N-SIDE

EAC Reserve and Response Algorithm

Public description

June 2024

Client:	National Energy System Operator (NESO)
N-SIDE Contact Persons:	Marcelo Torres (Principal - Energy Markets) mto@n-side.com, +30 69 51 921 831 Louise Adam (Consultant - Energy Markets) lad@n-side.com, +32 471 08 55 76
Date:	12th June 2024

Contents

Glossary

Acceptance ratio	A number between 0 and 1, indicating the proportion of the volume of a buy or sell order which has been matched by the Auction algorithm.
Accepted order	An accepted order is a (buy or sell) order which is fully or partially matched by the Auction algorithm i.e., an order with an acceptance ratio strictly greater than 0.
Auction	Process of matching sell and buy orders to return a selection of executed orders and market clearing prices.
Ancillary Services	Services procured by the Buyer to balance demand and supply and ensure security of electricity supply across Great Britain's transmission system.
Basket	Functionality to allow the grouping of sell orders with a common parent order. All orders have the same service window and are of the same service type.
Buyer	The entity responsible for buying Ancillary Services. In the context of the EAC, the National Energy System Operator, also referred to in this document as the Market Operator.
Buy order	A buy order expresses the price (£/MW/h) the Buyer is willing to pay to buy a certain quantity (MW) of a particular Ancillary Service Product for a certain service window.
Child order	An order linked to a parent order such that the child order cannot be accepted if its parent order has not been accepted.
Co-optimisation	On the sell side, ability for Market Participants to submit mutually exclusive or substitutable orders within or across service types. On the buy side, ability for the Buyer to submit substitutable orders. The clearing algorithm selects the set of orders that maximise welfare.
(Fully) Curtailable order	Order which can be partially accepted (i.e., with a Minimum Acceptance Ratio < 1) or fully curtailable order (i.e., a Minimum Acceptance Ratio equal to 0).
Enduring Auction Capability (EAC)	System that delivers co-optimised procurement of Ancillary Services. The EAC system consists of an Auction clearing algorithm, an in-house portal (part of the Single Markets Platform) for participants to interact with the Auction and associated internal business processes necessary for running the Auction and integrating data into core Electricity National Control Centre (ENCC) systems.
National Energy System Operator (NESO)	The Operator of the electricity transmission system in Great Britain. Also referred to in the document as the Buyer or the Market Operator
Linked orders	Two orders of which the first one is the parent of the second one (the child). The child order may be accepted (including partially accepted) only if the parent is accepted in the Auction results. A transfer of surplus from the child order to the parent order is possible.

Looped orders	Two orders which must both be accepted, or both be rejected in the Auction results.
Market Clearing Price (MCP)	Price at which a particular product will be procured by the Buyer. There is one market clearing price per product and per service window.
Market Operator	National Energy System Operator (NESO). Also referred to in the document as the Buyer.
Market Participant	A party participating in the Auction as flexibility provider. A Market Participant is linked to one or multiple users and owns one or multiple units.
Non-curtailable order	Order which must be fully accepted or rejected.
Non-substitutable order	Child curtailable order which cannot be substituted with other orders of its basket.
Objective function	Criterion to be optimised (maximised or minimised) by the selection of buy and sell orders of the Auction results.
Overholding	Over-procurement of capacity with respect to the Buyer's willingness to buy given the market clearing price.
Paradoxically accepted order	An order which is (fully or partially) accepted while its total surplus (including transfers of surpluses from/to other orders via links and surplus transferred to its substitutability group) is negative.
Paradoxically rejected order	An order which is (fully or partially) rejected while its total surplus (including the surplus transferred to its substitutability group) would be positive if it was accepted.
Parent order	An order that can be linked to one or more child orders such that each child order cannot be accepted (even partially) if its parent order has not been accepted. Note that, on the sell side, each basket contains strictly one parent order.
Partially accepted order	A partially accepted order is an order with an acceptance ratio strictly above 0 and strictly below 1.
Product	Term used to distinguish between the different types of Ancillary Services procured by the Buyer. A product is defined by a service and a product direction. For example, the service Dynamic Containment, includes two products: Dynamic Containment High and Dynamic Containment Low.
Product direction	Term used to distinguish between the two different directions of a particular product: High or Low for Response Services Negative or Positive for Reserve Services
Rejected order	A rejected order is an order which is fully rejected in the Auction results i.e., an order with an acceptance ratio equal to 0.
Service	Ancillary Service procured by the Buyer (e.g., Dynamic Containment)
Service type	Categorisation of the services in different types.

Sell order	A sell order expresses the price (£/MW/h) a Market Participant is willing to be paid to provide a certain quantity (MW) for one or more products, for a certain service window.
Seller	A Market Participant.
Service window	Pre-defined delivery period of a service.
Social welfare	Sum of surpluses of accepted sell and buy orders.
Splitting	Ability for a unit to split its capacity across different services on the same service window.
Substitutable order	On the sell side, child curtailable order which can be substituted with other substitutable child orders of its basket. On the buy side, fully curtailable buy order which can be substituted with other substitutable buy orders of the same substitutability family.
Surplus	For a sell order, the surplus is the difference between the payment received from the Auction (including transfers of surplus paid to/received from other orders via links and surplus transferred to its substitutability group) and the sell order bid price. For a buy order, the surplus is the difference between the buy order bid price and the payment made to the Auction (including surplus transferred to its substitutability group).
Unit	A collection of one or several assets entitled to offer capacity.
User	A user of the platform. It can be a user from a Market Participant as well as from the Buyer.

1. Introduction

The Enduring Auction Capability (EAC) addresses National Energy System Operator's (NESO) RIIO-2 commitment to deliver co-optimised auctions across Ancillary Services. Co-optimisation achieves a more efficient allocation of resources, thus unlocking cost savings.

The overall EAC system consists of an Auction clearing algorithm, an in-house portal (part of the Single Markets Platform) for participants to interact with the Auction, and associated internal business processes necessary for running the Auction and integrating data into core Electricity National Control Centre (ENCC) systems. The solution is intended to be flexible, scalable, and configurable to future services.

In September 2022, following a competitive procurement process, N-SIDE was selected to support the ESO in the development of the EAC market design and deliver the Auction clearing algorithm that implements the co-optimised procurement of Response & Reserve, alongside a set of enhanced functionalities and improved features, based on market participants' feedback.

This document provides a description of the N-SIDE Power Matching Algorithm, used to clear the EAC auctions for Reserve and Response. The purpose of this supporting document is to provide clarity and transparency on technical aspects of the algorithm calculations, including the objective function of the algorithm, co-optimisation, the order format and associated functionalities, and the approach to determining clearing prices[1].

[1] While N-SIDE considers that the information and opinions given in this work are sound, all parties must rely upon their own skill and judgement when making use of it. N-SIDE does not make any representation or warranty, expressed or implied, as to the accuracy or completeness of the information contained in this report and assumes no responsibility for the accuracy or completeness of such information. N-SIDE will not assume any liability to anyone for any loss or damage arising out of the provision of this report.

2. Overview

The table below provides an overview of the main design elements of the EAC.

Auction Design & Pricing	• Locational granularity: GB synchronous area. • Auction type: Closed double-sided auction. • Objective function: Maximisation of social welfare. • Pricing: Uniform clearing price per product for each service window. • Overholding allowed (cleared quantity may exceed ESO bid quantity). • Minimum and maximum price limits are set by the Market Operator.
Co-optimisation	• Participants can place orders for multiple products in the Auction. The Auction clearing algorithm will accept the order(s) which clears the market most efficiently (i.e., maximises social welfare). • A portion of the Buyer's requirements for a particular product can be substituted by provision of a different product (e.g., to allow excess offers for product X satisfy ESO requirements for product Y) if this should be a more efficient market clearing solution. The Buyer controls the quantity of each product that can be substituted.
Buy Order Features	• All buy orders are fully curtailable. • The Buyer can create substitutability families containing several buy orders to express that it is indifferent between accepting different sets of Ancillary Services. • Overholding is modelled by allowing buy orders to be paradoxically accepted. It means that the Buyer can procure additional capacity if it generates higher welfare.
Sell Order Features	• Via the use of baskets (the maximum number of baskets a unit can submit in each Auction is configured by the market operator), participants can submit several mutually exclusive orders (i.e., orders that cannot be accepted together). • All orders can assign volumes in a single or multiple products. • Each basket must contain one parent order (non-curtailable). • Child orders: Orders linked via a parent-child link are available so that child orders can be accepted only if the parent order is accepted. Child orders are fully curtailable. The number of child orders that a basket may contain is configured by the market operator. • Substitutable child orders: For each basket of orders, a family of substitutable child orders can be submitted so

	that the Seller can express its indifference between the acceptance of any combination of this set of orders. The number of substitutable child orders that a basket may contain is configured by the market operator. • Market Participants can loop baskets of non-overlapping service windows so that all of them are accepted or rejected.

3. Objective function

The optimisation problem used to clear the Auction seeks to maximise an objective function, which is chosen to be the welfare function in the EAC. More precisely, the welfare function is defined as the total costs of offers subtracted from the total utility of demand (see Figure 1). By maximising welfare, the Auction algorithm is able to determine the optimal acceptance ratios of all buy and sell orders submitted to the Auction such that the sum of the economic contributions of all orders (across all products and service windows) is maximal.

Indeed, using this particular objective function allows a matching of orders such that the total surplus generated by the Auction is maximised. The surplus of a buy order corresponds to the difference between its willingness to pay for capacity and the actual market clearing price, the total multiplied by the accepted capacity. With respect to the sell side, the surplus or profit of a sell order is defined as the revenue obtained by this order (accepted quantity times market clearing price) to which the cost incurred by the provision of this order is removed.

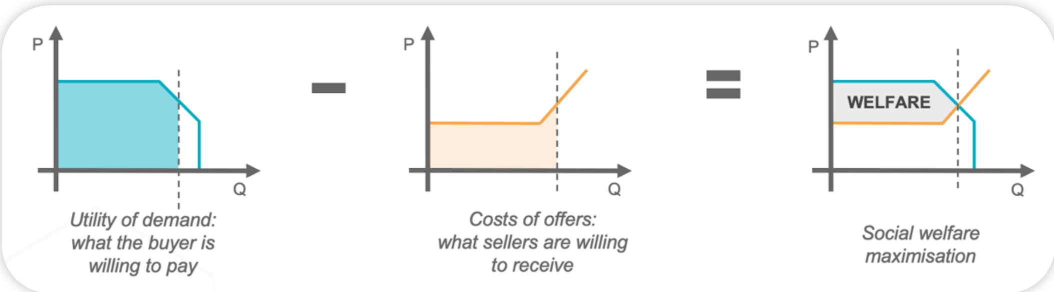

Figure 1 - Welfare maximisation principle.

The advantage of welfare maximisation is first a guarantee of the optimal order selection: offers with lowest costs are selected first and demand utility is maximised. Further, it will also decrease procurement costs in the long-term because of incentives to bid at true (actual or opportunity) costs and incentives to invest in the services where infra-marginal rents are the highest. Note that procurement cost is defined as the sum of the accepted sell volumes times their corresponding market clearing price. This notion is also accounted for while computing market clearing prices for the Auction after fixing the acceptance ratio of orders via welfare maximisation. This process is explained in Section 7.2.

The welfare maximisation process will select the solution with the highest objective function value that the algorithm has identified within a given time limit. In addition, there might exist several equivalent welfare-maximising solutions with different total accepted volumes or different selection of orders (for example, if only one of two exactly equivalent orders is matched) may exist. If several solutions produce the same welfare, the first solution found by the algorithm will prevail.

4. Co-optimisation

The capability to facilitate co-optimisation across different Ancillary Service markets means that the Auction clearing algorithm will allocate a participant unit to the Ancillary Service(s) where it most efficiently clears the market. In contrast, in a non-co-optimised market design (as has been the case prior to EAC), participants must select the single market and service in which each unit will compete in advance of market clearing. In the absence of co-optimisation, participants risk seeing their offer rejected for a given service while their unit could have efficiently delivered another service, for instance a service for which the ESO demand could not be fully matched or could only be fulfilled at a high price. Co-optimisation therefore benefits both the ESO (who can satisfy requirements for Ancillary Services at lower cost) and providers (who reduce the risk that a competitive offer of services will be rejected in market clearing).

Co-optimisation in the EAC has therefore been made available both to the Sellers and the Buyer.

Regarding the sell side, providers can choose to offer more than one Ancillary Service by a single unit for a given service window. This is done by submitting multiple possible offers. Each offer can contain a single service or a combination of services in case providers choose to split their capacity into multiple services/products (see Section 6.1).

For example, let us consider three Market Participants submitting offers for Product 1 and Product 2 (Figure 2). Market Participant 1 has 200MW of available capacity and decides to offer them on both products. The EAC algorithm will select the optimal offer through co-optimisation. In the example of Figure 2, Market Participant 1 will get their Product 2 offer selected.

Figure 2 - Co-optimisation example: Market Participant 1 offers capacity for 2 products and the algorithm chooses for which product it is selected so that the welfare is maximised. The capacity offered by Market Participant 1 on Product 2 is selected by the algorithm as it produces a higher welfare.

In practice, with respect to sell orders, Market Participants can submit multiple offers for the same service window for which only one can be accepted by using the concept of baskets presented in Section 6.1. Sellers also have the possibility to express their indifference for the algorithm to choose any combination of acceptance ratios of a particular set of orders. These orders will be flagged as substitutable child orders inside the same basket by a Market Participant (see Section 6.2).

The Buyer can also benefit from the co-optimisation feature. Indeed, the Buyer can flag some buy orders as belonging to the same substitutability family (see Section 5.2) to represent the possibility to swap between two different products to procure the desired Ancillary Services capacity.

With this co-optimisation approach, Market Participants no longer need to estimate which service is the most valuable, they can make multiple offers and be selected only for the optimal one. This market design aims to reduce both unused capacity in some services and deficits in the procurement of other services.

5. Buy orders

This report defines two types of orders for the procurement of Ancillary Services: buy and sell orders. This section introduces the design and the different features of the buy orders. These orders will be employed by the Buyer to express its demand for Ancillary services.

5.1. Formation of buy orders

A buy order specifies a quantity (MW) of a particular product that the Buyer is willing to buy paying up to a certain price (£/MW/h) during a certain time period. In practice, the product will be chosen from the set of available products auctioned through the EAC presented in Section 2 while the time period will correspond to a feasible service window with respect to this chosen product. All buy orders will be considered as fully curtailable meaning that any portion of a buy order can be accepted by the algorithm.

The Buyer has also the possibility to define two other parameters: the substitutability of the order and the permission for its order to be paradoxically accepted.

The former, introduced in Section 5.2, grants the opportunity for the Buyer to define families of orders from which the most suited orders or set of orders are chosen by the algorithm. This concept embodies the co-optimisation principle on the Buyer side by allowing the Buyer to submit sets of equivalent Ancillary Services requirements, the best choice among the different options within these sets of equivalent requirements being made by the EAC algorithm.

With the latter (i.e., the permission of a buy order to be paradoxically accepted), the Buyer has the possibility to allow its buy order to be accepted (partially or entirely) even though the market clearing price is above what the Buyer is willing to pay for this particular order. It should be noted that each buy order may be allowed or denied by the Buyer to be paradoxically accepted. Allowing paradoxical acceptance of buy orders enables overholding on the Buyer side as described in Section 5.3.

For the avoidance of doubt, no other features (e.g., baskets, parent-child links, looped orders and any other “all or nothing” constraints requiring binaries) will be made available to the Buyer.

Summarising the previous discussion, the following elements reproduce a complete list of the attributes required in order to construct a buy order:

- **ID**. A unique order ID.
- **Product**: One of the authorised products:
- **Service window**. The service window for which the specified capacity must be procured. Note that the service window must be compatible with the chosen product characteristics.
- **Volume**. An amount of capacity (MW) to be procured in the service window.
    - The volume must be provided as an integer in MW.

    - The volume must be between 0 (included) and a parametrised maximum allowed volume.
- **Price**. A price (£/MW/h) indicative of the willingness to pay to procure the capacity.
    - The price must be provided in pounds and pence (multiple of 0.01 £/MW/h).
    - The price must be between the minimum and maximum market prices.
- **Substitutability family ID** *[default: none]*. The identifier of the substitutability family to which the order belongs, if any (see Section 5.2). Note that an order belongs at most to one substitutability family.
- **Paradoxical acceptance allowed** *[default value: True]*. Buy orders with Paradoxical acceptance allowed set to True can be paradoxically accepted, meaning that buy orders can be accepted even though the market clearing price is above the limit price of the buy order.

5.2. How can buy orders be employed by the Buyer?

Following on from the definition of buy orders presented in Section 5.1, this section will focus on the usage of buy orders by the Buyer in order to represent its willingness to procure Ancillary Services.

Firstly, the Buyer needs to use different buy orders for different products. It has the possibility to submit several buy orders for the same product, each with different quantity and price (willingness to pay). This particularity allows the Buyer to represent that its willingness to pay for buying a particular product may be different for different quantities.

Moreover, as mentioned above, the Buyer has the possibility to flag buy orders as substitutable by assigning them to the same substitutability family. This allows the Buyer to express that any combination of acceptance ratios between the different buy orders of the same family can be chosen by the algorithm, as long as the sum of the acceptance ratios of all orders in the same family does not exceed 100%. For example, accepting one percent of the first order can be swapped with accepting one percent of the second order if the two orders belong to the same family.

The Buyer can therewith express that it is indifferent between accepting different sets of Ancillary Services. The Auction algorithm will thus choose the combination of acceptance ratios inside a substitutability family that maximises social welfare and therefore the combined interests of the Buyer and Sellers.

Finally, in addition to the rules concerning a buy order's input data provided in Section 5.1, the following restrictions apply at the level of the substitutability families:

1. A buy order can belong to at most one substitutability family.
2. A substitutability family can contain at most a certain number of orders, as configured by the market operator.

3. All orders belonging to the same substitutability family must be defined on the same service window or on concomitant[2] service windows.
4. All orders belonging to the same substitutability family must have the same direction (i.e., either all orders have direction high/negative, or all orders have direction low/positive).
5. Orders belonging to the same substitutability family must be offered for different products.

5.3. How are buy orders treated by the EAC algorithm?

Overholding on the Buyer side is allowed by the EAC algorithm. This feature enables the inclusion of more non-curtailable sell orders, when the procurement of some additional volume (compared to Buyer's actual demand) leads to a higher welfare.

Overholding of a particular product and service window is modelled by allowing one or more buy orders to be paradoxically accepted.

1. **Allowing at most one paradoxically accepted buy order:** Overholding can be modelled in its more restrictive version by introducing exactly one so-called *overholding buy order* per service window per product. An *overholding buy order* is an order that is allowed to be paradoxically accepted, with a price of 0 £/MW/h (such that it doesn't influence the objective function), and a volume corresponding to the maximum volume that the Buyer agrees to over-procure during the service window. The over-procured volume is thus limited by the overholding order which de facto also bounds the maximum cost of overholding. This approach is equivalent to enforcing that the matched buy volume can be larger than the matched sell volume by a quantity between 0 and a maximum overholding volume.

2. **Allowing more than one paradoxically accepted buy orders:** Overholding can be modelled in its more permissive version by allowing additional buy orders to be paradoxically accepted. This results in a more relaxed optimisation problem and can produce solutions of higher welfare for exactly equivalent data sets.

In the EAC algorithm, the possibility of a buy order to be paradoxically accepted is governed by a parameter that is set by the Buyer (and which defaults to TRUE). The paradoxical acceptance of an order by the algorithm is given by the fact that this order is accepted even though the market clearing price is higher than the Buyer's willingness to pay for that particular order.

[2] Concomitant service windows are two or more service windows containing time periods in common.

6. Sell orders

This chapter details the format of the sell orders and their associated features. First, we outline the design of sell orders in Section 6.1 and the associated clearing rules for sell orders in Section 6.2. Sellers will use these features to participate in the market and express their preferences, and we therefore illustrate the application of these concepts with some practical use cases in Section 6.3. Finally, in Section 6.4, we present the detailed specification of sell orders and elaborate some of the rules and validations that apply to the formation of sell orders.

6.1. Formation of sell orders

The assets of Market Participants are represented by different units. For each of their units, Market Participants communicate their offers of Ancillary Service capacity into the Auction by means of sell orders. Each sell order consists of an offered quantity (in MW) for one or more products, together with an offer price.

Sell orders are grouped together into baskets. A basket is defined for a service type and on a single unit. Each basket has a service type (e.g., Frequency Response). . Each sell order must be associated with exactly one basket, to which it belongs. The sell order specifies a set of product/quantity pairs, representing the offer of capacity for the various products corresponding to the service type of the basket. A sell order has a single price, in £/MW/h applicable to the entire order (and not a price per product).

A basket is defined on a single service window. The baskets regulate which of the sell orders may or may not be accepted simultaneously. In particular, baskets are used to model mutual exclusivity between sets of orders. Any two baskets are mutually exclusive if they are “concomitant”, i.e., if they are defined on the same service window, or on service windows that overlap in any time period.

Although a basket is defined only on a single service window, it may be “looped” to another non-concomitant basket defined on the same unit. This feature allows participants to build up offers that are defined over longer periods of time. Two or more baskets that are looped together are a “looped family”.

Three different types of sell orders, each with their own clearing rules, are admitted to the Auction: parent orders, child orders, and substitutable child orders. The significance of these order types is explained in the next section. A basket must contain exactly one parent order. The number of child order and substitutable child orders that a basket may contain is configured by the market operator.

6.2. Clearing rules for sell orders

The Auction clearing algorithm determines the acceptance ratio for each order in the Auction, such that the overall Auction welfare is maximised (as explained in Section 3). We say a sell order is “accepted” if its acceptance ratio is greater than 0. The order is “partially accepted” if the acceptance ratio is between 0 and 1, and “fully accepted” if the acceptance ratio is 1. It is “rejected” if the acceptance ratio is 0. We say that a basket is “accepted” to mean that its parent order is accepted.

For sell orders, the acceptance ratio applies to the order and not to each offered quantity for the individual products in the order. All products in a sell order are accepted to the same proportion. For sell orders which have non-zero offered quantities for more than one product, the acceptance ratios for all products must be equal (subject to rounding, discussed in Section 8).

Acceptance of sell orders is subject to the following clearing rules:

Acceptance Ratios:

1. A parent order is non-curtailable. It must have an acceptance ratio of either 1 or 0. It must either be fully accepted or rejected.
2. A child order or substitutable child order is curtailable. It may be accepted at any ratio between 0 and 1 inclusive. It may be partially accepted, fully accepted, or rejected.
3. All substitutable child orders in a basket form a single substitutable family. The sum of the acceptance ratios in a substitutable family must be less than or equal to 1.

Linking and Looping:

4. A child order or substitutable child order may be accepted only if the parent order in its basket is accepted.
5. If a basket is looped to another basket, then either both baskets must be accepted, or both rejected.

Mutual Exclusivity:

6. If a basket is accepted, then any other basket with which it is mutually exclusive (i.e., defined on the same or a concomitant service window) must be rejected.

Order Surplus:

7. A child order or a substitutable child order may be accepted only if it has non-negative order surplus.
8. A parent order in a basket which is not looped to another basket may be accepted only if the total order surplus of all accepted orders in the basket is non-negative. A parent order in a basket which forms part of a looped family may be accepted only if the total order surplus of all accepted orders in the looped family is non-negative.

Sell orders from the same basket can be accepted simultaneously (subject to clearing rule 3). Rule 6 means that no two orders from baskets that are mutually exclusive may both be accepted. Sell orders from non-overlapping baskets may be simultaneously accepted.

All child orders of the same basket can be accepted independently of each other. All substitutable child orders of the same basket can be accepted independently of each other, provided that the sum of all the acceptance ratios of all these orders cannot exceed 100%. This property is similar to the use of substitutability families for buy orders (see Section 5.1).

Note that because each product in an accepted order has the same acceptance ratio, there may be transfers of surpluses between the products in the order. In other words, a certain product in the order may be paradoxically accepted, but the order will not be paradoxically accepted (i.e., it will not have negative surplus).

Clearing rule 8 implies that a parent order need not have non-negative surplus to be accepted. A parent order may have negative surplus on its own but benefit from a transfer of surplus from other orders in its basket or from orders in baskets to which it is looped.

For the avoidance of doubt, a sell order may be paradoxically rejected. It may be rejected despite having a positive surplus.

Figure 3 shows an example of baskets defined for one unit in different periods that presents all the attributes and characteristics of sell orders and baskets detailed in this section (e.g., parent orders, child orders, looped baskets, mutually exclusive baskets, etc.).

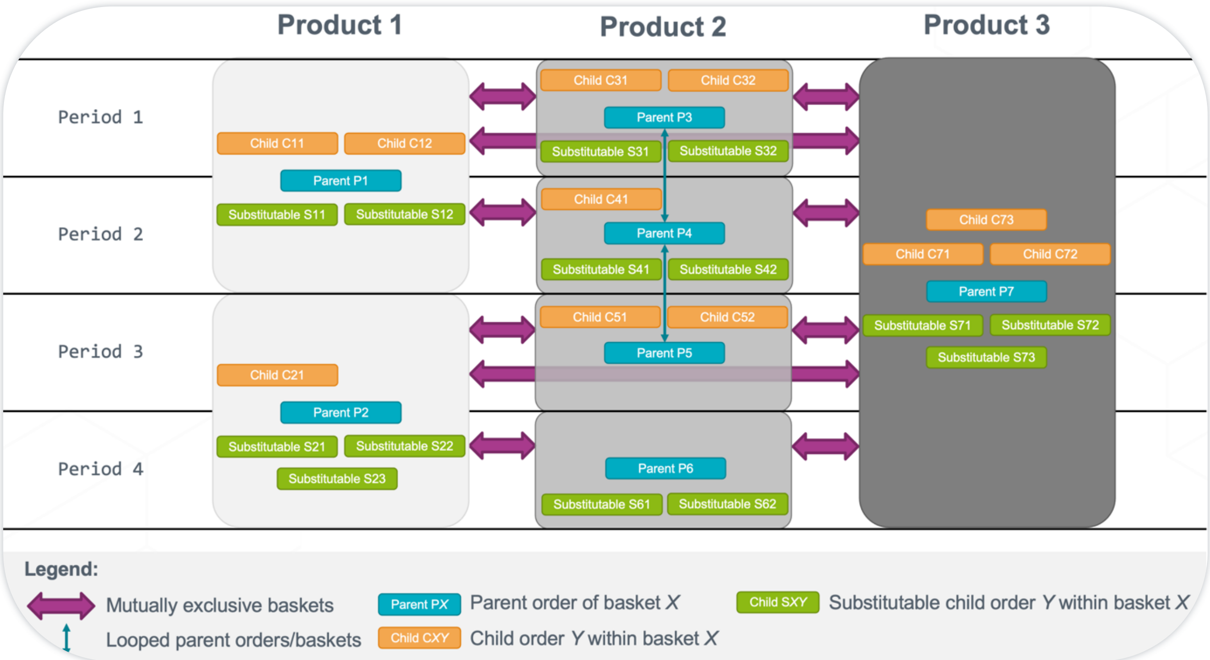

Figure 3 - An example of possible basket organisation over different service windows. Baskets 3, 4 and 5 are looped linked, meaning parent orders P3, P4 and P5 are looped linked and there can be transfers of surpluses between them. Here, the periods are such that the baskets of the different products match their respective service windows. For example, product 1 could have service windows of 2 hours, Product 2 of 1 hour and product 3 of 4 hours.

6.3. Use cases

The significance and application of the order structures and clearing rules outlined above is illustrated in the following section. In particular, this document explains how participants can:

- specify which portion (if any) of their offer may be curtailed;
- control the mix of different products that are delivered simultaneously by their unit;
- offer a menu of alternative services to the Auction, of which only a portion will be accepted;
- Link service windows through time within a service day, for example to amortise start costs over longer periods.

Curtailability:

Participants may wish to offer their capacity as "non-curtailable", meaning they require their order to be accepted for the entire offered quantity, or else rejected. Conversely, participants may be willing to have their offers curtailed, meaning that they may be willing to have their order cleared for a lower quantity than the total offered quantity. Participants who are willing to have their offer curtailed may require at least some minimum portion to be accepted, or they may be willing to be accepted for any quantity between 0 and the offered quantity i.e., to have a minimum acceptance ratio strictly between 0 and 1.

All these preferences can be expressed with the sell order structure in the market design. Parent orders are non-curtailable, while child orders and substitutable child orders are fully curtailable. Participants who want to offer their capacity as non-curtailable should use only parent orders. Participants who are willing to be curtailed subject to the acceptance of minimum quantity should use a parent order for the non-curtailable portion of the order, and child orders or substitutable child orders for the curtailable portion.

All orders must belong to exactly one basket, and all baskets must contain exactly one parent order. Participants who wish to offer fully curtailable orders (i.e., with no reserved minimum quantity) can use a parent order with an offered quantity of 0 MW, together with one or more child or substitutable child orders in the same basket.

Splitting:

A key feature of the market design is the opportunity (but not the obligation) for a unit to be accepted for different products in the same service window. In this document, we refer to this feature as "splitting". By offering quantities of more than one product in a single sell order, the provider can bundle products together and thus oblige splitting (subject obviously to the acceptance of the sell order). By including different products in different sell orders but within the same basket, the provider can indicate a willingness (but not the requirement) to split. An unwillingness to split (i.e., a preference to not have different, heterogeneous products selected in the same service window) is expressed by offering only one single product in any basket.

Co-optimisation:

When a unit has the capability of providing more than one of the services, the provider may wish to place multiple, alternative offers in the Auction, and let the Auction clearing algorithm allocate the unit to the service that will clear the market most efficiently (i.e., best maximise market welfare, subject to the constraints). We have defined this possibility as “co-optimisation”. In contrast, without co-optimisation the provider has to choose in advance which of the various services to offer into the Auction. From an overall market perspective, an Auction that does not admit co-optimisation is almost certainly less efficient (it is unlikely that all providers will make the best choices). From a provider perspective, there is the risk that the unit is not cleared for the selected service while an alternative service is under-subscribed. Co-optimisation therefore reduces risk for Market Participants while increasing overall market liquidity and reducing procurement costs.

The new market design for ancillary services implements co-optimisation by two different Auction features: “mutually exclusive baskets” and “substitutable families”.

All sell orders are associated with exactly one basket. Participants may offer multiple baskets into each Auction. If the service windows of any two baskets overlap in time, then the two baskets are mutually exclusive. This means that if any order in one of the baskets is accepted, then all orders in the other basket must be rejected. In this way, the baskets enable a participant to present a binary alternative to the Auction algorithm.

A participant on the sell side may designate some sell orders as “substitutable children”. All the substitutable children within any single basket form a single substitutable family. These orders may be rejected, partially accepted (i.e., accepted for only part of the offered volume), or fully accepted. We define the acceptance ratio of these orders as the proportion of the offered volume that is accepted (e.g., the acceptance ratio is 100% if the order is fully accepted, or 50% if only half of the offered volume is accepted). More than one of the orders within a substitutable family may be partially accepted (so they are not binary), but the sum of the acceptance ratios of all the orders in a substitutable family may not exceed 100%. Substitutable child orders are therefore partial substitutes for each other (within a substitutable family). Substitutability of child sell orders is similar to the concept of substitutability families for buy orders (see Section 5.2).

Substitutable families and mutually exclusive baskets have different applications within the market design. Substitutable children are curtailable (i.e., they may be partially accepted). Not all Market Participants may be willing to be curtailed. Some may prefer to offer non-curtailable capacity, so that it is either fully accepted or rejected. Furthermore, substitutable families enable a combination of the orders to be accepted, and this may imply the splitting of the unit across different services in the same service window. Some units may not have the technical capability to stack or split services, and indeed frequency response will not be allowed to be stacked or split with reserve on Day 1. Mutually exclusive baskets are therefore an additional feature of the market to facilitate participants to offer binary alternatives, and to co-optimise amongst different services without risk of undesired splitting.

Looped Baskets:

Two or more different baskets defined on the same unit and on non-overlapping service windows can be looped together. Looped baskets must either be all accepted or all rejected concurrently by the algorithm. In practice, this means that if one parent order within a set of looped baskets is rejected, then all the orders of all the baskets in this set of looped baskets must be rejected; if (at least) one parent order within a set of looped baskets is accepted, then (at least) the parent orders of all looped baskets must also be accepted.

Looping baskets implies that there can be transfers of “economic surplus” across looped baskets: the algorithm will ensure that the looped baskets are jointly profitable if accepted (whereas only non-looped baskets are individually guaranteed to be compatible with the market clearing prices).

6.4. Specifications and validations for baskets and sell orders

Specification of Baskets:

Element	Comment
Basket ID	Each basket has a unique identifier
Unit	A basket is associated with delivery of Ancillary Services from a single unit.
Service Type	One of the available service types (e.g. Frequency Response, Quick Reserve, Slow Reserve).
Service Window	A single service window, consistent with the service type
Looped Basket ID	A basket may be looped to another basket defined on the same unit and a non-overlapping service window.
Sell Orders	A set of sell order IDs that are contained in the basket

Unit: A unit consists of one or several assets of the Seller. A unit is identified to the Auction by a unique Unit ID. Sell orders with respect to a unit may be submitted only by an authorised user associated with the Seller.

Each unit is prequalified for the provision of some or all of the different products. Baskets and orders may be defined only on products for which the unit is prequalified.

The total available capacity of the unit in both directions is also registered to verify that the different volumes specified via the sell orders and the baskets are coherent with the unit’s capability. This validation is performed at the time of the order submission. However, notwithstanding any validation checks performed by the EAC system, Sellers remain responsible for the deliverability of their offers. (See discussion of “Offered Quantities”, below).

The maximum number of baskets a unit can submit in each Auction is configured by the market operator.

Service Type: A basket is defined on a single service type common to each order contained in the basket (e.g., Frequency Response). Inside a basket, orders can only indicate offered quantities from products from the chosen service type. Consequently, a basket can only contain products for which splitting is allowed.

Service Window: A basket is defined on a single service window common to each order contained in the basket. The service window must be compatible with the products covered by the service type.

The exclusivity relationships between baskets are fully determined by their service window. No information about mutual exclusivity is required from the user.

Looped Basket ID (optional): A basket may optionally be looped to another basket defined on the same unit. More than two baskets may be looped in this way. Two or more baskets looped together form a “looped family”. No baskets in a looped family may be defined on concomitant service windows (i.e., all baskets in a looped family must be defined on non-overlapping serviced windows).

For the avoidance of doubts, a basket may loop to another basket of a different service type.

Sell Orders: A basket must contain exactly one parent order. A basket may contain between 0 and a certain number of child orders, and additionally between 0 and a certain number substitutable child orders, configurable by the market operator. Each sell order is associated with exactly 1 basket.

Specification of Sell Orders:

Element	Comment
Sell Order ID	Each order has a unique identifier
Order Type	Either “Parent”, “Child”, or “Substitutable Child”.
Offered Quantities	A set of product-volume pairs, representing the offer of capacity for the products corresponding to the service type of the basket.
Price	A single price, in £/MW/h.

Order Type: A sell order must be a parent order, a child order, or a substitutable child order.

Offered Quantities: The offered products must be consistent with the service type of the basket. An order is not allowed to specify product-volume pairs for products for which its unit is not qualified.

All offered quantities are integer MWs. The offered quantity for a product may be zero. The offered quantities with respect to a parent order may be zero for all products (i.e., all quantities in the order may be zero). A child order and a substitutable child order must have at least one product with a positive offered quantity.

Offered quantities are validated at the time of order submission to ensure that the quantities do not imply a violation of the total capacity of the underlying unit. For each order, we calculate the total offered quantity for low/positive deliveries. Then for each basket, we calculate the sum of (i) the total quantity offered by the parent order, (ii) the total quantity offered by all child orders and (iii) the maximum total quantity from among all the substitutable child orders. This sum should be equal to or below the registered dispatch capability of the unit. A similar check is performed for high/negative deliveries.

It remains the responsibility of the Seller at all times to ensure that each basket offered into the Auction is fully deliverable by the unit (considering both capacity and energy capabilities and any other relevant constraints) in the event that the basket is fully accepted by the Auction clearing algorithm in accordance with the Auction clearing rules.

Price: The price is expressed in pounds and pence (multiple of 0.01 £/MW/h). The price must be between the market's minimum and maximum prices. For the avoidance of doubt, the price of a sell order may be negative.

7. Pricing rules

This section explains how the market clearing prices are computed once a welfare-optimal feasible selection of buy and sell orders is found. Different pricing approaches were assessed during the development of the EAC, including *Marginal Pricing*. Following consideration of the possible options, NGESO decided that *Cost Minimisation Pricing* will be used for the computation of clearing prices in the EAC. A description of *Cost Minimisation Pricing* as applied in the EAC is provided below.

7.1. Uniform market clearing price

For each service window and each product, a uniform market clearing price is determined. This means that all accepted buy and sell orders for a given service window and product are settled at the same clearing price. In this section, we explain what conditions will hold between the market clearing price and the acceptance levels of the buy orders and sell orders.

7.1.1. Market clearing price and sell orders

The market clearing prices must be such that any accepted sell order is not paradoxically accepted. This means that the revenue of a sell order must be greater than or equal to what the order requires, based on its offered quantities and its limit price, together with transfers of surplus to other sell orders.

The individual surplus of an accepted order can be negative, but its (net) surplus must be non-negative.

i. The individual surplus of a (fully or partially) accepted order is the difference between the value at which the order settles (i.e., market clearing prices multiplied by cleared volumes) and the contribution of the order in the objective function (i.e., the limit price of the order multiplied by the cleared volume). The individual surplus is thus the profit made by the order on its own.
ii. The (net) surplus is the individual surplus augmented by possible positive transfers of surpluses received from other accepted looped orders (or received from its child orders in case of a parent order) and diminished by possible negative transfers of surpluses given to other accepted looped orders (or given to its parent order in case of a child order).

Consequently, when the acceptances of multiple orders are linked, the entire set of accepted linked orders cannot be paradoxically accepted and hence cannot suffer from a loss (i.e., the links indicate that the Seller can agree that one order is paradoxically accepted provided that the loss of this order is at least compensated by the profit of another accepted linked order).

Sell orders can be (partially or fully) paradoxically rejected for the following reasons:

a) A non-curtailable sell order can be paradoxically rejected because its volume exceeds unsatisfied demand quantity and/or acceptance of another (possibly more expensive) sell order leads to higher market welfare.

b) A curtailable sell order can be paradoxically rejected because its parent order is rejected (as explained in (a), the parent order can itself be paradoxically rejected).
c) A curtailable sell order can be (fully or partially) paradoxically rejected so that its parent order can be accepted despite being more expensive (remember that all curtailable orders have a parent order).
d) A substitutable curtailable sell order can be paradoxically rejected in view of the market prices because a more profitable substitutable order in the same basket is selected.

7.1.2. Market clearing price and buy orders

Buy orders' limit prices have no impact on the computation of the market clearing prices. They are used as a mere indication of the Buyer's willingness to procure balancing capacity in the welfare maximisation problem but do not enforce any bounds on the market clearing prices.

In other words, any buy order can be
1. either (fully or partially) paradoxically accepted (if allowed to be paradoxically accepted, i.e. if the relevant parameter in the order is set to TRUE)
2. or (fully or partially) paradoxically rejected

if such treatment increases the welfare objective function value or decreases the total procurement cost. This includes notably the situations where the Buyer procures more volume than what it asked for at the given price (referred to as overholding, see Section 5.3§2 for more details).

7.2. Price indeterminacy and procurement cost minimisation

After conditions detailed in Section 7.1 have been imposed on the market clearing prices, there might remain several feasible sets of market clearing prices. Among all feasible sets of market clearing prices, the Auction will select one that minimises the procurement cost[3].
The procurement cost is defined as the sum of the accepted sell volumes times their corresponding market clearing price.

7.3. Technical minimum and maximum prices

All prices will be required to be between a minimum market price bound and a maximum market price bound.
1. The minimum and maximum market price bounds will be configurable numbers with maximum 2 decimals (pounds and pence). Their units will be £/MW/h.
2. Negative minimum market price bounds will be allowed.
3. The minimum market price bound is the same for all products and all service windows.
4. The maximum market price bound is the same for all products and all service windows.

[3] Note that there might be several sets of market clearing prices equally minimising the procurement cost. Hence, the minimisation of the procurement cost does not lift all price indeterminacies.

5. The minimum and maximum market price bound are set by the Market Operator.

The price bounds enforce that

a) The (unrounded and rounded) market clearing prices are within the minimum and maximum market price bounds.
b) The bidding prices of all buy and sell orders are within the minimum and maximum market price bounds.

8. Accuracy, precision and rounding

This section introduces the properties of the rounded and unrounded results.

Precisions to be satisfied by input data have been presented in Sections 5 and 6 (i.e., input volumes must be integers in MW and input prices must be pounds and pence).

Results will comply with the following restrictions.[4]

1. Unrounded prices will have an accuracy no worse than 0.0001 £/MW/h.
2. Rounded prices will have a precision of 0.01 £/MW/h (i.e., pounds and pence).
3. Unrounded volumes will have an accuracy no worse than 0.001 MW.
4. Rounded volumes will have a precision of 1 MW.
5. Unrounded transfers of surplus will have an accuracy no worse than 0.001 £.

8.1. Procedure for rounding prices

Prices are rounded up to the closest multiple of 0.01 £/MW/h. For example:

- If the unrounded price is 10.330 £/MW/h, the rounded price is 10.33 £/MW/h.
- If the unrounded price is 10.331 £/MW/h, the rounded price is 10.34 £/MW/h.
- If the unrounded price is 10.335 £/MW/h, the rounded price is 10.34 £/MW/h.
- If the unrounded price is 10.339 £/MW/h, the rounded price is 10.34 £/MW/h.
- If the unrounded price is 10.340 £/MW/h, the rounded price is 10.34 £/MW/h.
- If the unrounded price is -10.330 £/MW/h, the rounded price is -10.33 £/MW/h.
- If the unrounded price is -10.331 £/MW/h, the rounded price is -10.33 £/MW/h.
- If the unrounded price is -10.335 £/MW/h, the rounded price is -10.33 £/MW/h.
- If the unrounded price is -10.339 £/MW/h, the rounded price is -10.33 £/MW/h.
- If the unrounded price is -10.340 £/MW/h, the rounded price is -10.34 £/MW/h.

It was decided to round prices up to avoid that some sell orders become paradoxically accepted with respect to the rounded prices. In the example of Figure 4, the market clearing price will be 1000 £/MW/h for Product 1 and 4000/3 £/MW/h for Product 2.

- If prices are rounded up, rounded market clearing prices are 1000 £/MW/h for Product 1 and 1333.34 £/MW/h for Product 2. The surplus of the orange sell order is $100(1000 - 2000) + 300(1333.34 - 1000) = -100000 + 100002 = 2\,£$. The orange order is thus not paradoxically accepted with respect to the rounded prices.
- If prices are rounded to the closest multiple of 0.01 £/MW/h, rounded market clearing prices are 1000 £/MW/h for Product 1 and 1333.33 £/MW/h for Product 2. The surplus of the orange sell order is $100(1000 - 2000) + 300(1333.33 - 1000) = -100000 + 99999 = -1\,£$. The orange order is thus paradoxically accepted with respect to the rounded prices.

Finally, note that rounding prices up means some sell orders can become (partially) paradoxically rejected due to the rounding.

[4] We use the following wording: Precision of a data refers to its number of decimals. Accuracy of a data refers to its distance to the true value.

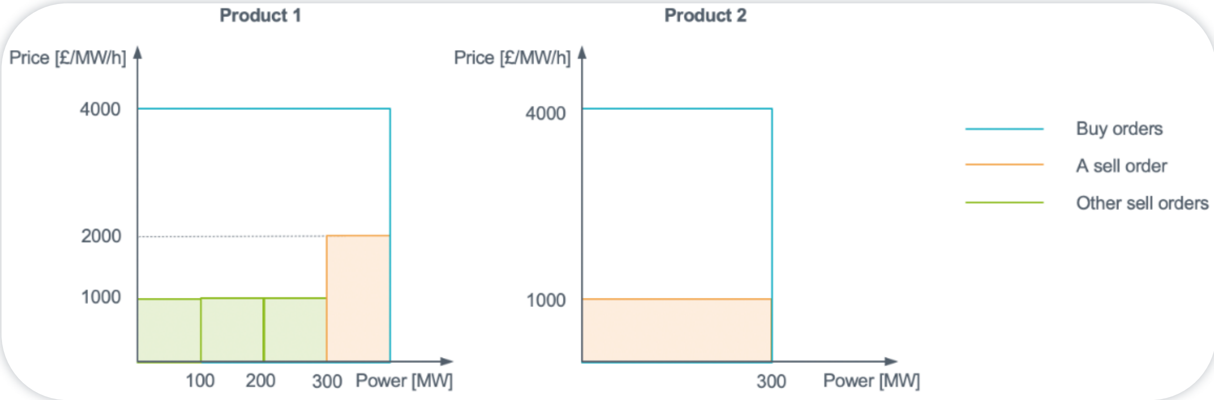

Figure 4 – An Auction covering two products and a single service window. The welfare maximising solution consists of fully accepting all orders. The prices minimising the procurement cost are 1000 £/MW/h (as small as possible due to the large amount of accepted volume) on Product 1 and 4000/3 £/MW/h (as small as possible but still compensating for the loss incurred to the orange order on Product 1) on Product 2.

8.2. Procedure for rounding accepted volumes

First, accepted sell volumes are rounded:

- to the nearest integer for parent orders and child orders (e.g., 10.5 is rounded to 11, and 10.3 is rounded to 10).
- down to the next smallest integer for substitutable child orders (e.g., 10.7 is rounded to 10, and 10.3 is rounded to 10). Rounding down is preferred to guarantee that the substitutable orders will not be accepted for more than the capacity of their unit (for example, 2 substitutable orders offering both 9 MW and accepted both to 50% would have a rounded volume of 5 MW each if rounding to the closest integer was preferred and would consequently violate the unit's capacity if it was 10 MW).

Second, all accepted buy volumes are rounded to the nearest integer. For each product and each service window, the sum of the rounded accepted buy volumes might then not match the sum of the rounded accepted sell volumes. The difference between the two is called the rounding residual. The rounding residual is guaranteed to be a multiple of 1. The rounding residual will be cancelled by distributing rounding ticks of 1 MW. Rounding ticks are distributed to the Buyer's orders only by increasing or decreasing their rounded accepted volume (while the unrounded volume is kept unchanged). Thus, in case the rounding residual is not zero, the Buyer will procure a little more or a little less volume than expected given the market clearing price. It can be shown that distributing rounding ticks to the Buyer is always feasible if we accept that the rounded buy volume can go above the total volume that the Buyer is willing to procure (in which case the Buyer's order with the smallest price will have a rounded accepted volume larger than its bidding volume).

Appendix A. Algorithm description

This section describes the algorithm used to clear the EAC Auction. The algorithm aims at finding the market clearing solution that generates the largest social welfare while satisfying all market rules. A market clearing solution provides an acceptance ratio for each submitted order and a market clearing price for each product and service window.

A.1 Algorithm overview

The welfare maximisation primal problem, also referred to as the volume problem, is a Mixed Integer Linear Program (MILP); the objective function and the constraints are all linear, but some variables are integer. The presence of integer variables (in this case binary variables) results from the modelling of exclusive constraints between relevant baskets. The N-SIDE algorithm relies on the branch and bound (B&B) method to find feasible solutions to the primal problem that are as close as possible to optimality. The B&B method is described in Section A.2.

Once a primal solution is found, the clearing algorithm solves the price problem to find the market clearing prices that follow the pricing rules described in Section 7, proceeds with the post-processing (rounding of prices and volumes), stores the solution, and resumes the exploration of the B&B tree in search of a better solution. Figure 5 provides an overview of the algorithm. Section A.2 further describes the Branch and Bound method.

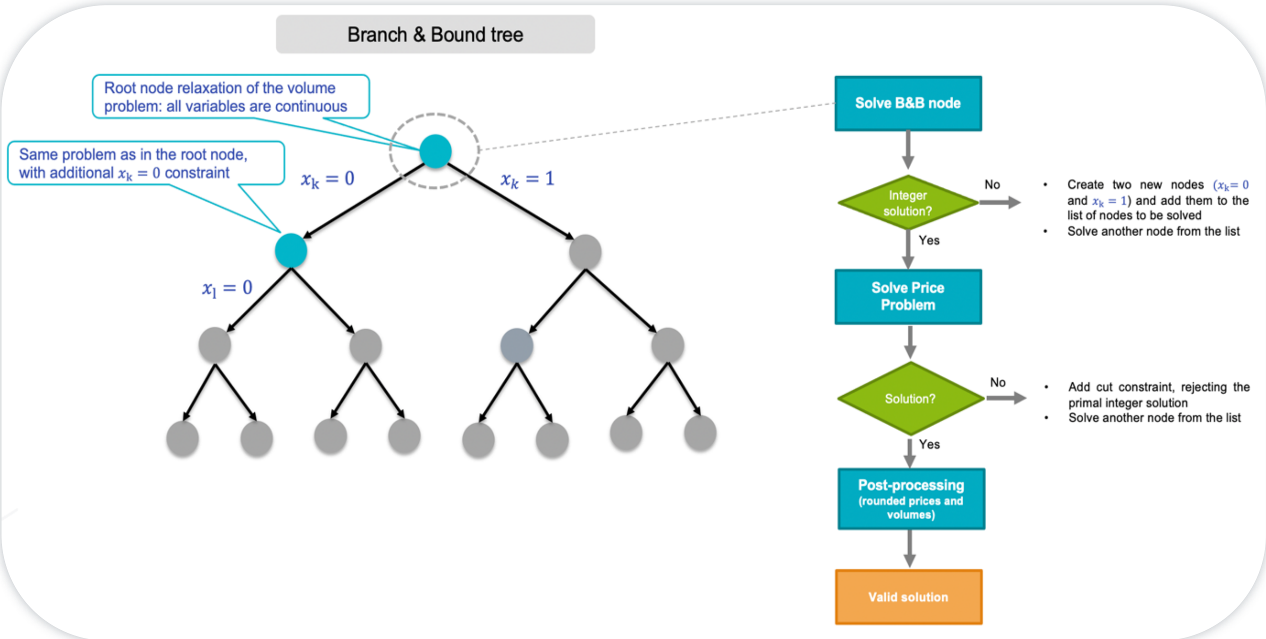

Figure 5 - Algorithm overview. The Branch & Bound tree is explored to find integer feasible solutions for the volume problem. Once such an integer solution is found, the algorithm computes the market clearing prices, proceeds with the post-processing (rounding), stores this validated solution, and resumes the exploration of the Branch & Bound tree.

A.2 The Branch & Bound method

The Branch & Bound method is a state-of-the-art method for solving mixed integer linear programs (MILP). The algorithm starts by solving a relaxation of the initial MILP problem where all variables are allowed to be continuous, including those required to be binary in the final solution, see Figure 5. This solution gives an upper bound on the attainable social welfare since the relaxed problem is less constrained. If the solution is integer, then it is the optimal primal solution. Otherwise, the algorithm branches on one of the variables that is violating its integer constraint, called the *branching variable*. Two nodes are then created:

- The first node is the same primal problem but with an additional constraint setting the *branching variable* to 0.
- The second node is the same problem, but with an additional constraint setting the *branching variable* to 1.

The algorithm picks one of the nodes and solves the relaxed primal problem with this additional constraint on the branching variable. If the solution is again non-integer, then the algorithm branches on an additional variable. The algorithm keeps iterating until finding an integer solution.

Once a primal integer solution is found, the algorithm solves the corresponding price problem. If this price problem is feasible (which is always the case with the *Cost Minimisation Pricing* rule) then the algorithm derives a valid market solution, and a lower bound on the primal problem.

The algorithm keeps exploring the B&B tree until finding the optimal solution or until reaching the Auction time limit.

Throughout the B&B exploration, the algorithm stores and refines the following information on the primal problem:

- An upper bound on the optimal welfare. The objective value of the root node solution provides a first upper bound on the optimal social welfare. This upper bound is refined when exploring the tree.
    - Child nodes are always more constrained than their parents and thus have a worse objective value.
    - The upper bound is the maximum of the objective values of the nodes with open child(ren).
- A lower bound on the optimal welfare, defined as the objective value of the best-found integer solution. When a node of the B&B tree provides a lower welfare than that lower bound, the branch does not need to be explored further and can be pruned since there is no chance to get a better solution by branching further in that direction.
- The optimality gap, defined as the difference between the upper bound and the lower bound. This provides an upper bound on the distance between the optimal solution and the best-found solution. If the gap is null, then the best solution found is proved to be optimal and the tree exploration can stop.

Appendix B. Illustrative examples on buy orders

B.1 Substitutability between products on same service window

This example illustrates a simple case of substitutability between Product 1 and Product 2. In its order book, the Buyer requests

i. 600MW of Product 1 (order 1),
ii. 200MW of Product 2 (order 3 and 4), and
iii. 200MW in either Product 1 or Product 2 (orders 2 and 5).

Order ID	Product	Service Window	Price [£/MW/h]	Volume [MW]	Substitutability family
1	Product 1	Period 1	10.00	600	
2	Product 1	Period 1	5.00	200	1
3	Product 2	Period 1	12.00	100	
4	Product 2	Period 1	8.00	100	
5	Product 2	Period 1	5.00	200	1

Table 3 - Buyer order book with 2 substitutable orders.

Figure 6 – Buy order book with orders 2 and 5 substitutable.

The order acceptance will follow the welfare maximisation principle. The substitutability constraint is enforced in the EAC by the constraint $x_2 + x_5 \leq 1$, where x_o is the acceptance ratio of order o. Some possible outcomes, depending on available sell orders may be:

- 600 MW Product 1 and 400 MW Product 2 (accepting 0% of order 2 and 100% of order 5)
- 650 MW Product 1 and 350 MW Product 2 (accepting 25% of order 2 and 75% of order 5)
- 800 MW Product 1 and 200 MW Product 2 (accepting 100% of order 2 and 0% of order 5)
- 800 MW Product 1 and 100 MW Product 2 (accepting 100% of order 2 and 0% of order 5, but also rejecting order 4, i.e., in case of scarcity on Product 2)

B.2 Several substitutability families

This example contains two substitutability families. Family 1 is identical to the family of the previous example. Family 2 states that the Buyer can substitute 200 MW of Product 2 with 300 MW of Product 3. The following substitutability constraints are enforced by the algorithm.

- Family 1: $x_2 + x_5 \leq 1$.
- Family 2: $x_6 + x_8 \leq 1$. Thus, $\frac{q_6}{200} + \frac{q_8}{300} \leq 1$ with $q_6$and q_8 the accepted volumes (MW).

Note two orders belonging to the same substitutability family may have different volumes.

Order ID	Product	Service Window	Price [£/MW/h]	Volume [MW]	Substitutability family
1	Product 1	Period 1	10.00	600	
2	Product 1	Period 1	5.00	200	1
3	Product 2	Period 1	12.00	100	
4	Product 2	Period 1	8.00	100	
5	Product 2	Period 1	5.00	200	1
6	Product 2	Period 1	5.00	200	2
7	Product 3	Period 1	8.00	100	
8	Product 3	Period 1	5.00	300	2

Table 4 – Buyer order book with 2 families of substitutable orders. Two substitutable orders of the same family can have different prices and volumes.

B.3 Substitutability between products on different concomitant service windows

This example illustrates a case of substitutability between products defined on concomitant service windows.

The substitutability constraint $x_1 + x_2 \leq 1$ is enforced by the algorithm (x_o is the acceptance ratio of order o). Thus, $\frac{q_1}{100} + \frac{q_2}{200} \leq 1$ with $q_1$and q_2 the accepted volumes (MW).

Order ID	Product	Service Window	Price [£/MW/h]	Volume [MW]	Substitutability family
1	Product 1	[23:00-03:00]	5.00	100	1
2	Product 2	[23:00-07:00]	8.00	200	1

Table 5 – Buyer order book with 2 substitutable orders having different prices and volumes. The substitutable orders span on different service windows but these service windows have at least one common instant in common.

N-SIDE is a deeptech company that empowers organizations in the life sciences and energy sectors to make better decisions and optimize the use of critical resources.

We're doing so by combining deep industry expertise with applied mathematics and artificial intelligence into easy to use and cutting-edge software that transforms uncertainty and complexity into deterministic outcomes.

In Life Sciences, we streamline the clinical supply of pharmaceutical and biotech companies by accelerating clinical plans, mitigating risks and curbing drug waste.

In Energy, we accelerate the transition towards renewables and electrification by enabling leading grids and market players in making better, faster and safer decisions.

N-SIDE is ranked among the Best Workplace™ of the Great Place to Work® Institute Belgium and is also a certified B Corporation™. For more information, visit www.n-side.com