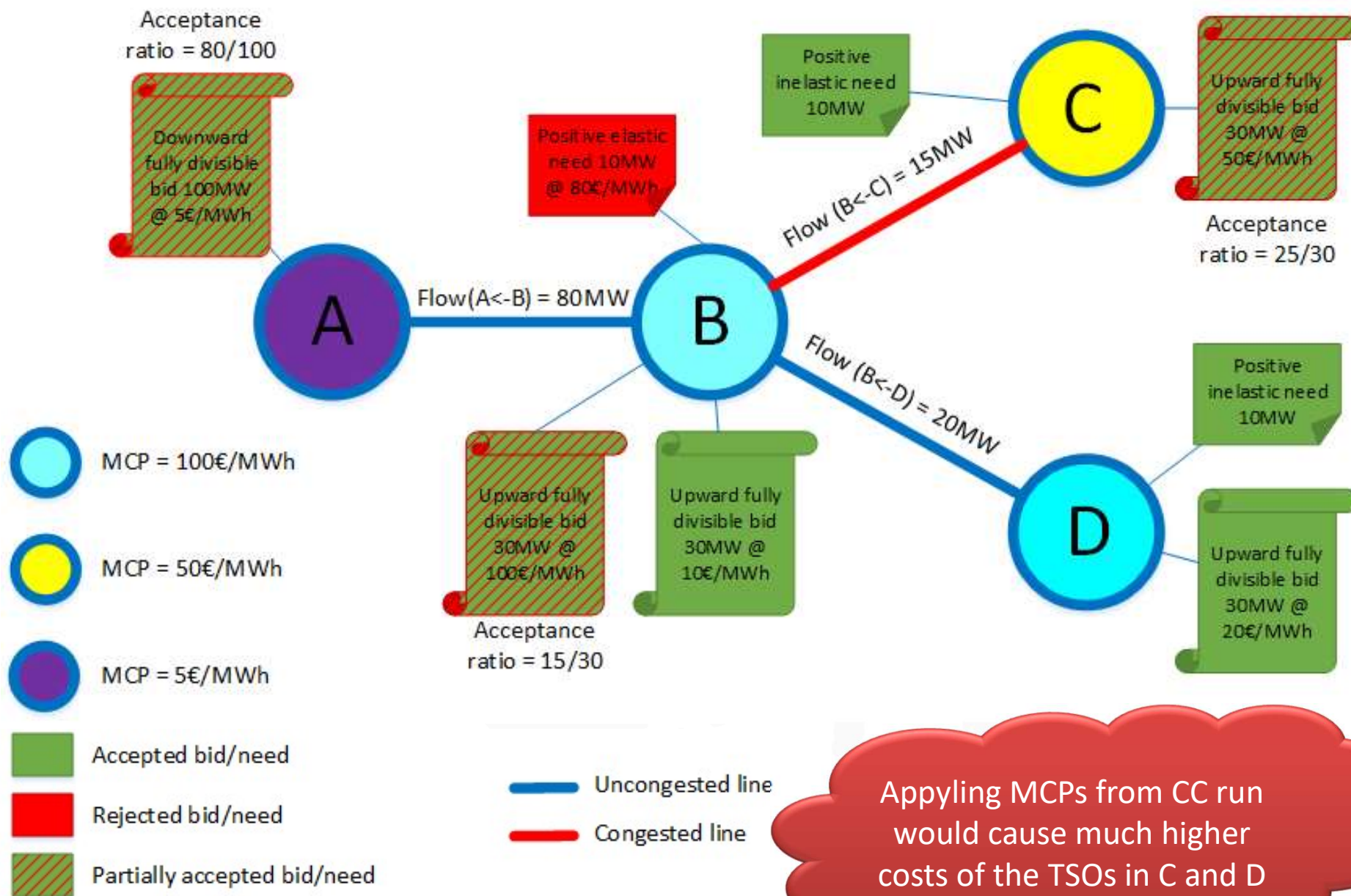
## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability – UC run results



## 5. Algorithm Optimisation Function description-part 2

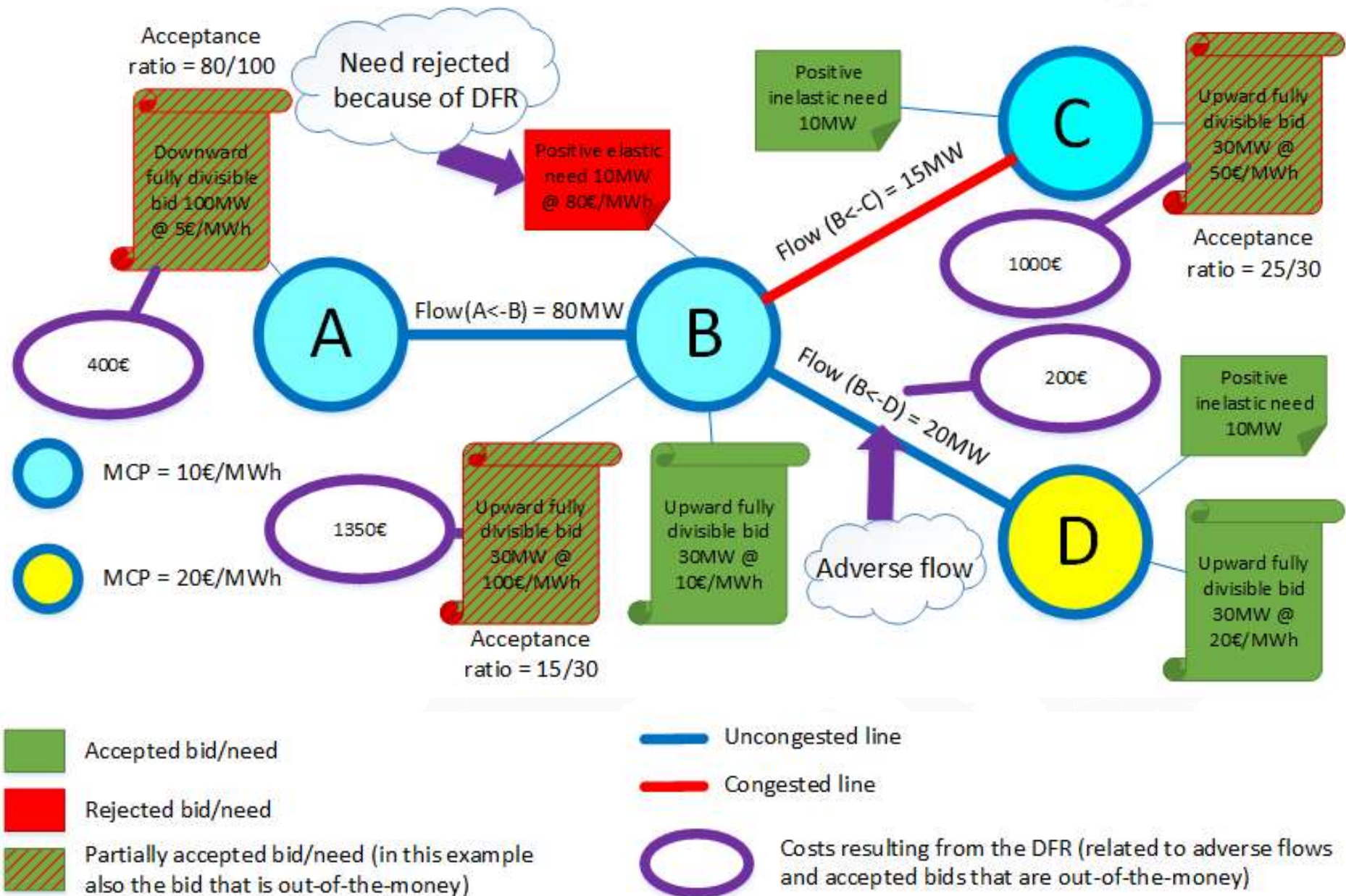
### Optimization Algorithm – Interconnection Controllability – CC run results



Applying MCPs from CC run would cause much higher costs of the TSOs in C and D

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability – Final (binding results)



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

---

#### Existing solution

- The existing implemented solution in LIBRA allows counter-activations, except “naked” counter-activations:
  - When no balancing needs are submitted within a given region, all bids are rejected
- As agreed in the RRIF, no later than twelve months after the go-live of the RR-platform, the AOF will minimise the counter activations which at least may not serve the balancing purpose.

#### Proposal for minimizing counter-activations

---

- Potential prevention of counter-activations:
  - Theoretical formulation
  - Implementation has not yet been done
  - Not possible to conclude if this option can return a feasible solution within the specified time limit

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

---

#### Splitting of needs and bids



#### Objective function

---

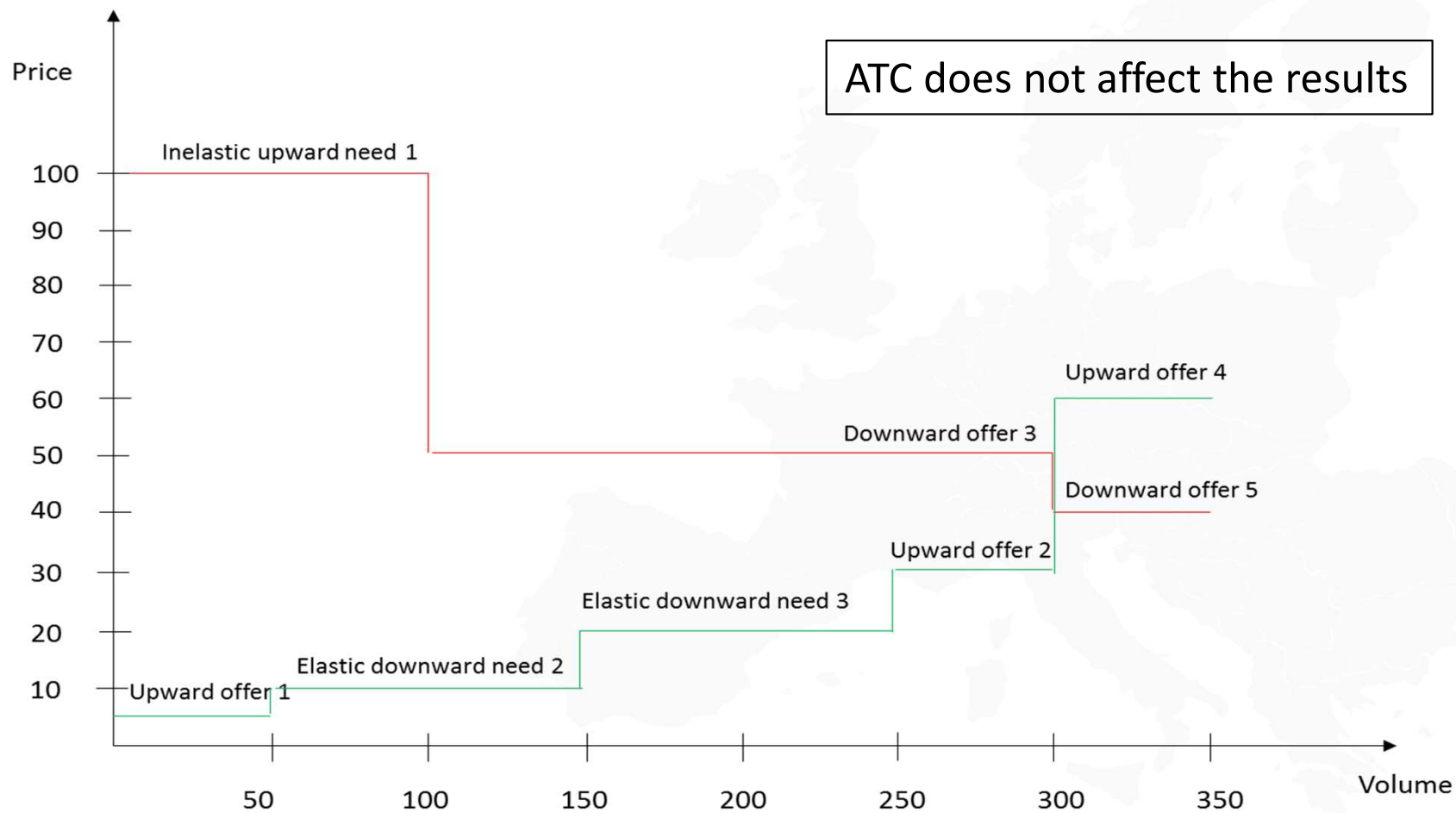
**Maximize:** Need satisfaction + **Need matching surplus** – **Counter activations**

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

#### Example (1/2)

- Inputs:



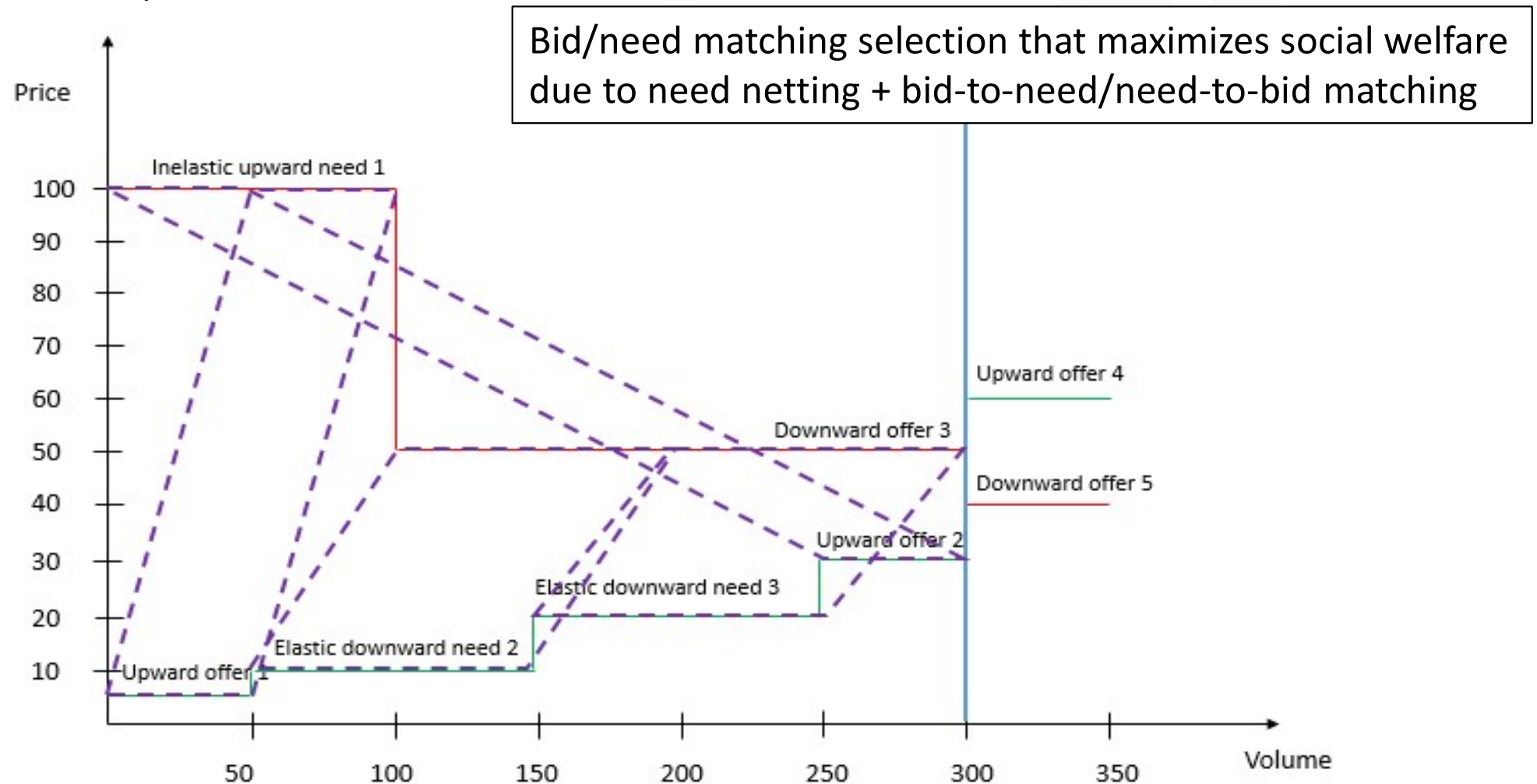


## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

#### Example (2/2)

- Outputs:

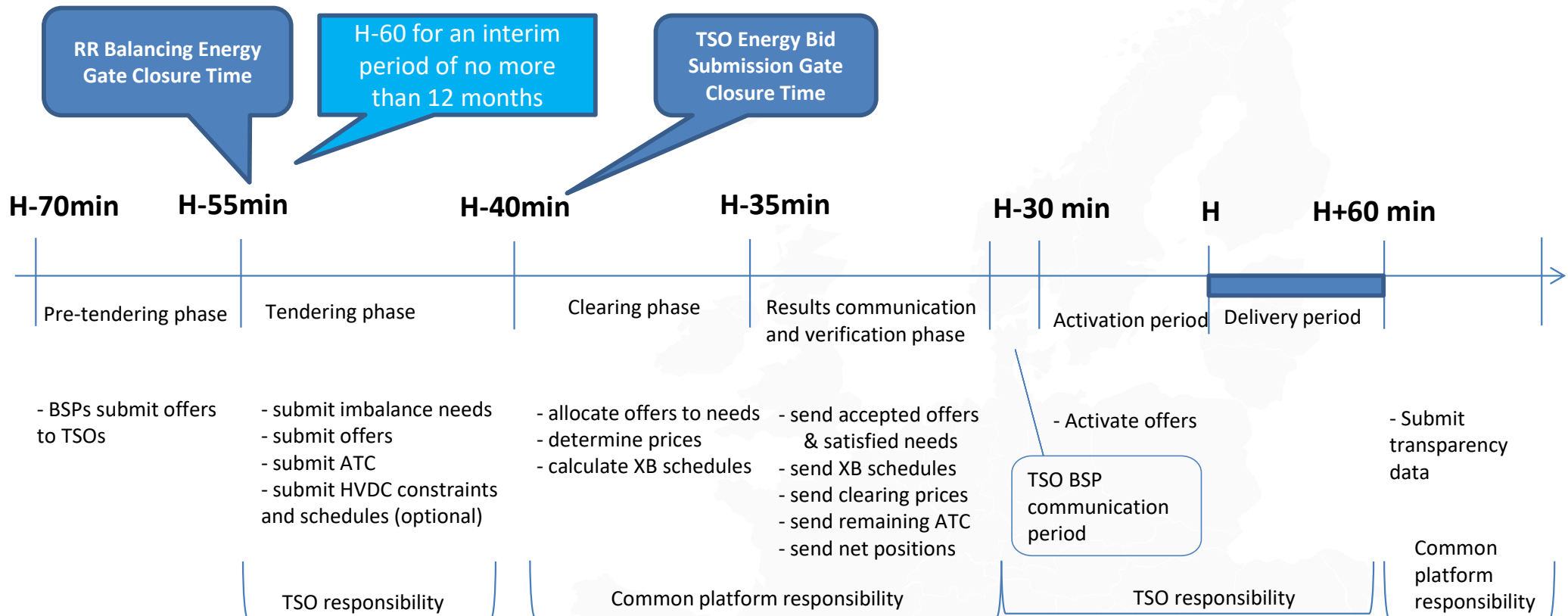




## **Topic 6: RR process timing**

## 6. RR process timing

### Timeline



### RR process consists of the following phases:

- pre-tendering phase
- tendering phase
- clearing phase
- results communication and verification phase
- activation phase
- delivery phase



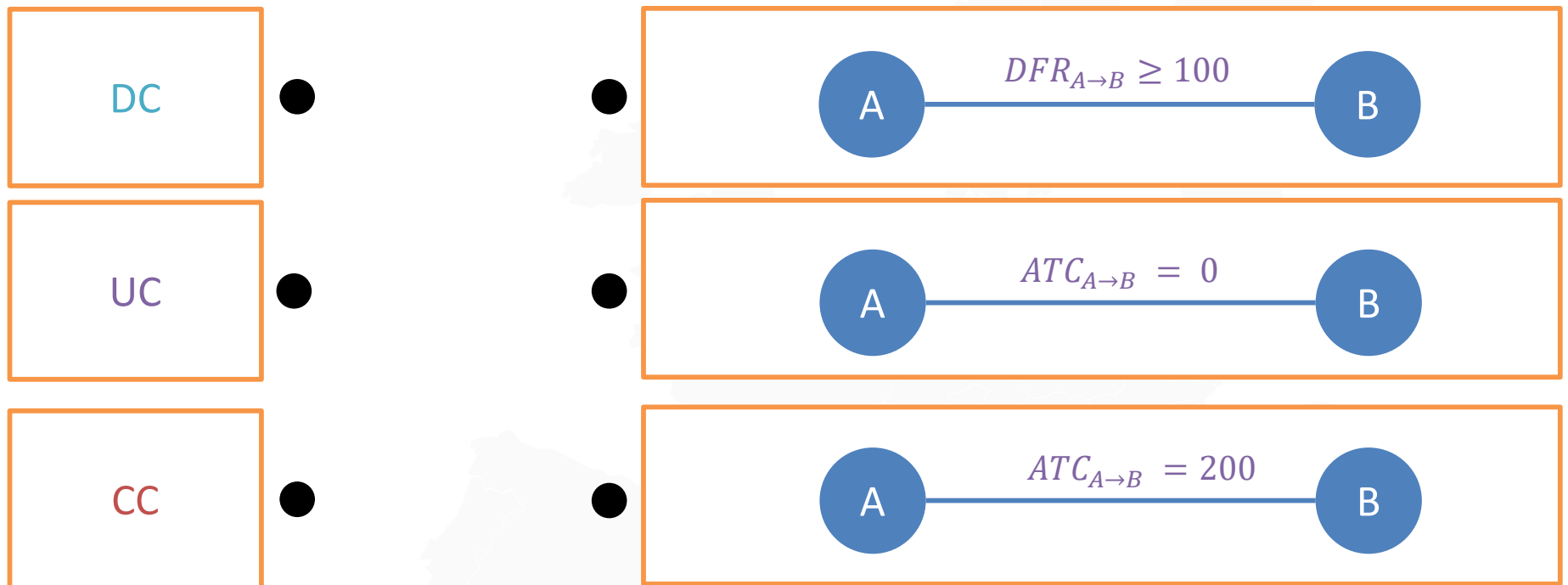
# **Topic 5: Algorithm Optimisation**

## **Function description-part 2**

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

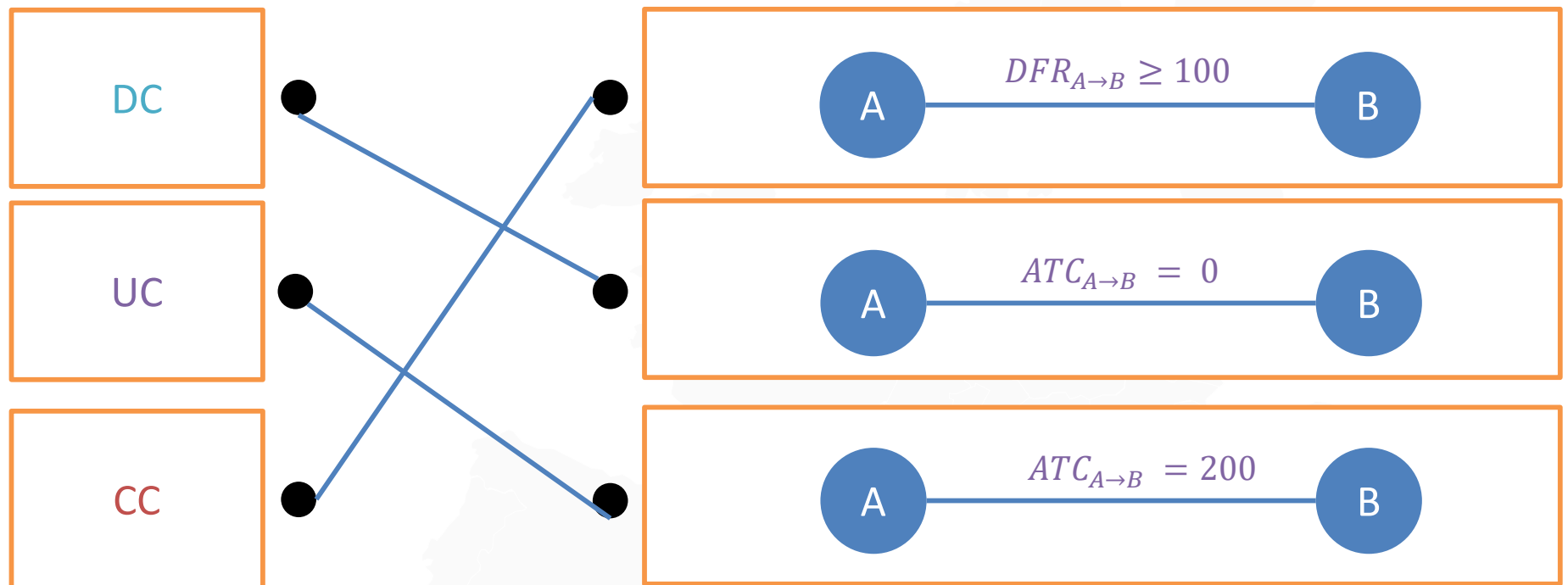
What are the 3 optimization modes?



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

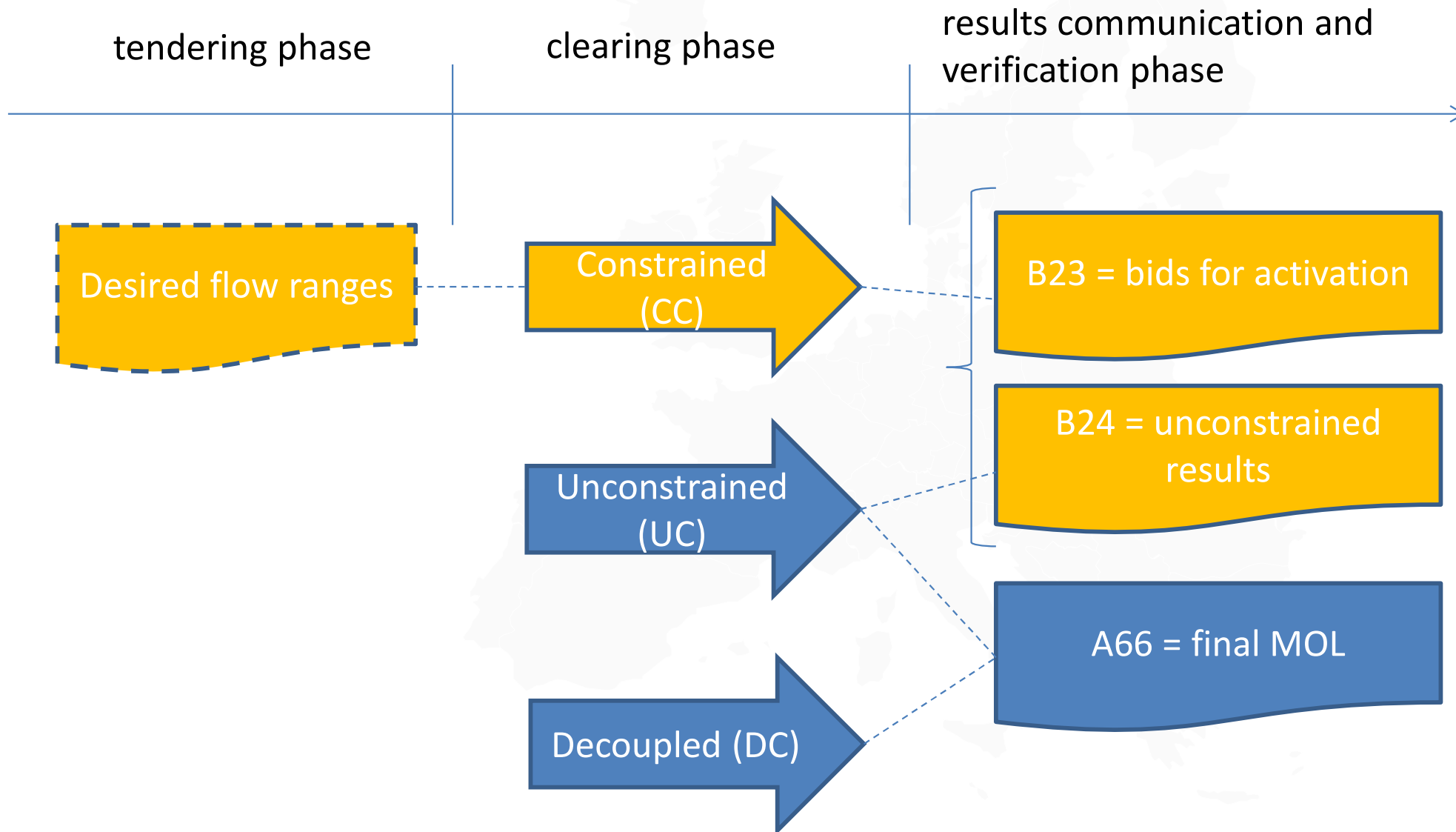
What are the 3 optimization modes?



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Optimization modes

Three runs at the same time: UC, DC, CC

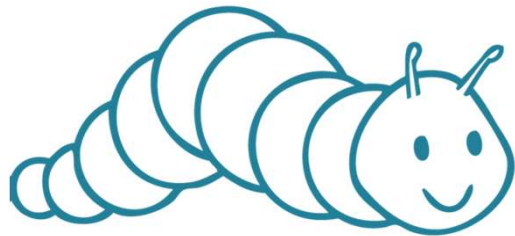


## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Possible solutions

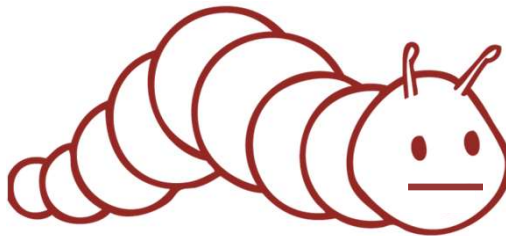
---

Three types of output are possible



#### Optimal solution found

*Meaning feasible solution and no better solution exist in terms of welfare*



#### Not optimal solution found

*Meaning feasible solution exists  
Max calculation time has been reached*



#### No solution found

*Meaning no feasible solution:*

- Timed out
  - Process stopped manually
  - Some part of the platform failed
- > decoupled fallback calculations



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability

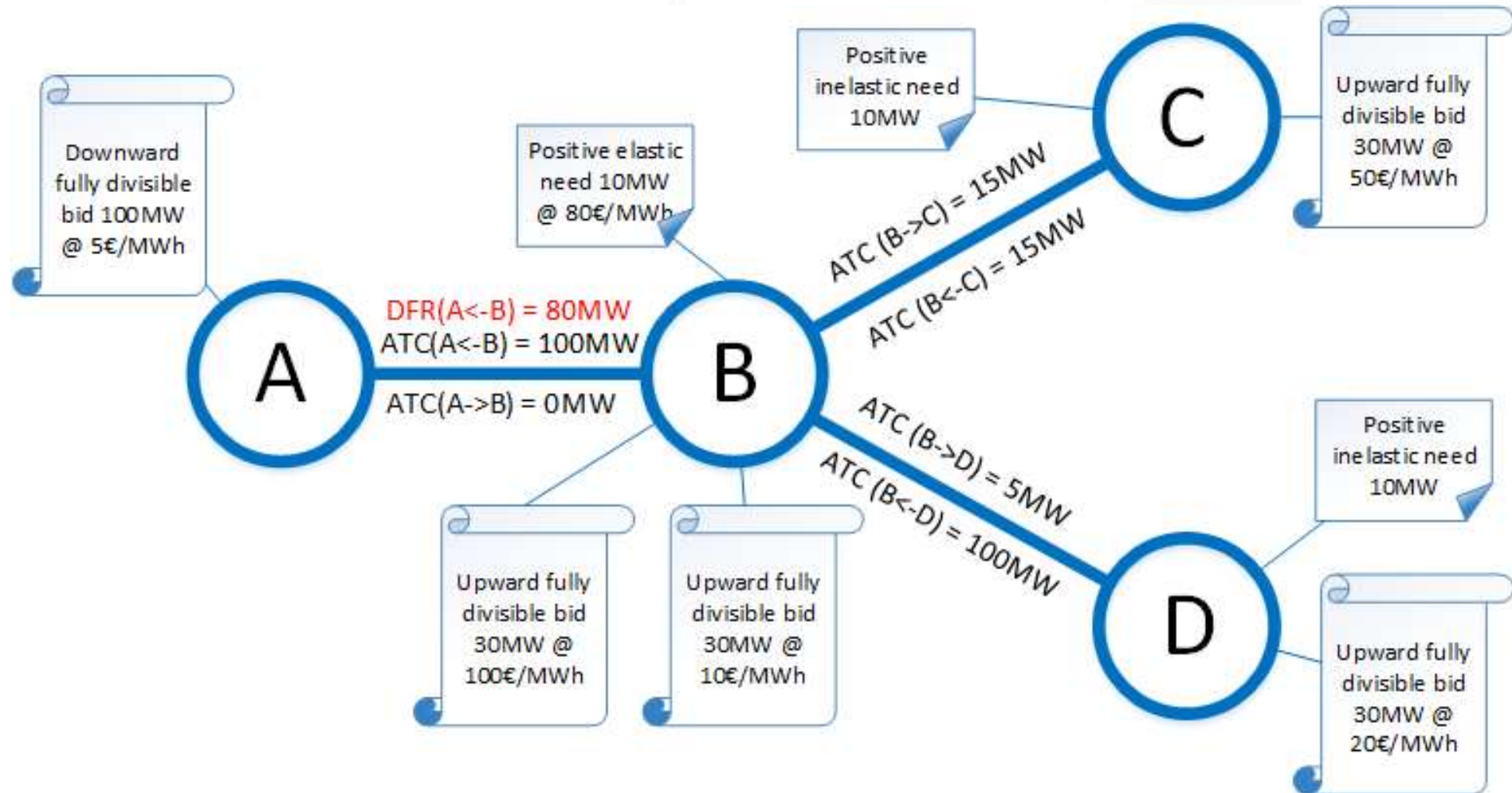
---

- CC run takes into account desired flows at interconnectors:
  - Prevent adverse flow rule and enforce price convergence rule do not apply to the borders on which the desired flow is defined
- UC run does not consider the desired flow ranges
- Both runs determine the accepted volumes of bids/needs and MCPs, but only accepted volumes of bids/needs from CC and the MCPs from the UC run are binding:
  - The bids/needs accepted in CC but not accepted in UC are considered as to be activated because of system constraint purpose and are settled according to pay-as-bid rule
  - The bids/needs rejected in CC but accepted in UC are considered as to be rejected because of the system constraints
  - The interconnectors congested in CC but not congested in UC will not generate congestion rent
  - The interconnectors uncongested in CC but congested in UC could generate (positive or negative) congestion rent

## 5. Algorithm Optimisation Function description-part 2

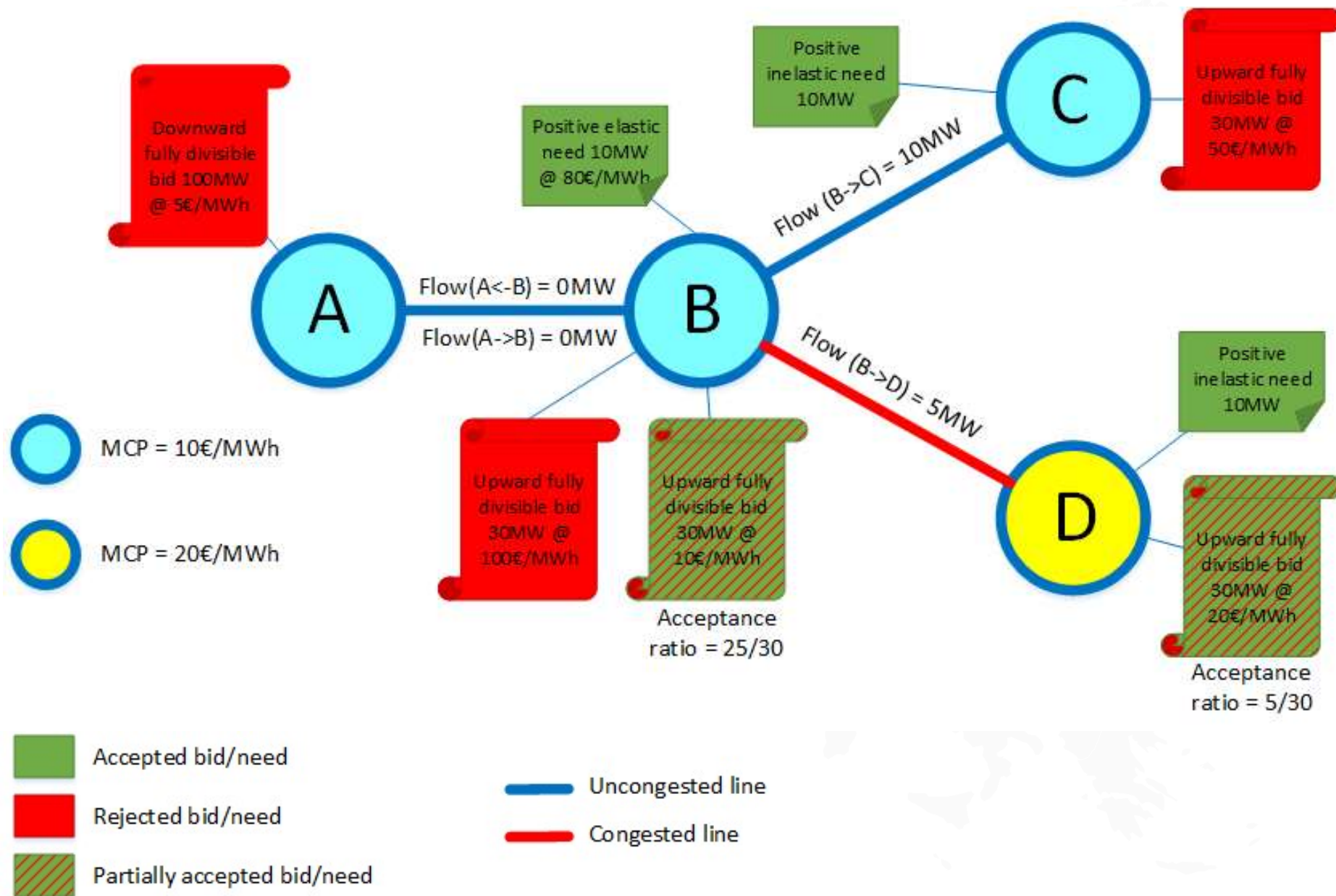
### Optimization Algorithm – Interconnection Controllability

#### Example – input data



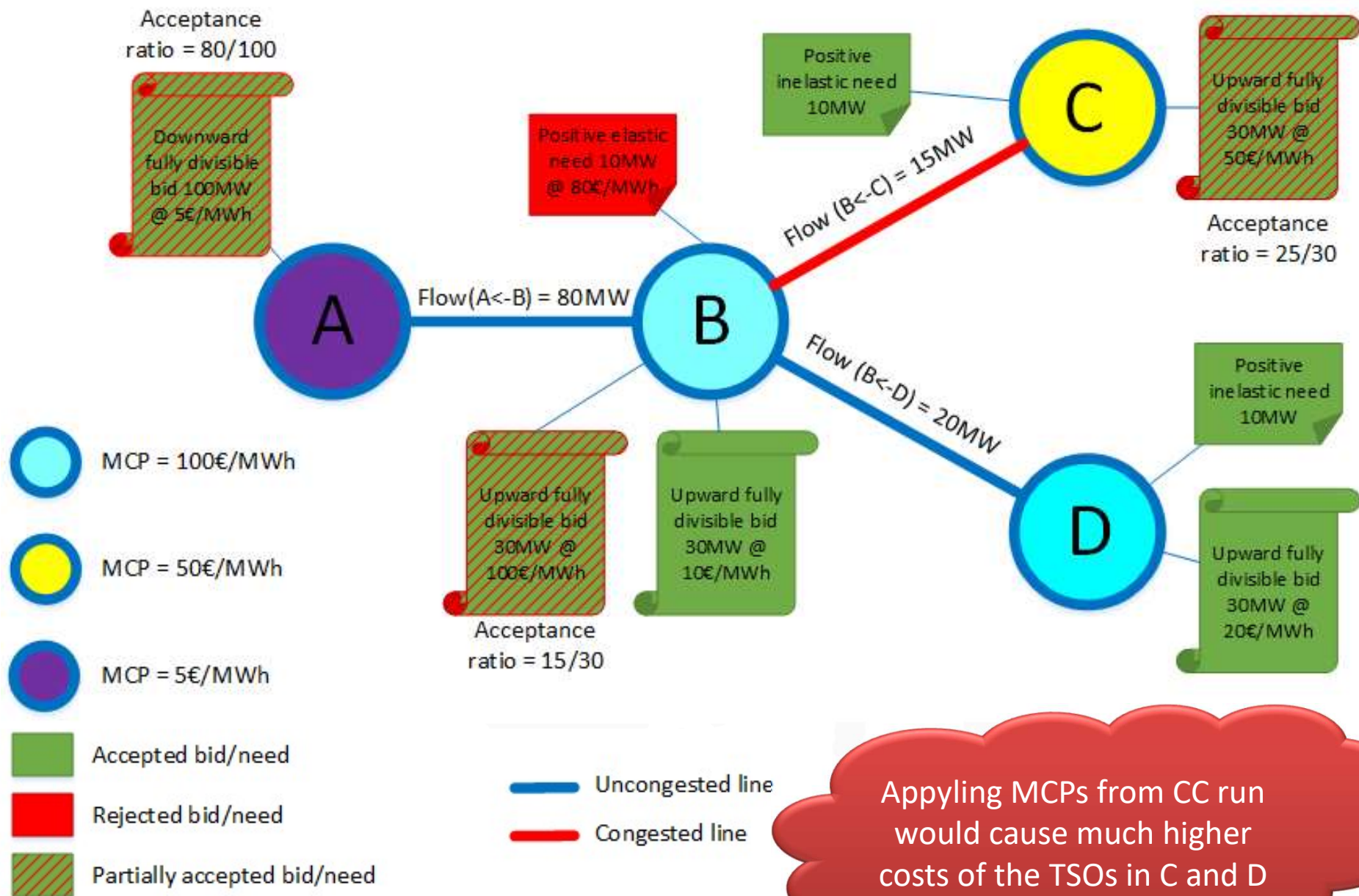
## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability – UC run results



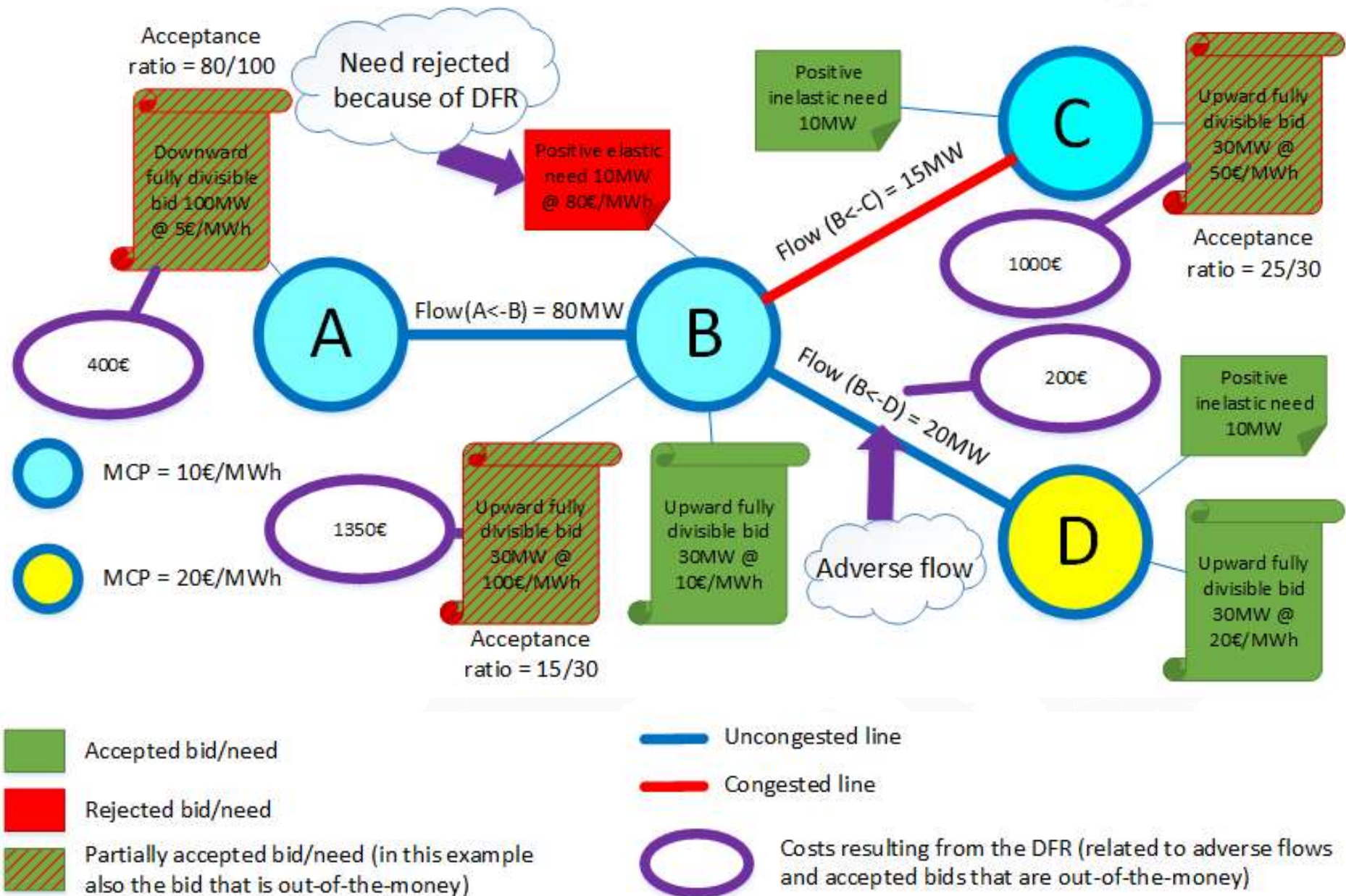
## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability – CC run results



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Interconnection Controllability – Final (binding results)



## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

---

#### Existing solution

- The existing implemented solution in LIBRA allows counter-activations, except “naked” counter-activations:
  - When no balancing needs are submitted within a given region, all bids are rejected
- As agreed in the RRIF, no later than twelve months after the go-live of the RR-platform, the AOF will minimise the counter activations which at least may not serve the balancing purpose.

#### Proposal for minimizing counter-activations

---

- Potential prevention of counter-activations:
  - Theoretical formulation
  - Implementation has not yet been done
  - Not possible to conclude if this option can return a feasible solution within the specified time limit

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

---

#### Splitting of needs and bids



#### Objective function

---

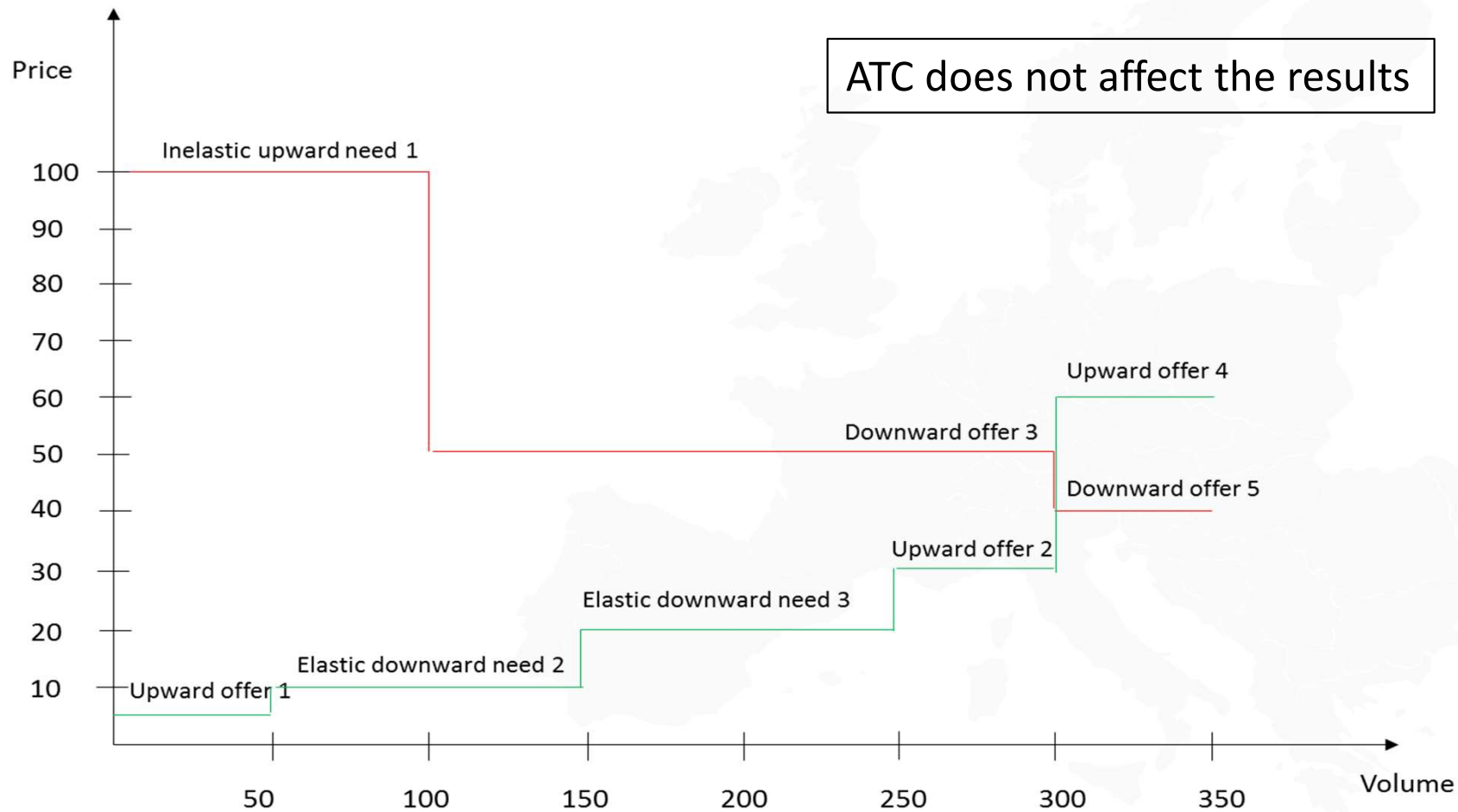
**Maximize:** Need satisfaction + **Need matching surplus** – **Counter activations**

## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

#### Example (1/2)

- Inputs:



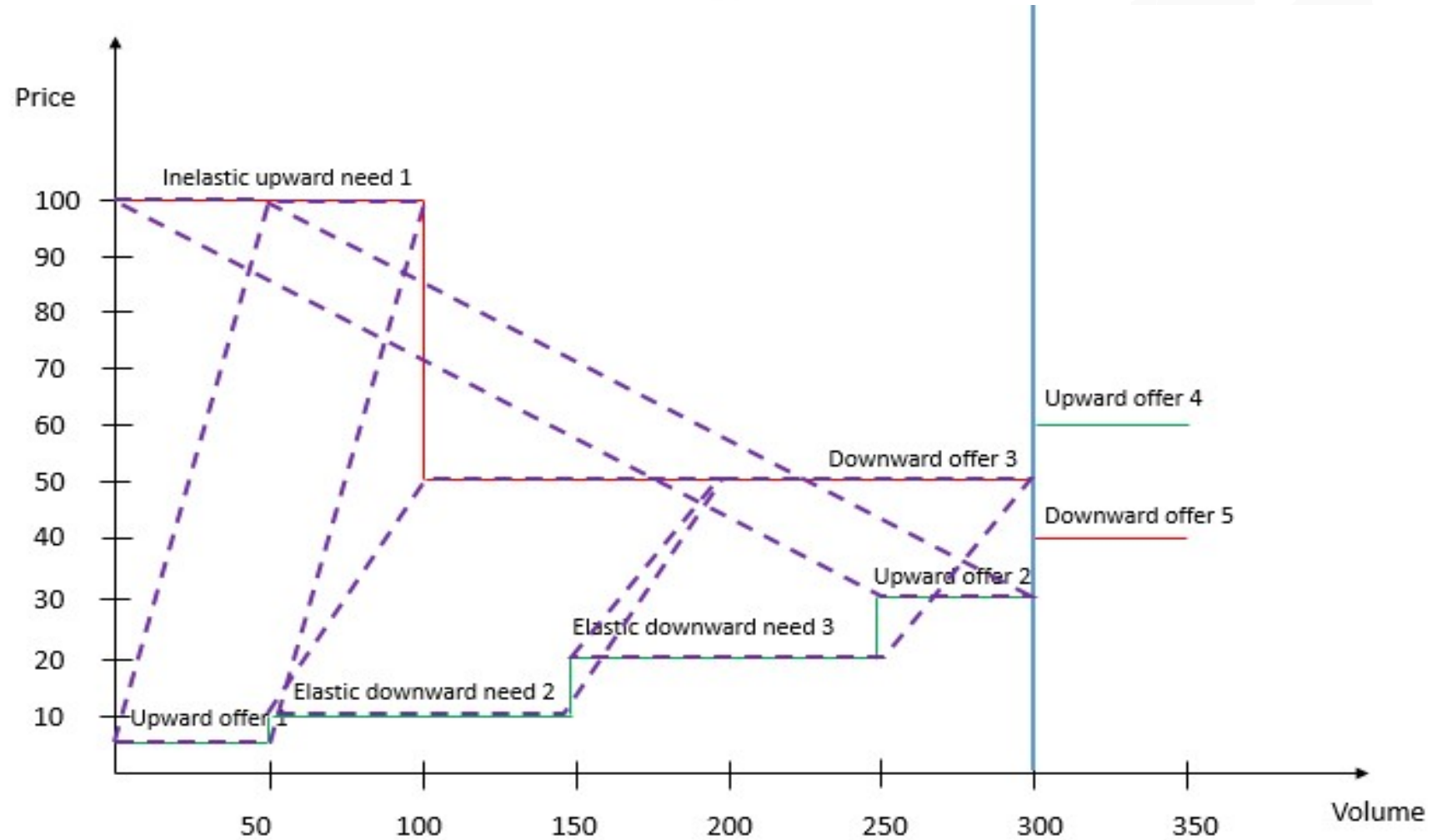


## 5. Algorithm Optimisation Function description-part 2

### Optimization Algorithm – Counter-activations' treatment

#### Example (2/2)

- Outputs:

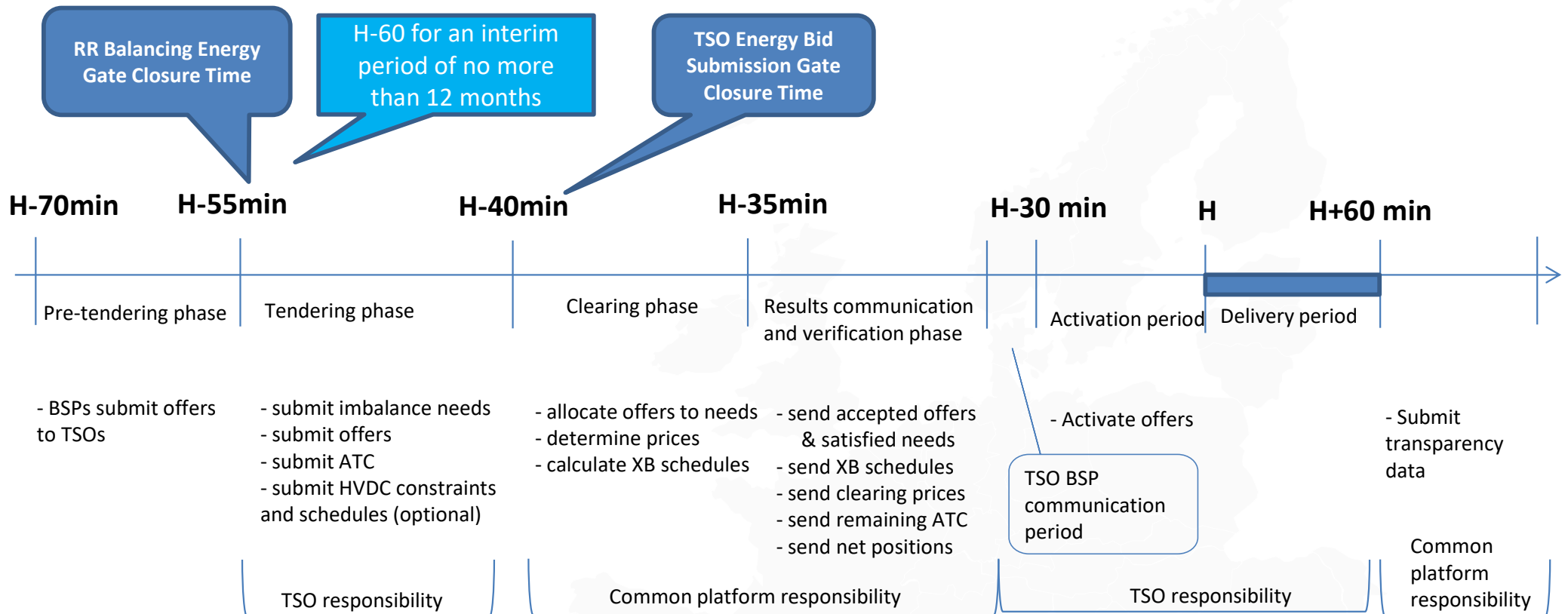




## **Topic 6: RR process timing**

## 6. RR process timing

### Timeline



### RR process consists of the following phases:

- pre-tendering phase
- tendering phase
- clearing phase
- results communication and verification phase
- activation phase
- delivery phase



## **Topic 7: Parallel Run phase, BSP involvement and planning for Go- Live(s)**

## 7. Parallel Run phase, BSP involvement and planning for Go-Live(s)

### Objectives

---

- Verify correct interaction of platform and TSO systems when all connected – prove that configuration is correct and all data & message paths fully functional
- Check on operational procedures and exception handling, including human communication paths and processes, fall-back procedures
- Demonstrate end-to-end process operation
- Gain experience of result of optimisation with data as close to “real-life” as possible
- Confirm readiness for Go-Live

## 7. Parallel Run phase, BSP involvement and planning for Go-Live(s)

### Scope

---

- All platform functions including financial outputs
- West Region only – all TSOs in the region (though not necessarily starting at the same time)
- All bid types
- Use of Desired Flow Ranges
- Area de-coupling



## 7. Parallel Run phase, BSP involvement and planning for Go-Live(s)

### Go live forecasts

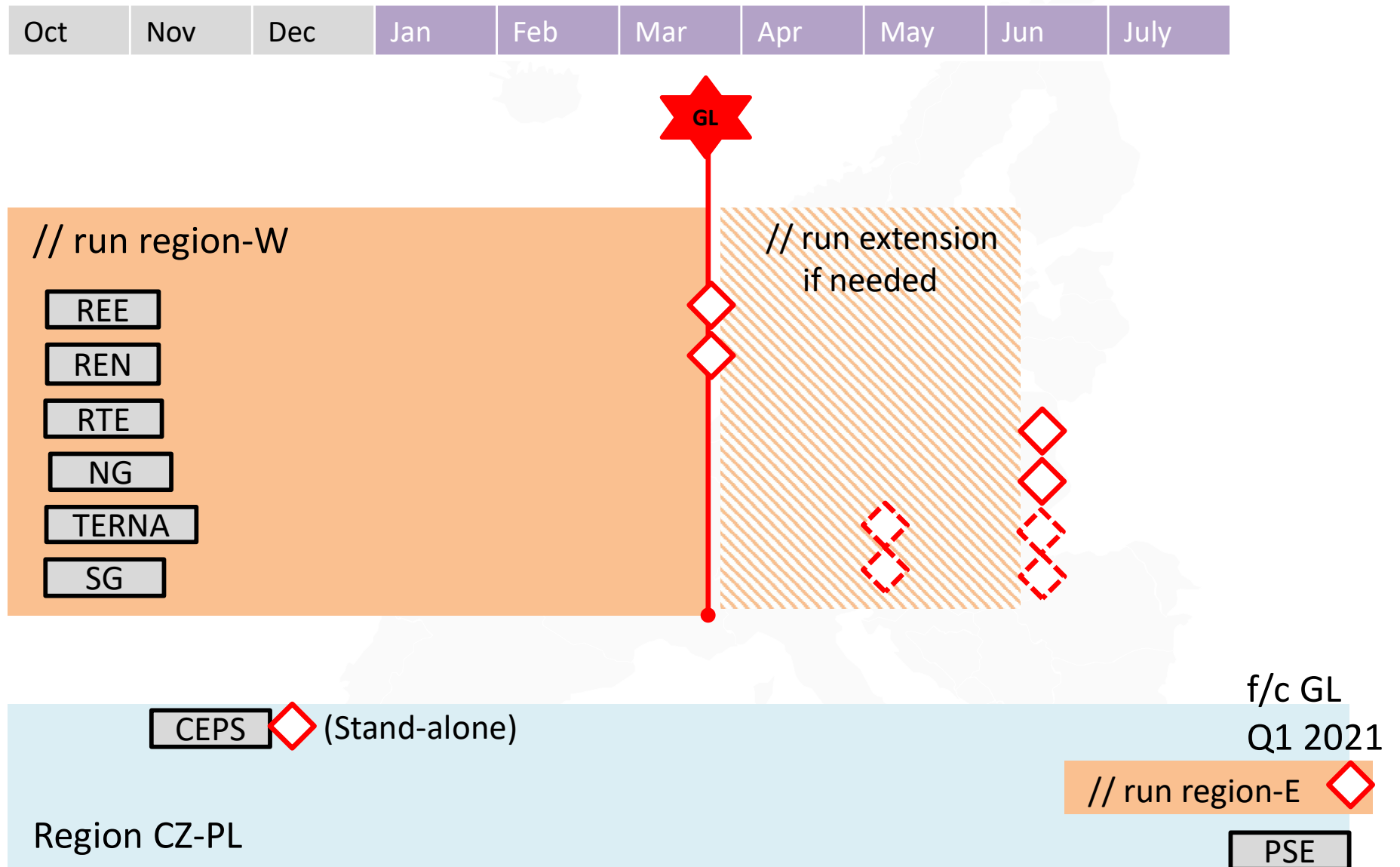
---

TSO	Go live forecast
CEPS	December 2019
REE	Q1 2020
REN	Q1 2020
SG	Q2 2020
TERNA	Q2 2020
NG	June 2020
RTE	June 2020
PSE	Q1 2021

- These new forecasts will require requests for derogation which will be submitted to NRAs for approval

# 7. Parallel Run phase, BSP involvement and planning for Go-Live(s)

## Timeframe to reflect TSO forecast go-live dates







## **Topic 8: Transparency and publication**

## 8. Transparency and publication

### Transparency reporting from Libra

---

#### Legal background

- Libra undertakes transparency reporting in accordance with Transparency Regulation (TR) and Guideline on Electricity Balancing (EB GL).
- Following articles are in scope for Libra:
  - TR art. 17.1.f – Balancing energy prices
  - EB GL art. 12.3.b&c - Balancing energy bids and their conversion
  - EB GL art. 12.3.e(iii & iv) – Aggregated offered, activated and unavailable balancing energy bids

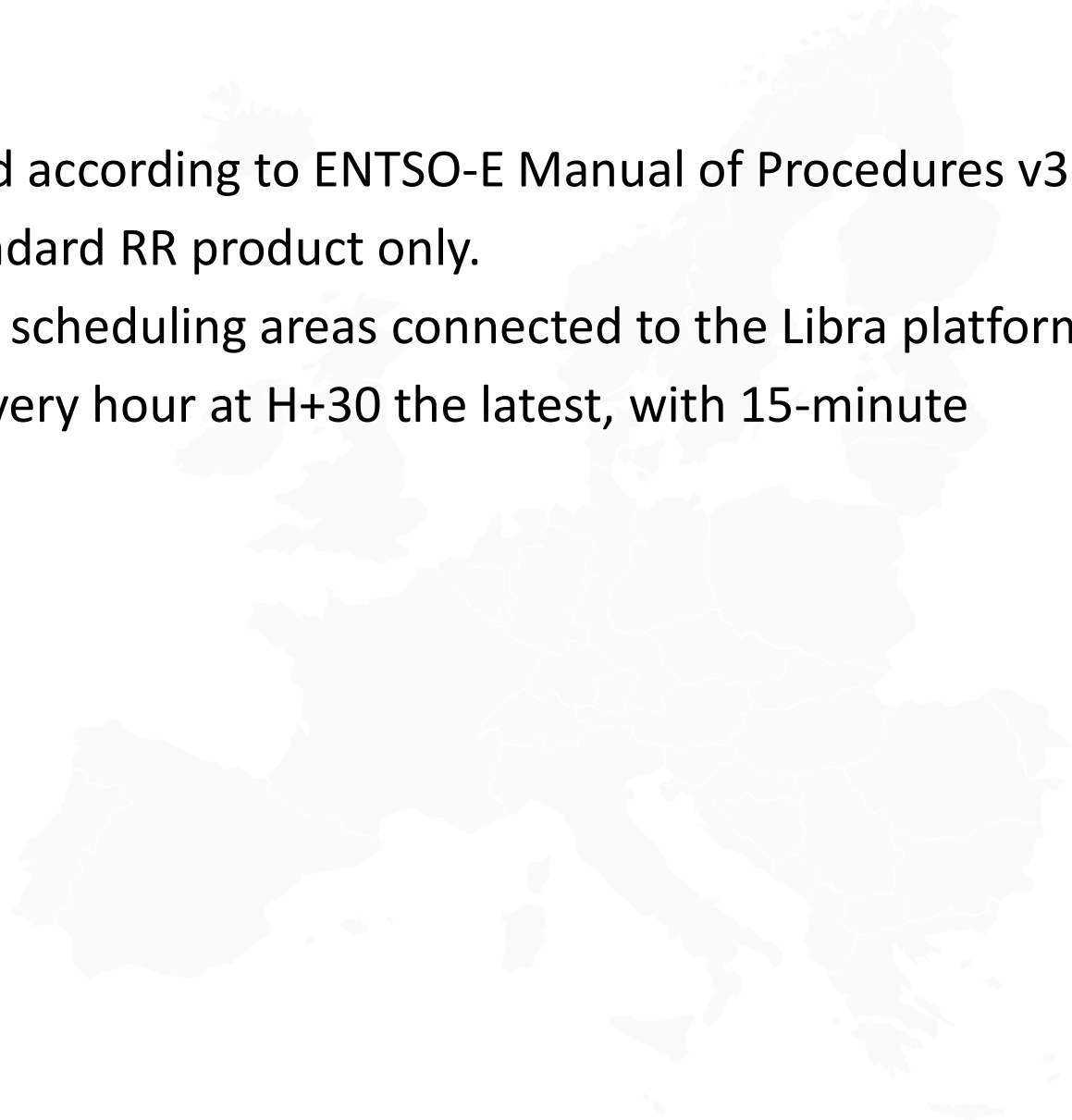
## 8. Transparency and publication

### Transparency reporting from Libra

---

#### Implementation

- Reporting performed according to ENTSO-E Manual of Procedures v3r2.
- Libra reports on standard RR product only.
- Data reported for all scheduling areas connected to the Libra platform.
- Data delivered for every hour at H+30 the latest, with 15-minute resolution.



## Topic 9: Q&A





**Thanks for your participation**