Capturing Electricity Market Dynamics in Strategic Market Participation Using Neural Network Constrained Optimization

Dolanyi, Mihaly; Bruninx, Kenneth; Toubeau, Jean Francois; Delarue, Erik

DOI
10.1109/TPWRS.2023.3242356

Publication date
2024

Document Version
Final published version

Published in
IEEE Transactions on Power Systems

Citation (APA)
Dolanyi, M., Bruninx, K., Toubeau, J. F., & Delarue, E. (2024). Capturing Electricity Market Dynamics in Strategic Market Participation Using Neural Network Constrained Optimization. *IEEE Transactions on Power Systems*, *39*(1), 533-545. https://doi.org/10.1109/TPWRS.2023.3242356

Capturing Electricity Market Dynamics in Strategic Market Participation using Neural Network Constrained Optimization

Mihály Dolányi, Kenneth Bruninx, Jean-François Toubeau, and Erik Delarue

***Abstract*—In competitive electricity markets, the optimal bid or offer problem of a strategic agent is commonly formulated as a bi-level program and solved as a mathematical program with equilibrium constraints (MPEC). If the lower-level part of the problem can be well approximated as a convex problem, this approach leads to a global optimum. However, electricity markets are governed by non-convex (partially known) constraints and reward functions of the participating agents. In this paper, an alternative data-driven paradigm, labeled as a mathematical program with neural network constraint (MPNNC), is developed. The method uses a neural network to represent the mapping between the upper-level (agent) decisions and the lower-level (market) outcomes, i.e., it replaces the lower-level problem with a surrogate model. In the presented case studies, the proposed model is used to find the optimal load shedding strategy of a strategic load-serving entity. First, the MPNNC performance is compared to the MPEC approach, both in convex and non-convex environments, showing that the proposed MPNNC achieves similar performance to an ideal MPEC that has perfect knowledge of the simulated market environment. Then, aggregated supply curves from the Belgian spot exchange are used to assess the potential gains of using the developed model in real-life applications.**

***Index Terms*—Strategic bidding, electricity markets, mathematical program with neural network constraint**

Nomenclature

Parameters

α_i	Minimum fuel cost of power plant i (€/t)
β_i	Marginal cost of power plant i (€/MWh)
$\underline{\gamma_i}$	Minimum emission of power plant i (tCO_2/t)
$\overline{\Delta G_i^{\downarrow}}$	Maximum ramp-down rate of power plant i (MW/dt)
$\overline{\Delta G_i^{\uparrow}}$	Maximum ramp-up rate of power plant i (MW/dt)
$\overline{G_i}$	Maximum capacity of power plant i (MW)
$\underline{G_i}$	Minimum capacity of power plant i (MW)
$\overline{AF_{i,t}}$	Availability factor of power plant i at time-step t (-)
c_i^{gen}	Average generation cost of power plant i (€/MWh)
D_t	Demand at time-step t (MWh)
em_i	Average emissions of power plant i (tCO_2/MWh)
p^{CO_2}	Carbon price (€/tCO2)
$VOLL$	Value of lost load (€/MWh)
MDT_i	Minimum down time of power plant i
MUT_i	Minimum up time of power plant i
STC_i	Start-up cost of power plant i (€/start-up)

Sets and indices

$i \in \mathcal{I}$	Set of generators
$i \in \mathcal{I}^D$	Set of dispatchable generators
$i \in \mathcal{I}^V$	Set of variable (renewable) generators
$t \in \mathcal{T}$	Time-steps

Variables

λ_t	Electricity market price at time t (€/MWh)
ens_t	Energy not served at time t (MWh)
$\mathrm{g}_{i,t}$	Generation of unit i at time-step t (MWh)
$\mathrm{v}_{i,t}$	Start-up status of i at time step t $\{0,1\}$
$\mathrm{w}_{i,t}$	Shut-down status of i at time step t $\{0,1\}$
$\mathrm{z}_{i,t}$	On/off-status of power plant i at time step t $\{0,1\}$

I. Introduction

In competitive electricity systems, the production, and consumption of electrical energy are coordinated by markets wherein self-interested agents trade demand for and supply of electrical energy. In order to improve their individual strategies, market actors are interested in defining mathematical models that capture the market dynamics. Such a model may then be integrated into the market agent's scheduling problem to maximize its reward. Traditionally, this goal is achieved by modeling a Stackelberg Game [1] as a bi-level program, most frequently solved as a single-level Mathematical Program with Equilibrium Constraints (MPEC) [2], [3].

In such MPECs, the electricity market is represented in the lower level via its optimality conditions, and the decision problem which intends to optimally stimulate the market forms the upper-level (as depicted in Fig 1). However, the convex market model that is most frequently used in the bi-level programming paradigm may be an erroneous approximation of the real market dynamics due to three main reasons: (i) the dynamically changing risk-averse and potentially non-rational objectives of the market agents are difficult to estimate and are

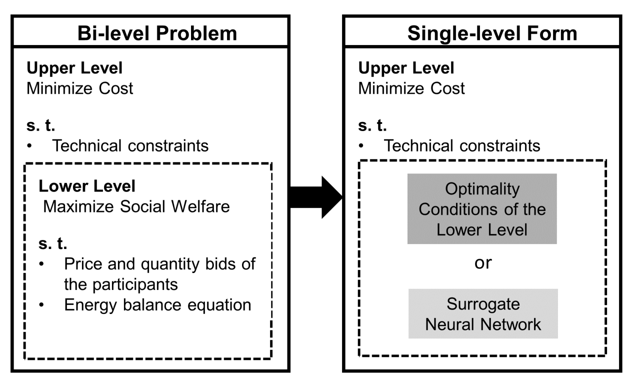

Fig. 1: Schematic showing the bi-level programming structure on the left, and the MPEC and MPNNC reformulations on the right side.

continuously impacted by information asymmetry, (ii) there is a complex set of non-convex constraints (e.g., minimum up and down times of power plants) characterizing the feasibility space of the market clearing problem, and (iii) the model parameters, as well as the representation of uncertainties, are only partially known.

To address problem (i), Guo et al. [4] propose an inverse learning approach to reconstruct the market agents' reward functions for the Australian electricity market. To deploy similar methods in other markets, the agents' bids need to be available, which is not the case, e.g., in European power exchanges.

To address (ii), Ye et al. [5] offer an option for incorporating non-convexities into MPECs, but it happens at the expense of increased computational burden. Including every non-convex feature of a real market environment may lead to computationally intractable problems. Moreover, (i) remains unaddressed in this line of research.

In conclusion, current model-based representation of the lower-level problem would not lead to a reliable replication of the price formation mechanism, thus the resulting bidding strategy may exhibit poor real-life performance.

To improve consistency between models and real-world data, reinforcement learning (RL) may be seen as an alternative to the model-based MPEC approach, leading to a fully data-driven solution method wherein the optimal policy is directly learned from interactions with the market environment. In [6], Boukas et al. improve the trading strategy in the (multi-stage) intra-day market using deep Q-learning, i.e., a combination of value-based RL with deep neural networks [7]. The same problem is tackled in [8] using policy-based RL, where a stochastic threshold policy is approximated with a parametric function to boost both performance and computational tractability. However, typical model-free RL methods are limited in their ability to embed expert knowledge (i.e., they are not able to consider specific analytical equations to drive the solution towards feasible and optimal points), which may slow the convergence of the learning procedure, and can lead to sub-optimal decisions for unobserved system states.

Motivated by the above challenges in existing approaches, hybrid modeling strategies intend to combine the strengths of model-based and model-free approaches. This can be achieved by (i) incorporating data-driven surrogate learning models into existing optimization problems [9]–[11], or by (ii) developing novel methodologies to embed hard constraints in model-free [12], [13] and in model-based reinforcement learning methods [14], [15]. To fully benefit from the modeling power, as well as the convergence and potential guarantees of optimality of optimization models, this paper focuses on the first category. This approach has previously been used to capture the feasible space and non-convexities of optimal power flow problems [16]–[21], focusing on reducing the computational burden. This work instead aims to leverage the learning abilities of neural networks to enable a data-driven representation of the (imperfectly known) LL optimization problem in the bi-level formulation of a load agent's optimal bidding strategy, leading to a Mathematical Program with Neural Network Constraint (MPNNC) (right part of Fig 1).

The integration of various machine learning models, e.g., neural networks, into optimization models may be done by reformulating the trained neural network as mixed-integer constraints [9], [20] or by directly differentiating the machine learning-informed optimization model, using derivative-based algorithms [10], [11]. The latter strategy, as the one employed in this paper, offers the flexibility of integrating any NN architecture, as an exact mixed-integer reformulation does not exist for all NN models. A second-order Sequential Least Squares Programming (SLSQP) method [22] is employed to solve the MPNNC. Despite the more robust performance, second-order methods are seldom used in the *training* of NNs due to a large number of decision variables. The MPNNC, however, has a much-reduced number of decision variables (compared to the NN's training problem), allowing for the efficient application of second-order methods.

The main contributions of this paper are:

1) The learning capability of a neural network is used to derive a surrogate representation of the (imperfectly known) LL market clearing problem in the bi-level optimal bidding model, leading to the novel MPNNC formulation that, opposed to classical MPEC models, can capture the partially known parameters and non-convex characteristics of electricity markets. Due to the explicit integration of the strategic agent's optimization problem in the MPNNC, interpretability and data efficiency are improved compared to model-free solution strategies.

2) The optimal bidding problem of a demand agent with load-shifting capabilities is introduced, illustrating that the MPNNC accommodates both box and inter-temporal constraints, which is then solved by a second-order optimization technique.

3) First, a stylized case study illustrates that the MPEC and the MPNNC are within 18% in terms of aggregated savings in the idealized environment (i.e., a convex lower-level problem that captures price formation with known parameters). Second, to benchmark the proposed MPNNC model in a non-convex environment, the MPEC formulation of [5] is implemented that has a unit commitment problem in the LL with non-convex techno-economic constraints. Lastly, the model's performance is further validated on the replication of the Belgian spot exchange, for which MPECs can not be realistically employed. This validation step intends to bridge the gap between theoretical and operable strategic bidding models.

While existing works in the field of power systems and electricity market research aim to reformulate the various ML models as (i) either a mixed-integer constraint set [23], (ii) or as a set of disjunctive constraints (resulting in a non-convex non-linear programming formulation), the proposed MPNNC: (i) encodes the neural network as a single non-linear constraint, $y = NN(x)$, and (ii) utilizes the fully differentiable structure of the NN to apply automatic differentiation w.r.t. the model's input (x), to optimize its output (y) in the implemented SLSQP optimization method. The benefit of this approach lies in the flexibility of incorporating various state-of-the-art NN architectures, for which, the exact mixed-integer reformulation is not yet developed in the literature.

The proposed MPNNC structure allows characterizing most UL problem classes relevant in electricity markets, e.g., the optimal bid problem of an asset owner or the welfare maximization problem of a social planner. To the best of the authors' knowledge, this is the first strategic bidding model that captures the price formation process in real-life market environments in the LL. As such, the presented methodology may serve two purposes, (i) it may be utilized to find the optimized market stimulus to achieve various upper-level objectives, (ii) it can facilitate the recognition of a strategic agent's bidding behavior on various markets.

This paper continues as follows. Section II introduces the modeled bi-level problem, and the implemented optimization techniques to tackle both convex and non-convex LL problems. Then, in Section III, the proposed MPNNC model is introduced. Section IV presents the benchmarking of the MPNNC model in two stylized market environments (a convex and a non-convex one). Section IV investigates the performance of the proposed model in the realistic replication of the Belgian power exchange. Lastly, Section VI derives the conclusions and potential outlook.

II. The classical bi-level programming approach

Let us assume that the electricity market clearing, formulated as the maximization of the social welfare, subject to techno-economic constraints, characterizes (i) the mapping between its inputs x (e.g., the participants' quantity bids: $\{\overline{G_i} \cdot AF_{i,t}, D_t\}$, and price bids: c_i^{gen}) and the resulting dispatch ($y^{DP} = g_{i,t}$) of the generation $f^{ED} : x \mapsto y^{DP}$, (ii) the mapping between the resulting dispatch and corresponding market equilibrium price (λ), which is defined by the merit order curve $f^{MO} : y^{DP} \mapsto \lambda$. After connecting the two functions, the relationship between the price formation of the electricity market (EM) and its inputs may be captured as: $f^{EM} : x \mapsto \lambda$. This mapping is represented in most models via an economic dispatch (ED) or a unit commitment (UC) problem, as presented in the stylized examples of Section IV-A3 and Section IV-B2. This paper, however, proposes a neural network NN(x, θ) as a surrogate LL model, which may better capture the underlying dynamics by exploiting past real-life market realizations.

The effectiveness of the proposed MPNNC model is illustrated for a load agent that has the potential to shed a small fragment of the Belgian system's load. Therefore, the decision is to determine how to distribute the available load shedding potential (LS) over 24 hours.

In the bi-level setting, a hierarchy exists between the market participants' decision problem and the market clearing problem, i.e., the decision vector of the UL agent D_t, parametrizes the decision space $y \in Y$ of the LL problem. Below the optimal bidding problem of the strategic agent is formulated.

A. Upper-level problem: Cost minimization of the load agent

The objective function (1a) represents the electricity cost summed over the optimization horizon of the load agent for procuring its desired amount of electricity, which is the market price λ_t multiplied by the requested electricity demand D_t:

$$\text{UL} : \min_{D_t} \mathcal{C}(D_t, \lambda_t(x)) = \lambda_t(x) \cdot D_t \tag{1a}$$

subject to

$$0 \leq D_t \leq \overline{D_t} \tag{1b}$$

$$\sum_{t \in T} (\overline{D_t} - D_t) \leq LS \tag{1c}$$

Note that the market price depends on the inputs (x), including the decided quantity bids (D_t) of the UL agent, which is why the bidding strategy of the load agent and the market clearing are inherently inter-related. Equation (1b) is a box constraint that enforces that the demand bid does not exceed a predefined maximum ($\overline{D_t}$) or takes negative values. Equation (1c) is the inter-temporal constraint that enforces a maximum budget on the aggregated load shedding over the optimization horizon (here over the day).

B. The ED problem describing the stylized LL

Although the NN used in the LL part of the MPNNC model has the capability to capture the complex, often nonlinear and non-convex dynamics of the real market environment [24], in the first numerical analysis (Section IV), a stylized economic dispatch (ED) model (Problem (2) below) is used to simulate the market environment. This allows an ideal benchmark, namely the classical bi-level model (MPEC), which can be solved to global optimality for the stylized environment. This ideal benchmark can then be used to assess the performance of the presented data-driven MPNNC model.

Objective function (2a) includes the generation cost along with emission costs of operating the system and the social cost of the curtailed load, i.e., the energy that is not served:

$$\begin{aligned} \text{ED} : \min_{g_{i,t}, ens_t} \mathcal{C}(g_{i,t}, ens_t) = & \qquad (2a) \\ \sum_{i \in \mathcal{I}^D} \sum_{t \in \mathcal{T}} (c_i^{gen} \cdot g_{i,t} + p^{CO_2} \cdot em_i \cdot g_{i,t}) & \\ + \sum_{t \in \mathcal{T}} ens_t \cdot VOLL & \end{aligned}$$

subject to

$$\sum_{i \in \mathcal{I}} g_{i,t} + ens_t = D_t \quad (\lambda_t) \quad \forall t \in \mathcal{T} \tag{2b}$$

$$0 \leq g_{i,t} \leq AF_{i,t} \cdot \overline{G}_i \quad \forall i \in \mathcal{I}, \forall t \in \mathcal{T} \tag{2c}$$

$$0 \leq ens_t \leq D_t \quad \forall t \in \mathcal{T} \tag{2d}$$

Constraint (2b) enforces the energy balance for each time step t, while Eq. (2c) represents the technical limits of the generation units, and Constraint (2d) imposes bounds on the energy not served ens_t. λ_t shown on the RHS of Eq. (2b) is the corresponding dual variable that, by definition, is the market price resulting from the ED model [25].

Problem (2) represents an electricity market clearing in its simplest form. Firstly, the solution space of the above ED problem, defined by Eq. (2b-2c) is fully convex, while in reality, due to the technical constraints (e.g., minimum generation levels and up/down times in US markets) and economic considerations (e.g., complex cost structures or constraints on the surplus by each accepted bid in the European day-ahead market), it is non-convex.

C. Extending the LL with unit commitment decisions

We incorporate binary unit commitment decisions $z_{i,t}$, start-up $v_{i,t}$ and shut-down $w_{i,t}$ decisions, as well as the corresponding ramping limits (3f - 3g) and minimum up- and down-times (3i-3h) in Problem (2). The extended objective minimizes the aggregation of (i) the fuel cost (3b), (ii) the emission cost (3c), (iii) the start-up cost (3d), and (iv) the cost of energy not served:

$$\text{UC}: \min_{g_{i,t}, ens_t} \mathcal{C}(g_{i,t}, ens_t) = \tag{3a}$$

$$\sum_{i\in\mathcal{I}, t\in\mathcal{T}} \left(fcd_{i,t} + ccd_{i,t} + scu_{i,t}\right) + \sum_{t\in\mathcal{T}} VOLL \cdot ens_t$$

subject to

$$fcd_{i,t} =$$

$$\alpha_i \cdot z_{i,t} + \beta_i \cdot (g_{i,t} - z_{i,t} \cdot \underline{G_i}) \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3b}$$

$$ccd_{i,t} =$$

$$p^{CO_2} \cdot \left(\gamma_i \cdot z_{i,t} + em_i \cdot (g_{i,t} - z_{i,t} \cdot \underline{G_i})\right) \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3c}$$

$$scu_{i,t} = STC_i \cdot v_{i,t} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3d}$$

$$\underline{G_i} \cdot z_{i,t} \leq g_{i,t} \leq \overline{G_i} \cdot z_{i,t} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3e}$$

$$g_{i,t} - g_{i,j-1} \leq \overline{\Delta G_i^{\uparrow}} \cdot z_{i,t} +$$

$$+ (\underline{G_i} - \overline{\Delta G_i^{\uparrow}}) \cdot v_{i,t} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3f}$$

$$g_{i,j-1} - g_{i,t} \leq \overline{\Delta G_i^{\downarrow}} \cdot z_{i,t} + \underline{G_i} \cdot w_{i,t} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3g}$$

$$z_{i,t} \geq \sum_{t'=t+1-MUT_i}^{t} v_{i,t'} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3h}$$

$$1 - z_{i,t} \geq \sum_{t'=t+1-MDT_i}^{t} w_{i,t'} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3i}$$

$$z_{i,j-1} - z_{i,t} + v_{i,t} - w_{i,t} = 0 \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3j}$$

$$z_{i,t},\ v_{i,t},\ w_{i,t} \in \{0,1\} \quad \forall i \in \mathcal{I}^D, t \in \mathcal{T} \tag{3k}$$

$$\sum_{i\in\mathcal{I}} g_{i,t} + ens_t = D_t \quad (\lambda_t) \quad \forall t \in \mathcal{T} \tag{3l}$$

$$0 \leq g_{i,t} \leq AF_{i,t} \cdot \overline{G}_i \quad \forall i \in \mathcal{I}^V, \forall t \in \mathcal{T}$$

For sake of brevity, we do not further discuss this standard UC problem here but refer the interested reader to [26].

D. Bi-level formulation of the optimal bidding problem assuming a convex LL

In order to maximize its own objective, the UL agent recognizes the dependency between its actions and the LL market clearing's outcome $f^{EM} : x \mapsto \lambda$. This hierarchical relation is a typical instance of a Stackelberg Game. The leader anticipates the reaction of the follower $y(x), \lambda(x)$ to its decision x in order to maximize its utility or minimize its cost $\mathcal{C}(x, y)$. In the context of optimal load shedding, this game may be formulated as:

$$\text{MPEC}: \min_{D_t} \mathcal{C}(D_t, \lambda_t(x)) = \lambda_t(x) \cdot D_t \tag{4a}$$

subject to

$$0 \leq D_t \leq \overline{D_t} \tag{4b}$$

$$\sum_{t\in\mathcal{T}} (\overline{D_t} - D_t) \leq \sum_{t\in T} \overline{LS_t} \tag{4c}$$

$$\lambda_t(x) = \arg\max_{y} \left\{ f\left(x, y\right) : y \in \mathcal{Y}(x) \right\} \tag{4d}$$

The difference of the above formulation from the one in Section II-A is that the LL's response is explicitly captured as the optimum of Eq. (4d), in which, $f\left(x, y\right)$ refers to the objective function (2a), and $\mathcal{Y}(x)$ to the feasible set defined by Eq. (2b)-(2c).

Standard practice in the literature is to model and solve such games as Mathematical Programs with Equilibrium Constraints (MPECs). The solution procedure's first step is to transform the bi-level optimization problem into the single-level form (as indicated in Fig 1), by reformulating the LL (the market clearing). This reformulation may be done, e.g., by using the Karush-Kuhn-Tucker (KKT) conditions, or by the primal-dual reformulation combined with strong duality [2]. The KKT conditions of a convex optimization problem subject to equality ($h(y) = 0$) and inequality constraints ($g(y) \leq 0$) read:

$$\Delta_y \mathcal{L}(y, \mu, \gamma) = 0 \tag{5a}$$

$$\gamma \cdot g(y) = 0 \tag{5b}$$

$$\gamma \geq 0 \tag{5c}$$

$$g(y) \leq 0 \tag{5d}$$

$$h(y) = 0 \tag{5e}$$

Eq. (5a) denotes the Lagrangian stationarity condition, while Eq. (5b) the complementary slackness conditions. Eq. (5c) enforces dual feasibility and Eq. (5d)-(5e) the primal feasibility of the original optimization problem. When all equations of Problem (5) hold, and that the Hessian of the Lagrangian ($\Delta_{yy}\mathcal{L}(y, \mu, \gamma)$) is positive definite, the KKT conditions are necessary and sufficient for optimality.

For a convex economic dispatch model as Problem (2), the KKT conditions are necessary and sufficient for optimality. However, after including the non-convex constraints of the participating generators (Problem (3)), or modeling agents' possibly imperfect bidding behavior as required to accurately capture real-life markets, the KKT conditions would often become neither necessary nor sufficient. Assuming that one can find meaningful KKT conditions for the underlying LL problem, it still may be numerically difficult to solve the MPEC (single-level form) to global optimality, because the complementary slackness conditions and the objective function consists of nonlinear and non-convex terms. To approximate the bi-linear terms stemming from the complementarity slackness conditions, a mixed integer reformulation that is often referred to as big-M method [27] is used. The resulting problem is an exact reformulation of the original problem if the M parameters are appropriately chosen. However, this is a non-trivial task [28], [29].

In this paper, the nonlinearities in the objective function are not further linearized as in [2], thanks to the recent advances of Gurobi's non-convex solver [30], which can effectively handle the resulting MINLP, while guaranteeing global optimality of the solution. On the other hand, if the complementarity slackness conditions are also kept in the bi-linear format, i.e., not reformulated via the big-M method, the problem cannot be solved in a reasonable time.

E. Bi-level solution method assuming a non-convex LL

Problem (3)'s non-convexity implies that traditional MPEC solution techniques are not applicable as one cannot attain meaningful KKT conditions to reformulate the LL problem. Moreover, the dual variable of the market clearing constraint in Problem (3) is not available. To overcome these limitations, i.e., formulate an MPEC model with non-convexities represented in the LL, Ye et al. [5] propose a primal-dual reformulation-based solution strategy (denoted as MPEC* in the following), inspired by [31] and summarized below:

1) The binary decision variables of Equation (3k) are first relaxed, i.e., $0 \leq z_{i,t}, v_{i,t}, w_{i,t} \leq 1$, leading to the relaxed primal UCM form (UCM_{rel}^{primal}).

2) Next, the equivalent dual formulation of UCM_{rel}^{primal} is attained (UCM_{rel}^{dual}). Following the primal-dual optimality conditions for the LL reformulation [2], the feasibility set formed by the constraints of the dual formulation (UCM_{rel}^{dual}) and the *original* primal formulation (Problem (3)) replace the LL of the MPEC* model.

3) As strong duality does not hold, the primal and dual objective values may not be equal at the optimum of the MPEC* due to the additional binary variables of the primal problem. The duality gap, defined here as the difference between the optimal objective of the original, primal Problem (3) and the dual associated with its relaxed verison UCM_{rel}^{dual}, is penalized in the UL objective.

4) Lastly, the strategic agent's objective function and constraints are included in UL of the MPEC*.

The resulting objective function is composed of two terms (i) the duality gap and (ii) the strategic agent's objective. Therefore, a weighting factor (W) is introduced (i.e., UL objective = strategic agent's objective + W · gap). As noted by Ye et al. [5], the optimal choice of parameter W in the above formulation is a challenging task. Furthermore, every change in the inputs of the LL, including the UL decisions, may trigger a new optimal choice for parameter W. Consequently, the optimal range of W is not transferable from one study to another.

III. Surrogate Neural Network Capturing the Market Dynamics

This section introduces the modeling framework that exploits a surrogate NN model to capture the market dynamics inside a bi-level setting, i.e., the optimal bid problem of a strategic demand agent.

Let α_i denote the non-linear activation function that enables capturing complex dependencies for a NN with $i = 1, ..., l$ number of layers. Then, z_i represents the mapping between the previous layer (z_{i-1}) and the inputs (x) via the corresponding weights (W_i^z, W_i^x) and bias term (b_i) in the following way:

$$z_{i+1} = \alpha_i\big(W_i^z \cdot z_i + W_i^x \cdot x + b_i\big) \tag{6}$$

where $W_1^z, z_1 = 0$, as the first layer is not connected to any previous hidden layer. The output of the NN(x, θ) is the result of the last layer (z_l). While various ML models may be used to approximate the electricity market dynamics [32], NNs have the known capability to capture arbitrary nonconvex functions on a compact domain. The obvious difficulty is to choose the correct hyper-parameter setting of the NN [33]. Neural networks have consistently shown superior performance in forecasting the market prices, i.e., capturing the market's behavior [24], showcasing their efficiency in capturing the non-convex operational constraints and economic objectives of the electricity market actors. On the contrary, one has to keep in mind that the use of a NN introduces larger complexity in the training phase due to the larger number of tune-able hyperparameters.

A. Training of the neural network

In the stylized case study (Section IV), a chained NN architecture is used to achieve a close match with the explicit LL representation of the MPEC. The first NN captures the dispatch of the participating generation units, while the second one forecasts the market price given the predicted generation dispatch. The neural networks are denoted as $NN(x, \theta)$, where θ refers to the weights W and biases b parameters, and x to the vector of input features. The optimal weights and biases are obtained in the ex-ante training that minimizes the mean-squared-error (MSE) loss function between the prediction $\hat{y}$ and the actual observation y: $\theta^* = \arg\min_y ||\hat{y} - y||_2^2$.

The training is split into two parts, (i) fitting the dispatch forecast $y^1 = NN^1(\theta^1, x)$, (ii) fitting the price forecast given the results of the forecasted dispatch $\lambda = NN^2(\theta^2, y^1)$. The resulting chained model forecasts the prices (the output of the lower-level model) in the following way: $\theta^* = \arg\min_\lambda ||\big(\arg\min_{y^{DP}} ||\hat{y}^{DP} - y^{DP}||_2^2\big) - \hat{\lambda}||_2^2$. By decoupling the forecaster into two sub-models (dispatch and price formation), the electricity market mechanisms are better reflected, and thus the behavior of the forecaster $NN(x, \theta)$ can be better interpreted. Moreover, the intermediate dispatch variables can be constrained by embedding any techno-economic constraint (e.g., generators' output limits) into the first NN, in a similar fashion as in [12], to enhance both computational performance and accuracy of the NN in capturing Problem (2).

This chained NN structure is, however, not strictly required in the MPNNC approach. We illustrate this for the real market data in the second case study (Section V), the intermediate step to forecast the generators' dispatch profiles, $y^1 = NN^1(\theta^1, x)$, is not exploited. However, it should be noted that our innovative implementation of the MPNNC with a chained NN structure may offer benefits for other real market applications.

B. NN-based formulation of the optimal bidding problem

In Fig. 2, we show how the pretrained surrogate NN is used in the demand agent's optimal bid problem to decide on the optimal load shedding strategy. Due to the backpropagation step of the training procedure, NNs are formulated as differentiable programs. Therefore, their inclusion in existing optimization algorithms, exploiting derivative information, can be done effectively. The MPNNC formulation contains the cost minimization problem of the UL agent constrained by the $NN(x, \theta)$ replacing the LL that captures the mapping between input variables ($D_t \subseteq x$), and the resulting price (λ_t):

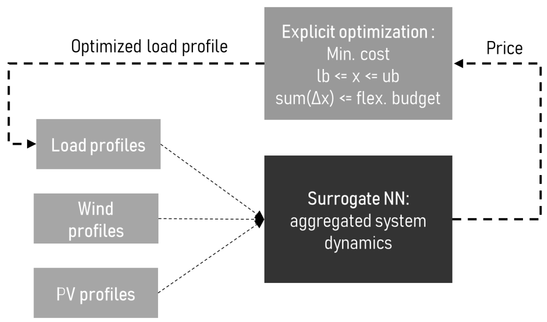

Fig. 2: Scheme showing the generic setting for the NN-based optimization.

$$\text{MPNNC} : \min_{D_t} \mathcal{C}(D_t, \lambda_t(x)) = \mathcal{C}(D_t, \lambda_t(x)) \tag{7a}$$

subject to

$$\mathcal{C}(D_t, \lambda_t(x)) = \lambda_t(x) \cdot D_t \tag{7b}$$

$$0 \leq D_t \leq \overline{D_t} \tag{7c}$$

$$\sum_{t \in \mathcal{T}} (\overline{D_t} - D_t) \leq \sum_{t \in \mathcal{T}} \overline{LS_t} \tag{7d}$$

$$\lambda_t(x) = NN(x, \theta) \tag{7e}$$

Compared to the training step (detailed in Section IV), in the MPNNC problem the weights (θ) are fixed in $NN(x, \theta)$, whereas a subset of input variables, i.e., the chosen demand bid is optimized. Neural Networks using the ReLU activation functions are non-convex from the input to the output, rendering Problem (7) a nonconvex optimization problem. Given that the number of variables (x) is significantly reduced when solving the MPNNC compared to the NN training stage when the weights θ are optimized, second-order, e.g., Newton or quasi-Newton, methods can be efficiently applied. They capture curvature information that can lead to quick convergence and high robustness [34]. Moreover, second-order methods reduce the number of manually tuned hyperparameters [35].

IV. MPNNC and MPEC model comparison in a stylized environment

The proposed MPNNC model is benchmarked against the MPEC model. A stylized problem setting is used, in which we first assume that the convex economic dispatch problem exactly captures the market environment (Section IV-A). Hence, the optimality conditions of the LL problem are an exact representation of the environment, which enables the comparison of the MPNNC's performance to a theoretical upper bound. Second, we illustrate how the MPNNC allows capturing non-convex LL problems (Section IV-B).

The training and test results are generated by altering the inputs of the ED (Section IV-A) or UC problem (Section IV-B). The CO_2 price (p^{CO_2}) is set to 25 €/ton , while the Value of Lost Load (VoLL) is 1000 €/MWh. The load profile and the availability factors of the wind and solar generation on the considered day are plotted in Fig 3. We create 500 different load, wind, and solar profiles by multiplying each time step of the original profiles by a coefficient sampled from a normal distribution with a mean of 1.0 and a standard deviation of 0.05. Considering these 500 combinations of load, wind and solar and the generators' characteristics summarized in Table I, 500 altered dispatch profiles and market price profiles are generated by solving the ED (2) or UC problem (3). The first 400 are used for training the NN and the last 100 as a test set, which allows comparing the performance of the MPEC and MPNNC.

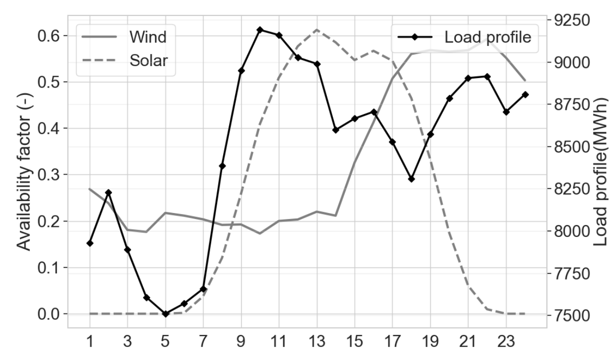

Fig. 3: The availability factor of the renewable energy sources, i.e., the wind and solar generation (left-axis), and the aggregated demand in the system (right-axis) throughout an example day.

A. Convex economic dispatch problem as LL

As discussed in Section III-A, since the electricity market price formation may be seen as a composition of two subproblems, the proposed NN model contains two NNs, which are chained into a single NN after the training procedure.

The NN weights are optimized through a mini-batch gradient descent algorithm using the Adam optimizer [36] with the goal of obtaining the lowest mean squared error (MSE) w.r.t. the output of the ED model. Each day is handled as an independent batch of 24 hourly steps, leading to 400 batches of size 24. The maximum number of epochs is chosen as 50, whereas the learning rate is 0.001. Both NNs in the chained structure (see Fig 2) are trained under the same setting. Each NN is composed of three layers, from which the first one is a dense layer and the other two others are custom-made input convex layers [37]. All hidden layers have 64 neurons. The NN models are implemented in the Julia programming language [38] using the Flux package [39].

TABLE I: Generator data, used as LL as inputs for the convex and non-convex "ground truth" models.

	number of units	c_i^{gen}	em_i	G_{min}	G_{max}	STC_i	MDT_i	MUT_i	$\overline{\Delta G_i^{\uparrow}}, \overline{\Delta G_i^{\downarrow}}$	α_i	β_i	γ_i
CCGT	4	36.36	0.38	0	450	17100	1	2	1620	8139.53	29.9	85.5
Nuclear	3	9.09	0	0	1200	14040	8	8	1800	10000	4.6	0
Coal	2	25.0	0.85	0	800	111200	3	6	960	13513.5	21.6	459.5
OCGT	5	50.0	0.53	0	50	0	1	1	300	1153.85	38.5	12.1

1) Learning the economic dispatch model: The first NN is trained to best describe the mapping between input time series and the dispatch profile of the 16 generators, including 14 conventional units (summarized in Table I) and 2 renewable generators (see Fig 3). The forecasted dispatch for the 100th test day is shown in Fig 4. In addition, the load profile is plotted by the dotted black line that needs to be satisfied in order to maintain system balance. While the error for the individual generator's predicted dispatch may differ, especially for the ones with smaller capacity as their contribution to the overall MSE is smaller, it is visible that the aggregated predicted generation follows closely the load profile.

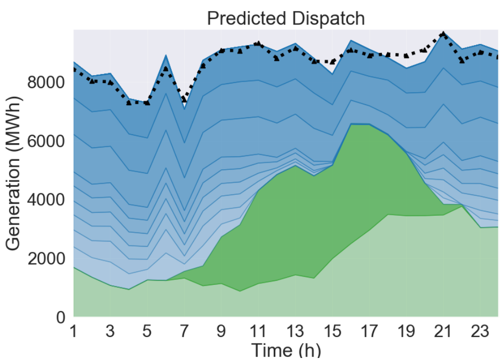

Fig. 4: Stacked plot of a single day's predicted dispatch for all conventional generators (blue) and for all renewable generators (green). The aggregated load is indicated by the black dotted line.

2) Learning the price formation process: The resulting market prices are predicted from the dispatch prediction generated by the first NN. After training the second network, it is chained with the first one to capture the relation between the input series (load, RES generation), and the resulting prices (Fig 2). While achieving the best forecasting performance is not the main focus of this paper, a certain accuracy is desired for a well-performing MPNNC model as the dynamics of the underlying price formation are based on the trained NN.

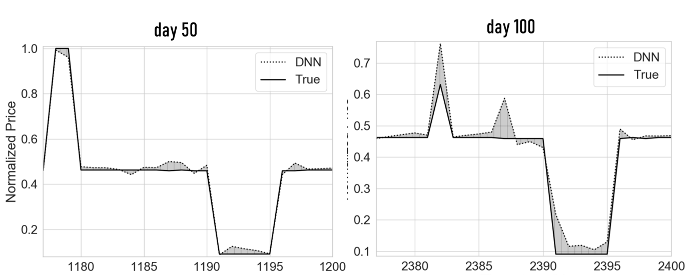

Fig. 5: The scaled price forecast of the 50th and 100th day of the test set. The scaling was done by dividing all values by the maximum observed price (the VoLL).

Figure 5 shows that the price profile of the 50th test day is very closely captured. Both the peak and the off-peak prices are well-represented by the prediction, suggesting that price fluctuations are efficiently captured by the trained NN. On the contrary, the predictions made for the 100th test day show larger errors, by overestimating the first peak and creating another price spike that did not occur. Overall, the average daily RMSE was 0.048 €/MWh with a standard deviation of 0.020 €/MWh, while day 50 had a RMSE of 0.02 €/MWh, and day 100 had a RMSE of 0.048 €/MWh.

3) Optimal bidding in case of convex LL: In this section, the results of the proposed MPNNC model are investigated in a stylized environment through the lens of a strategic demand agent's optimal bidding strategy (as presented in Section II and Section III). The maximum load shedding over the day is limited to 5% of the total load: $\sum_{t\in\mathcal{T}}(\overline{D_t} - D_t) \leq 0.05 \cdot \sum_{t\in\mathcal{T}} \overline{D}_t$. The MPNNC results are benchmarked against the *naive* price-taker (PT) and against the MPEC outcomes. The former strategy involves first forecasting the price profile, and then subsequently shedding the load at the most expensive time-step(s). The latter is an ideal case for the fully continuous and convex ED environment, where the agent has access to all necessary information. The MPEC's lower-level problem representation is thus exact, and the KKT conditions are necessary and sufficient for optimality. The MINLP is solved by the Gurobi solver [30], using the non-convex optimization method that guarantees global optimality. However, due to the optimistic LL response assumed in the MPEC, the MINLP model still may not capture the best bidding strategy. The MPNNC model is solved using the optimizer of the *scypy* library [40], selecting the SLSQP algorithm, introduced in [22]. The tolerance in the MPNNC models is set to the optimality gap (*MIPGap*) of Gurobi (0.0001). To inter-operate with the *scypy* Python library, the PyCall julia package is used.

The final price profiles are obtained by integrating the load's quantity bids (i.e., decisions of the optimization) in Problem (2). This process eliminates the inherent biases of the models, e.g., the optimistic LL response in the MPEC model, or the NN's approximation error in the MPNNC model.

Table II summarizes the median and maximum solution times to solve the MPNNC and MPEC models and the mean daily cost attained by the three models (MPNNC, MPEC, PT) over the 100 test days. The costs of procuring the demand are substantially reduced by both MPEC and MPNNCS models compared to the PT case, showing their efficiency. The MPNNC reduced the daily procurement cost by 18.5%, while the MPEC achieved a 24.6% reduction, leading to a 6.1 percentage point difference between the theoretically ideal (MPEC) and proposed MPNNC model. It should be emphasized that the MPEC can achieve this performance only in the

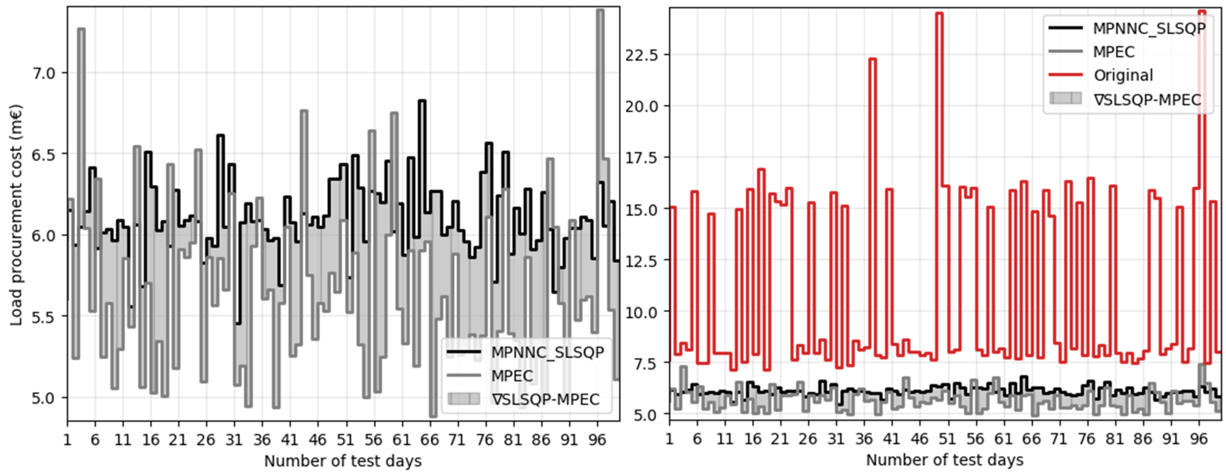

Fig. 6: Comparing the daily cost reductions of both models (MPEC, MPNNC) to the original costs (right figure). For better illustration, the left figure, only depicts the results of the two optimized costs.

stylized setting. In addition, the MPNNC leads to significantly lower simulation times, suggesting better scalability.

TABLE II: The median and maximum solution times and mean daily procurement costs of the MPNNC, MPEC and PT models. The original average cost of load procurement, without load shedding, is 1.09e7 €.

	MPNNC	MPEC	PT
Median Time (s)	0.53	8.2	-
Max Time (s)	1.3	50.1	-
Avg. Cost (M€)	6.09	5.64	7.48

Figure 6 shows the evolution of the daily costs over the test set (100 days). The right-hand side shows the optimized costs (by both MPNNC and MPEC models) in comparison with the original outcomes. The improvement attained by both models is significant and their differences are not substantial compared to the overall decrease of the procurement cost. Despite the improved solutions of the MPEC model, the MPNNC attains lower costs 15% of the time. As shown below, this may be due to the LL's optimistic response in the MPEC model [41] that translates to a great discrepancy between the estimated market price (by the MPEC) and the one produced by Problem (2).[1] The best performance of the MPNNC model is achieved on day 4, with an improvement of 1.22 M€ w.r.t. the MPEC, and the worst outcome occurred on day 67, with total costs 1.39 M€ higher than the MPEC.

For these extreme days, the optimized bids and resulting market price profiles are plotted in Fig 7. In day 4, when the MPNNC model has the best performance, it is visible that the MPNNC model deviates more significantly from the original profile in a limited number of time steps, while the MPEC deviates significantly in multiple time steps, e.g., between 8:00 and 11:00. The resulting price profiles (bottom of Fig. 7) show that the MPNNC model reduces the price to below 10 € per MWh from 45 € per MWh for a longer period, between 11:00 and 20:00, while the MPEC achieves the same reduction in

[1]Furthermore, the choice of the 'M' parameter in the reformulation of the complementary slackness constraint impacts the exactness of the LL. Optimally choosing this parameter, however, is an NP-hard task.

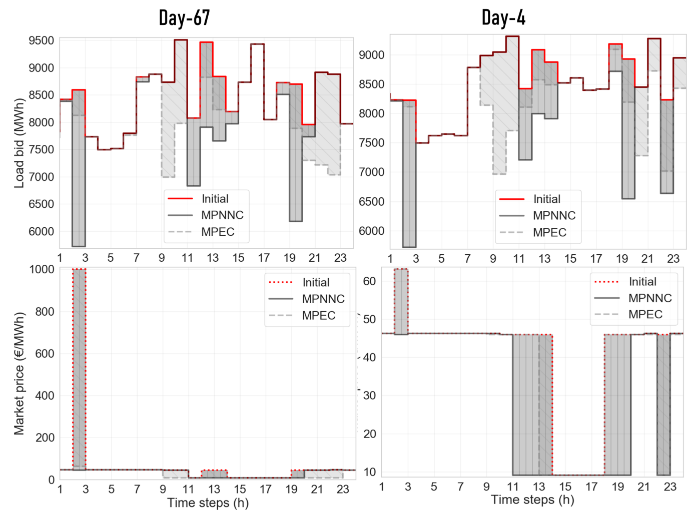

Fig. 7: The best (right two sub-figures) and worst performing days of the MPNNC model compared to the MPEC's performance.

a shorter time window (13:00-20:00). The main cause of the MPEC's inferior performance is that in many steps the large load deviation does not result in a stronger price reduction. As the stylized case study involves a low number of market participants, the offer curve exhibits pronounced discrete steps. Therefore, the MPEC's inherent optimistic response assumption may lead to amplified differences between the assumed and actual response of the market.

Looking at the opposite case (day 67), in which the MPNNC achieved the worst performance, it can be noted that prices are no longer equal to the price cap (1000 €/MWh) in both results. In the second half of the day, however, the MPEC model manages to keep the prices low for a longer time horizon (9:00-23:00), resulting in superior performance. The fact that the MPNNC model did not manage to exploit the same price reduction opportunity in the second half of the day is due to utilizing more flexibility than needed in other time steps, inheriting from the NN's imperfect approximation of the market.

B. Non-convex unit commitment problem as LL

Generating the required training and test data when the market clearing problem is represented as a non-convex unit commitment problem (3) is not straightforward, as dual variables, thus market prices, are not readily available. To circumvent this issue, we leverage the MPEC* formulation (Section II-E), from which we remove the UL objective to attain market prices for the 500 problem instances as above.

1) Learning the price formation process: The average daily RMSE, in case of the non-convex "ground-truth" model, changed to 0.094 (€/MWh) with a standard deviation of 0.027 (€/MWh), which is around twice as much as in the convex "ground-truth" model case. Figure 8 shows the forecasted and actual daily prices for the same days (50, 100) as in the previous section. Contrary to the previous results, it can be observed that the prediction of day 50 does not capture well the peaks and valleys of the price profile, indicated by the higher than average RMSE 0.129(€/MWh). For day 100, on the other hand, the prediction closely follows the actual profile, leading to an RMSE of 0.082 (€/MWh).

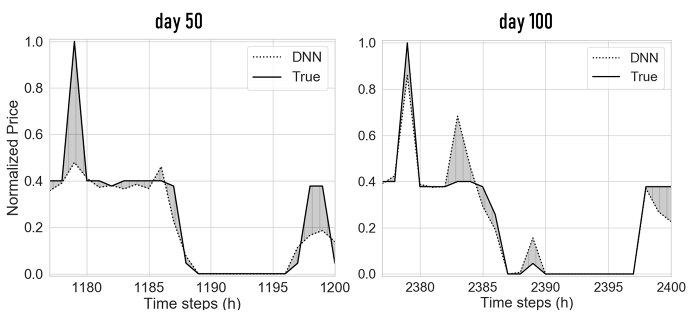

Fig. 8: The scaled price forecast of the 50th and 100th day of the test set, in case of non-convex "ground-truth" model. The scaling was done by dividing all values by the maximum observed price (the VoLL).

2) Optimal bidding in case of non-convex LL: We compare the optimal strategic bids attained by the proposed MPNNC model and the MPEC* alternative. To evaluate the quality of the attained bids, we enforce these bids in the MPEC* formulation, from which the strategic agent's objective function and constraints are removed. Note that this approach may lead to overestimating the performance of the MPEC*.

As highlighted in Section II-E and in [5], choosing the 'W' parameter of the MPEC* model is not trivial and may have a significant impact on the ex-post reward of the strategic agent. Therefore, in this case study, the outcomes of two different strategies are presented. The first one uses the value suggested in [5] (W=1000), while the second one runs the optimization with three different values (W $\in$ [100, 500, 1000]), and retains the outcome that produces the best ex-post reward for the load shedding agent. Note that in real-life applications, one would have to determine this 'W' parameter ex-ante, challenging the use of this approach.

Table III summarizes the load procurement cost (in the ex-post evaluation) of the strategic load shedding agent if using the proposed MPNNC model or the MPEC* with the two above-described strategies. The MPEC^{best} model performs 6% better than the MPNNC. However, the MPNNC outperforms the MPEC^{1000} model by 4%, indicating the MPEC*'s sensitivity w.r.t. the 'W' parameter.

TABLE III: The mean daily procurement costs of the MPNNC, and the MPEC in case of a fixed weighting factor ($MPEC^{1000}$), and the best chosen from the list of [100, 500, 1000] ($MPEC^{best}$). The original average cost of load procurement, without load shedding, is 9.34 M€.

	MPNNC	**MPEC^{best}**	**MPEC^{1000}**
Avg. Cost (M€)	7.05	6.61	7.36

Figure 9 further illustrates the daily load procurement costs of the 100 simulated days. Similar to the case with a convex lower level, the $MPEC^{best}$ benchmark achieves lower load procurement cost than the MPNNC in most instances.

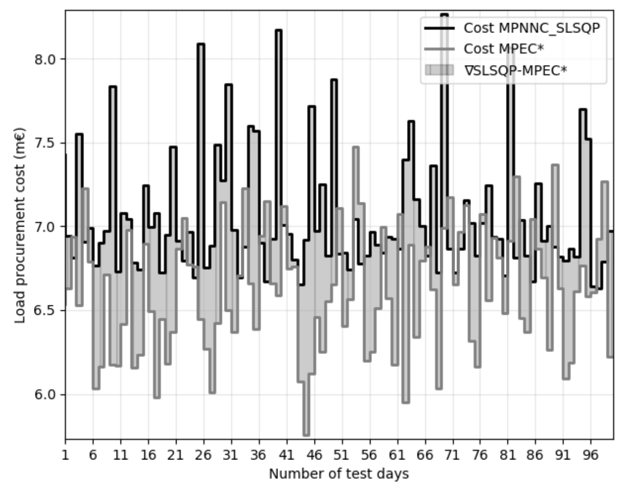

Fig. 9: Comparing the daily procurement costs of both models of the MPNNC and the MPEC^{best} models.

V. Strategic load shedding in the Belgian power exchange

In this section, the MPNNC model's bidding strategy is assessed on the Belgian day-ahead power exchange to prove the viability of the proposed approach in a real market environment. The results are contrasted to a price-taker strategy, in which the same trained NN is used to forecast the market prices, informing the optimal load shedding problem.

A. Experimental data and forecasting methodology

Similarly to Section IV, a feed-forward NN model is used to forecast day-ahead prices, with one hidden layer and 32 neurons. The hidden layer has leaky ReLU activation function, while the other two layers use ReLU functions. The following exogenous features are inputs to the NN model:

1) Belgian electricity load forecast (available from the website of ELIA [42] at the moment of bidding);
2) The last 12 and 24 hourly moving average values of the load;
3) The last 12 and 24 hourly moving average values of the price (the to-be-forecasted forecasted variable);

4) Temporal information, such as the: day-of-the-week, day-of-the-month, hour-of-the-day.

The model is trained on the period of 01/01/2019-14/10/2021. The learning rate is set to 0.0005, the number of epochs to 100, and the batch size to 50. The test period spans 15/10/2021 to 14/02/2022.

Figure 10 shows the observed and predicted prices for 10 days starting from 15/10/2021 (the beginning of the test period). It can be observed that the prices and their predictability vary significantly over the test period. This may be the consequence of the turbulent market environment of the modeled time period. Table IV summarizes the mean and median errors

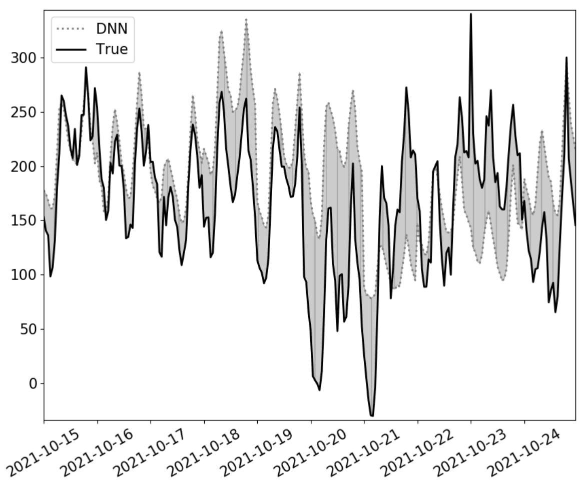

Fig. 10: Forecasted and actual prices of the Belgian spot exchange between 15/10/2021 and 24/10/2021. The grey filling is used to indicate the error.

of the predictions made during the test period, using the common root mean squared error (RMSE) and the symmetric mean absolute percentage error (sMAPE) advocated, e.g., in [32]. The mean and median sMAPE values are comparable (but slightly worse) to the performance reported in the elaborate study of Lago et al. [32]. Note that attaining a state-of-the-art forecasting performance is not the aim of this paper, yet a better forecaster can enhance the consequent bidding strategy. The MPNNC model optimizes the bidding strategy based on two criteria: (i) the predicted day-ahead prices (here, capturing the ratio of price peaks and valleys is crucial), (ii) the price elasticity, i.e., the predicted resilience of the price w.r.t. the optimized input variable (the load profile in our case). Note that assessing the former criterium is straightforward, while judging whether the forecaster accurately captures the resilience is more difficult.

TABLE IV: Mean and median calculated errors measured on the test period (15/10/202- 14/02/2022).

	sMAPE	RMSE
Mean error (%)	27.1	56.3
Median error (%)	24.4	49.8

B. Ex-post price approximation model

In order to reconstruct the actual (ex-post) price resulting from the altered actions of the UL agent, the *true* price elasticity of the wholesale market needs to be captured, e.g., by using the aggregated supply curves. The aggregated supply and demand curves of the Belgian power exchange are available to purchase in the European Power Exchange [43] (the product is named as *All-Certified Exchanges' Day-Ahead Auction aggregated curves*). They incorporate executed block volumes and net positions since 15/10/2021, leading to roughly 4 months available for testing. While this data allows constructing a realistic ex-post assessment tool, several discrepancies compared to the actual spot market remain present: (i) only the accepted block bids are incorporated in the aggregated curves, which prevents to fully capture the price resilience, (ii) similarly for the cross-border trades, net positions are fixed to their historically cleared values such that the dependence between net cross-border positions and domestic loads is neglected and (iii) at the intersection of the supply and demand curves there may exist multiple acceptable prices. Due to the first two limitations, the price responsiveness, captured by the aggregated curves, is limited to the Belgian market's sensitivity, which may be more sensitive to changes, i.e., less liquid, than the coupled European market. The multiplicity of prices at the supply and demand intersection may lead to differences between the constructed and true market clearing prices. For a complete representation, all connected countries and the market clearing and pricing algorithm of Euphemia [44] needs to be implemented, which is beyond the scope of this paper.

To construct the ex-post approximation model, two piecewise linear curves are fitted to the supply and demand bids using special order set (type-II) variables [45]. The intersection of the curves is calculated in a root-finding problem, which results in the cleared price-quantity pair. As Fig. 11 illustrates, the resulting price approximation closely follows the real prices on the presented day.

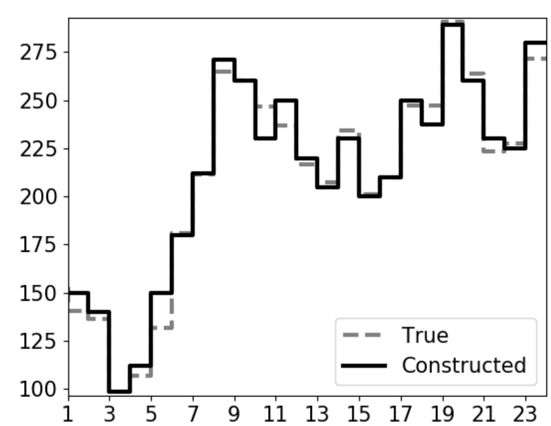

Fig. 11: Constructed and true price profiles at the beginning of the test horizon (15/10/2021).

C. Cost reductions via MPNNC-based bidding strategy

This section presents the attained cost reductions of the MPNNC model relative to a price-taker (PT) decision maker, which uses the forecasted price profile to shed the load at the most expensive hour. The daily cost of load procurement is calculated by the ex-post price approximation model (Section V-B). This model first calculates the original daily load profile[2] from the Belgian aggregated supply and demand curves. Then, the amount of load shed by both PT and MPNNC models is deducted from the original load. Lastly, the gain is defined as the difference in the daily load procurement cost between the PT and the MPNNC strategy and divided by the amount of load shedding. The resulting gain thus reflects the saving on the total load procurement cost per shed MWh (denoted as €/sMWh). For each day of the test period, the following load shedding capacities are tested: {1, 5, 10, 20, 30, 50, 100, 200} (MWh).

Table V summarizes the gains over the test period (*original* column). The median cost reduction (2417.2 €/sMWh) is complemented with a large standard deviation (25950.3 €/sMWh) suggesting extreme outlier values. Some extreme gains may arise from the large price sensitivity as the aggregated supply and demand curves ignore the available (not accepted) block bids and possible changes in cross-border net positions. Therefore, different levels of outlier gains are removed to decrease the influence of such distorting outcomes. In the subsequent columns of Table V, the top and bottom n-th (5th, 10th, 20th) percentiles of the results are eliminated and the corresponding gains are summarized. When the top and bottom 20th percentile is cut-off (illustrated in detail in Fig 12), the mean gain significantly reduces as well as its standard deviation. While the mean gain is much reduced compared to the original case, even such conservative estimation means that the same amount of load shedding leads to 125.3 € cost reduction per shed MWh. From the detailed results of Fig 12 it is also visible that the losses are smaller and they occur less frequently than the gains. From the spread of results, one can observe that the gains and losses both remain in the same range, confirming the lower variance. Also note that in all cases the median gain is 0, meaning that the MPNNC model selects the PT strategy more than 50% of the time.

TABLE V: Mean, median gains, and their standard deviations of the aggregated results (original column). The upper and lower 5th, 10th and 20th percentile of the results are eliminated, shown in columns 95-5, 90-10, 80-20.

	original	**95-5**	**90-10**	**80-20**
mean gain (€/sMWh)	2417.2	489.4	239.9	125.3
median gain (€/sMWh)	0.0	0.0	0.0	0.0
std. gain (€/sMWh)	25950.3	2749.9	1210.3	382.2

While the forecasting error on its own is not sufficient to assess the foreseeable model performance, the attained gains in the test cases are strongly correlated with the errors of the considered results. Fig 13 illustrates this relationship by showing the increase in main gains as a function of the maximum allowed forecast error level (i.e., by excluding results above a user-defined sMAPE limit). The results suggest that if the MPNNC model incorporates an improved forecaster, the attainable gains can increase, providing motivation for using more complex NN architectures in future research.

[2]The demand cleared on the spot market differs significantly from the overall demand in the country.

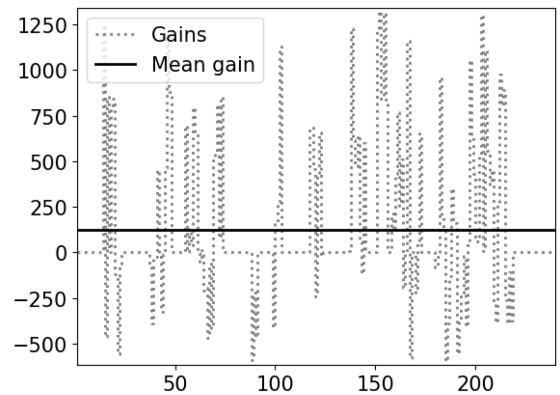

Fig. 12: The gains of the remaining results (y-axis) after removing the top and bottom 20th percentiles (80th, 20th) of outcomes.

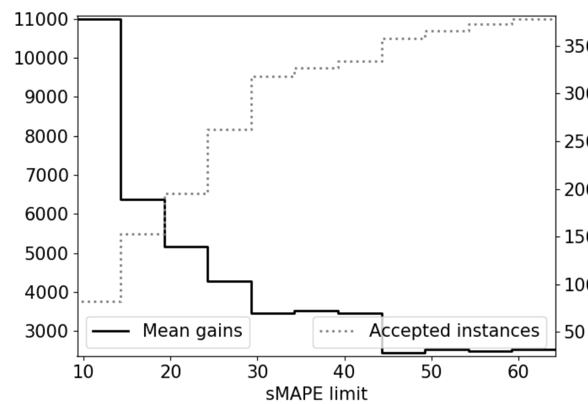

Fig. 13: Main gains (left y-axis) and accepted results (right y-axis) plotted against the maximum sMAPE limit allowed in the considered set of results.

VI. Conclusion

Existing model-based (MPECs) and model-free (RL formulations) approaches cannot sufficiently address the various challenges (model and parameter ambiguity, incorporating non-convexities) associated with modeling strategic market participation. In this paper, a learning-based MPNNC model was presented to formulate the optimal bidding problem of a strategic load agent. Compared to the classical MPEC formulation, the presented approach has the flexibility to capture real electricity market behavior in the lower level, while the upper level allows including various types of constraints and objective functions.

In a stylized, hypothetical market environment, the MPNNC model exhibits similar performances to the theoretical MPEC benchmark, and their difference is limited compared to the overall gains associated with behaving strategically. These observations hold also when the lower-level becomes non-convex, which challenges the application of state-of-the-art MPEC approaches. To prove efficiency in a real market setting, the MPNNC's performance is tested on the Belgian power exchange, for which a realistic ex-post assessment model is constructed that uses the aggregated supply curves of Belgium. Findings indicate the effectiveness of the MPNNC model and potential improvements (e.g., if the underlying NN has more accurate forecasting capabilities).

In future research, using alternative NN architectures to achieve state-of-the-art price forecasting results, holds significant potential to enhance the MPNNC performance. As an example, probabilistic neural network-based forecasting models (e.g., the temporal fusion transformer used in [46]) can be implemented to provide the inputs to a stochastic UL optimization problem, in forms of, e.g., scenarios or probability distributions.

Acknowledgment

The authors would like to thank the insightful comments of Ján Drgoňa in the conceptualization phase of this research.

References

[1] H. Von Stackelberg, *Market structure and equilibrium*. Springer Science & Business Media, 2010.

[2] S. A. Gabriel, A. J. Conejo, J. D. Fuller, B. F. Hobbs, and C. Ruiz, *Complementarity modeling in energy markets*. Springer Science & Business Media, 2012, vol. 180.

[3] A. J. Conejo, L. Baringo, S. J. Kazempour, and A. S. Siddiqui, "Investment in electricity generation and transmission," *Cham Zug, Switzerland: Springer International Publishing*, vol. 106, 2016.

[4] H. Guo, Q. Chen, Q. Xia, and C. Kang, "Deep inverse reinforcement learning for reward function identification in bidding models," *IEEE Transactions on Power Systems*, 2021.

[5] Y. Ye, D. Papadaskalopoulos, J. Kazempour, and G. Strbac, "Incorporating non-convex operating characteristics into bi-level optimization electricity market models," *IEEE Transactions on Power Systems*, vol. 35, no. 1, pp. 163–176, 2019.

[6] I. Boukas, D. Ernst, T. Théate, A. Bolland, A. Huynen, M. Buchwald, C. Wynants, and B. Cornélusse, "A deep reinforcement learning framework for continuous intraday market bidding," *Machine Learning*, pp. 1–53, 2021.

[7] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski *et al.*, "Human-level control through deep reinforcement learning," *nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[8] G. Bertrand and A. Papavasiliou, "Adaptive trading in continuous intraday electricity markets for a storage unit," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2339–2350, 2019.

[9] D. Maragno, H. Wiberg, D. Bertsimas, S. I. Birbil, D. d. Hertog, and A. Fajemisin, "Mixed-integer optimization with constraint learning," *arXiv preprint arXiv:2111.04469*, 2021.

[10] Y. Chen, Y. Shi, and B. Zhang, "Modeling and optimization of complex building energy systems with deep neural networks," in *2017 51st Asilomar Conference on Signals, Systems, and Computers*. IEEE, 2017, pp. 1368–1373.

[11] ——, "Optimal control via neural networks: A convex approach," *arXiv preprint arXiv:1805.11835*, 2018.

[12] B. Chen, P. L. Donti, K. Baker, J. Z. Kolter, and M. Bergés, "Enforcing policy feasibility constraints through differentiable projection for energy optimization," in *Proceedings of the Twelfth ACM International Conference on Future Energy Systems*, 2021, pp. 199–210.

[13] A. Nagabandi, G. Kahn, R. S. Fearing, and S. Levine, "Neural network dynamics for model-based deep reinforcement learning with model-free fine-tuning," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 7559–7566.

[14] E. King, J. Drgona, A. Tuor, S. Abhyankar, C. Bakker, A. Bhattacharya, and D. Vrabie, "Koopman-based differentiable predictive control for the dynamics-aware economic dispatch problem," *arXiv preprint arXiv:2203.08984*, 2022.

[15] J. Drgoňa, S. Mukherjee, A. Tuor, M. Halappanavar, and D. Vrabie, "Learning stochastic parametric differentiable predictive control policies," *arXiv preprint arXiv:2203.01447*, 2022.

[16] X. Pan, M. Chen, T. Zhao, and S. H. Low, "Deepopf: A feasibility-optimized deep neural network approach for ac optimal power flow problems," *arXiv preprint arXiv:2007.01002*, 2020.

[17] F. Fioretto, T. W. Mak, and P. Van Hentenryck, "Predicting ac optimal power flows: Combining deep learning and lagrangian dual methods," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 01, 2020, pp. 630–637.

[18] A. Venzke, D. T. Viola, J. Mermet-Guyennet, G. S. Misyris, and S. Chatzivasileiadis, "Neural networks for encoding dynamic security-constrained optimal power flow to mixed-integer linear programs," *arXiv preprint arXiv:2003.07939*, 2020.

[19] Y. Chen, Y. Shi, and B. Zhang, "Data-driven optimal voltage regulation using input convex neural networks," *Electric Power Systems Research*, vol. 189, p. 106741, 2020.

[20] I. Murzakhanov, A. Venzke, G. S. Misyris, and S. Chatzivasileiadis, "Neural networks for encoding dynamic security-constrained optimal power flow," *arXiv preprint arXiv:2003.07939*, 2020.

[21] L. Halilbašić, F. Thams, A. Venzke, S. Chatzivasileiadis, and P. Pinson, "Data-driven security-constrained ac-opf for operations and markets," in *2018 power systems computation conference (PSCC)*. IEEE, 2018, pp. 1–7.

[22] D. Kraft *et al.*, "A software package for sequential quadratic programming," 1988.

[23] R. Nellikkath and S. Chatzivasileiadis, "Physics-informed neural networks for ac optimal power flow," *Electric Power Systems Research*, vol. 212, p. 108412, 2022.

[24] J. Lago, F. De Ridder, and B. De Schutter, "Forecasting spot electricity prices: Deep learning approaches and empirical comparison of traditional algorithms," *Applied Energy*, vol. 221, pp. 386–405, 2018.

[25] D. S. Kirschen and G. Strbac, *Fundamentals of power system economics*. John Wiley & Sons, 2018.

[26] K. Bruninx, K. Van den Bergh, E. Delarue, and W. D'haeseleer, "Optimization and allocation of spinning reserves in a low-carbon framework," *IEEE Transactions on Power Systems*, vol. 31, no. 2, pp. 872–882, 2015.

[27] J. Fortuny-Amat and B. McCarl, "A representation and economic interpretation of a two-level programming problem," *Journal of the Operational Research Society*, vol. 32, no. 9, pp. 783–792, 1981.

[28] T. Kleinert, M. Labbé, F. a. Plein, and M. Schmidt, "There's no free lunch: on the hardness of choosing a correct big-m in bilevel optimization," *Operations research*, vol. 68, no. 6, pp. 1716–1721, 2020.

[29] S. Pineda and J. M. Morales, "Solving linear bilevel problems using big-ms: Not all that glitters is gold," *IEEE Transactions on Power Systems*, vol. 34, no. 3, pp. 2469–2471, 2019.

[30] Gurobi Optimization, LLC, "Gurobi Optimizer," 2021. [Online]. Available: https://www.gurobi.com

[31] C. Ruiz, A. J. Conejo, and S. A. Gabriel, "Pricing non-convexities in an electricity pool," *IEEE Transactions on Power Systems*, vol. 27, no. 3, pp. 1334–1342, 2012.

[32] J. Lago, G. Marcjasz, B. De Schutter, and R. Weron, "Forecasting day-ahead electricity prices: A review of state-of-the-art algorithms, best practices and an open-access benchmark," *Applied Energy*, vol. 293, p. 116983, 2021.

[33] K. Hornik, M. Stinchcombe, and H. White, "Multilayer feedforward networks are universal approximators," *Neural networks*, vol. 2, no. 5, pp. 359–366, 1989.

[34] J. Martens, *Second-order optimization for neural networks*. University of Toronto (Canada), 2016.

[35] P. Xu, F. Roosta, and M. W. Mahoney, "Second-order optimization for non-convex machine learning: An empirical study," in *Proceedings of the 2020 SIAM International Conference on Data Mining*. SIAM, 2020, pp. 199–207.

[36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[37] B. Amos, L. Xu, and J. Z. Kolter, "Input convex neural networks," in *International Conference on Machine Learning*. PMLR, 2017, pp. 146–155.

[38] J. Bezanson, S. Karpinski, V. B. Shah, and A. Edelman, "Julia: A fast dynamic language for technical computing," *arXiv preprint arXiv:1209.5145*, 2012.

[39] M. Innes, E. Saba, K. Fischer, D. Gandhi, M. C. Rudilosso, N. M. Joy, T. Karmali, A. Pal, and V. Shah, "Fashionable modelling with flux," *CoRR*, vol. abs/1811.01457, 2018. [Online]. Available: https://arxiv.org/abs/1811.01457

[40] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, ..., P. van Mulbregt, and SciPy 1.0 Contributors, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020.

[41] S. Dempe, *Foundations of bilevel programming*. Springer Science & Business Media, 2002.

[42] Elia NV, "Grid data," Available at http://www.elia.be/en/grid-data, 2022.

[43] EPEX SPOT, "Market data," Available at https://www.epexspot.com/en/market-data, 2022.

[44] P. PXs, "Euphemia public description. pcr market coupling algorithm," *Tech. Rep.*, 2016.

[45] J. Huchette and J. P. Vielma, "Nonconvex piecewise linear functions: Advanced formulations and simple modeling tools," *arXiv preprint arXiv:1708.00050*, 2017.

[46] J. Bottieau, Y. Wang, Z. De Grève, F. Vallée, and J.-F. Toubeau, "Interpretable transformer model for capturing regime switching effects of real-time electricity prices," *IEEE Transactions on Power Systems*, 2022.

Mihaly Dolanyi received his Bachelor's degree in mechanical engineering at the Budapest University of Technology and Economics. He obtained InnoEnergy's ENTECH M.Sc. degree, from Institut Polytechnique de Grenoble, and Karlsruhe Institute of Technology. He received his Ph.D. degree in mechanical engineering, from KU Leuven. He is currently a postdoctoral researcher in the Power Systems and Markets Research Group in the University of Mons. His research concerns the joint exploitation of learning and mathematical programming methods in power systems problems.

Kenneth Bruninx (M'16) received the M.Sc. degree in energy engineering in 2011, the M.Sc. degree in management in 2015 and the Ph.D. degree in mechanical engineering in 2016, all from the University of Leuven (KU Leuven), Belgium. Currently, he is working as an assistant professor at the Faculty of Technology, Policy & Management, TU Delft (The Netherlands) and as a research fellow at the University of Leuven Energy Institute, TME branch (energy conversion) (Belgium).

Jean-François Toubeau received the degree in civil electrical engineering, and the Ph.D. degree in electrical engineering, from the University of Mons (Belgium) in 2013 and 2018, respectively. He is currently a postdoctoral researcher of the Belgian Fund for Research (F.R.S/FNRS) within the "Power Systems and Markets Research Group" of the same University. His research mainly focuses on bridging the gap between machine learning and decision-making in modern power systems.

Erik Delarue received the M.S. degree in mechanical engineering and the Ph.D. degree in mechanical engineering, both from the University of Leuven (KU Leuven), Leuven, Belgium, in 2005 and 2009, respectively. Currently, he is an Assistant Professor at the University of Leuven Energy Institute, TME branch (energy conversion) and active in EnergyVille.