
Travail de fin d'études et stage[BR]- Travail de fin d'études : Development of machine learning-based surrogate model in the European power system[BR]- Stage

Auteur : Cloux, Romain

Promoteur(s) : Quoilin, Sylvain

Faculté : Faculté des Sciences appliquées

Diplôme : Master en ingénieur civil électromécanicien, à finalité spécialisée en énergétique

Année académique : 2023-2024

URI/URL : <http://hdl.handle.net/2268.2/20874>

Avertissement à l'attention des usagers :

Tous les documents placés en accès ouvert sur le site le site MatheO sont protégés par le droit d'auteur. Conformément aux principes énoncés par la "Budapest Open Access Initiative"(BOAI, 2002), l'utilisateur du site peut lire, télécharger, copier, transmettre, imprimer, chercher ou faire un lien vers le texte intégral de ces documents, les disséquer pour les indexer, s'en servir de données pour un logiciel, ou s'en servir à toute autre fin légale (ou prévue par la réglementation relative au droit d'auteur). Toute utilisation du document à des fins commerciales est strictement interdite.

Par ailleurs, l'utilisateur s'engage à respecter les droits moraux de l'auteur, principalement le droit à l'intégrité de l'oeuvre et le droit de paternité et ce dans toute utilisation que l'utilisateur entreprend. Ainsi, à titre d'exemple, lorsqu'il reproduira un document par extrait ou dans son intégralité, l'utilisateur citera de manière complète les sources telles que mentionnées ci-dessus. Toute utilisation non explicitement autorisée ci-avant (telle que par exemple, la modification du document ou son résumé) nécessite l'autorisation préalable et expresse des auteurs ou de leurs ayants droit.



University of Liège - Faculty of Applied Sciences

MASTER'S THESIS

**Development of machine learning-based surrogate
model in the European power system.**

Master's thesis completed to obtain the degree of Master of Science in
Electromechanical Engineering by CLOUX Romain

Supervisor:

Prof. S. Quoilin

Academic year 2023-2024

Acknowledgements

Thanks to Professor QUOILIN S. for his patience and help throughout this thesis.

Thanks to STRAET F. for his preliminary work [1] and for the many hours spent with me, on computational concepts explanation and for bringing his advices.

Thanks to TAREEN U. and ORELLANA NAVIA M. (PhD candidates) for their help.

Finally, thanks to my family and my friends for the encouragement during all this work.

Abstract

This work centers on the development of a machine learning-based surrogate model to enhance the efficiency of simulating the European power system. Initially, a series of simulations will be conducted using Dispa-SET, a short-term dispatch optimization model employed to manage the operation of the European power grid. Dispa-SET is essential for balancing electricity supply and demand, ensuring grid efficiency, and minimizing operational costs by optimizing the dispatch of generating units while considering various operational constraints and market conditions.

To inform the development of the surrogate model, these simulations will involve varying different inputs and analyzing the resulting outputs. This comprehensive analysis will identify key patterns and relationships within the data, which will then be used to construct a surrogate model that accurately approximates Dispa-SET's outcomes. The surrogate model will focus on replicating critical results related to load shedding and curtailment. Both aspects are crucial for effective power system simulation and management, as they significantly affect grid stability and the integration of renewable energy sources.

By accurately mirroring these key outcomes, the surrogate model aims to provide a more efficient and flexible tool for analyzing and optimizing power grid operations, ultimately facilitating better decision-making and operational planning in the European energy system.

Contents

Keywords	vi
1 Introduction	1
1.1 Context	1
1.2 Short-term dispatch models	1
1.3 Surrogate models - state of art.	1
1.4 Objectives	2
1.5 Contributions	2
1.6 Outline	3
2 Dispa-SET	4
2.1 Objective function	5
2.2 Constraints	5
2.2.1 Supply-demand balance	5
2.2.2 Other constraints	6
2.3 Rolling horizon	6
2.4 Mid-Term Scheduling	7
2.5 Model formulations	8
2.5.1 Integer Clustering	8
2.5.2 Linear Programming Clustering	8
2.5.3 Comparison	8
2.6 Reference case simulation	9
2.6.1 MILP vs LP formulation	9
2.7 MILP computational costs	11
3 Dispa-SET model results	13
3.1 Surrogate model dimensionless features	13
3.2 Reference Values : October 2019	15
3.3 Sensitivity analysis	15
3.3.1 Share of VRES : PV, wind onshore and offshore	15
3.3.2 Share of flexible units	19
3.3.3 Ratio of Net transfer capacity (rNTC)	22
3.3.4 Share of storage	24
3.3.5 Capacity Ratio	27
4 Sampling - Dispa-SET simulation - Dataset generation	29
4.1 Overview	29

4.2	Data preparation and initial parameters	29
4.2.1	Unit groupings	29
4.2.2	Parameters estimated [1]	30
4.3	Design space	30
4.3.1	Six dimensionless features of the surrogate model.	31
4.3.2	Reference simulation and bounds	31
4.3.3	Targets of the surrogate model.[1]	34
4.4	Generation of the dataset	35
4.4.1	Adjusting ShareFlex	35
4.4.2	Adjusting Capacity Ratio, share of Storage/PV/Wind	35
4.4.3	Adjusting rNTC	36
4.4.4	Extracted outputs and Datasae features	37
4.5	Simulations failed	37
5	The surrogate Models	40
5.1	Overview	40
5.2	Data preparation and correlations	40
5.2.1	Cleaning	40
5.2.2	Correlations	40
5.3	Scaling and Splitting	41
5.3.1	Splitting	41
5.3.2	Scaling	42
5.4	Model selection	42
5.4.1	Multi-layer Perceptron (MLP) Regressor	42
5.4.2	MLP - hyperparameters	44
5.4.3	Random Forest	45
5.4.4	RF - hyperparameters	47
5.4.5	Comparison between RF and ANN	48
5.5	Model Building	48
5.6	Curtailement - Hyperparameters and models comparison	49
5.7	Curtailement - Results	50
5.7.1	Execution Times	52
5.7.2	Learning Curve	52
5.7.3	Loss curves	53
5.7.4	Feature importances	54
5.7.5	Surface plots - Curtailement	55
5.8	Load Shedding - Hyperparameters and models comparison	57
5.9	Load Shedding - Results	60

5.9.1	Execution Times	61
5.9.2	Feature importances	62
5.10	Load Shedding - MLP Regressor with over-sampling	62
5.10.1	Execution Times	63
5.10.2	Learning Curve	63
5.10.3	Loss curves	65
5.10.4	Surface plots - Load shedding	65
5.11	Further results beyond the regression range	67
6	Conclusion	70
6.1	Future prospects for continuing this work	71

Keywords

AF	Availability factor
ANN	Artificial neural network
CF	Capacity factor
UE	European union
GAMS	General algebraic modeling language
IAMs	Integrated assessment model
LP	Linear programming
LHS	Latin hypercube sampling
MILP	Mixed integer linear program
MLP	Multi-layer perceptron
MTS	Mid-term scheduling
rNTC	Ratio of Net transfer capacity
PHS	Pumped hydro-storage
PV	Photovoltaic
RES	Renewable energy sources
SLURM	Simple utility for Resource Management
UCM	Unit commitment model
VRES	Variable renewable energy sources

1 Introduction

1.1 Context

Surrogate modeling [2] is an interdisciplinary research field that leverages data-efficient machine-learning techniques to speed up the analysis and optimization of high-fidelity simulations that might otherwise take weeks or even months to complete. Surrogate models serve as fast-running approximations of complex, time-consuming computer simulations, enabling quicker decision-making and more efficient resource allocation.

In the context of European energy systems, surrogate modeling can be applied to short-term dispatch models which are used to optimize the operation of power grids. These models are crucial for balancing electricity supply and demand on a daily basis while minimizing operational costs and ensuring grid stability. However, running these optimizations frequently can become computationally expensive and time-intensive, particularly when simulating large and complex power systems like the European one.

By developing surrogate models that accurately approximate the outcomes of these dispatch optimizations, it becomes possible to significantly reduce computational costs and time. This allows for more frequent and flexible simulations, which are essential for better real-time decision-making, scenario analysis, and planning in response to the dynamic nature of energy markets.

1.2 Short-term dispatch models

Tools have been developed to evaluate the behavior of large electrical systems with significant variable renewable energy sources (VRES). These tools predict electricity flows and optimize power plant dispatch to balance production with demand, using short time steps (e.g., one hour) and covering simulation periods up to a year [3]. This temporal resolution is essential for accurately capturing renewable energy variability and it includes detailed simulations of all generation units.

For this work, the Dispa-SET model [4] has been selected. Dispa-SET is an open-source tool designed to address balancing issues in the European grid using linear programming. It defines and solves a set of linear constraints and an objective function through the GAMS solver [5], optimizing the dispatch of power plants to maximize the objective function while meeting all constraints.

1.3 Surrogate models - state of art.

Surrogate models [6] are simplified versions of complex, high-fidelity models, used to approximate the relationship between input and output data when direct evaluation is too costly or

unknown. They are particularly valuable in scenarios involving expensive simulations or poorly understood relationships.

Surrogate models using artificial neural networks (ANNs) have been successfully applied to optimize hydropower operations under environmental constraints. In a recent study [7], a high-fidelity hydrodynamic and water quality model was emulated by an ANN, which was then integrated into an optimization algorithm. This approach, applied to a reservoir in Tennessee, increased hydropower production while meeting dissolved oxygen limits, demonstrating how ANNs can reduce computational demands and support efficient, environmentally compliant hydropower management.

Still in the energy sector, surrogate models, particularly artificial neural networks (ANNs), have been applied to improve the calibration of building energy models (BEMs) [8]. A recent study developed an automated method using an ANN to directly infer unknown building parameters, accounting for factors like weather and building schedules. This approach not only outperformed traditional optimization methods but also demonstrated greater consistency. Based on the study's results, the ANN method requires only a single calibration trial, while multiple runs are needed with other methods, like RBFOpt, to achieve consistent calibration.

1.4 Objectives

The primary objective of this work is to develop a surrogate model specifically designed to approximate the outcomes of the DISPA-SET dispatch optimizations. By accurately mimicking the behavior of the DISPA-SET model, the surrogate model will enable much faster simulations, significantly reducing computational costs and time.

In a future phase of this research, the developed surrogate model will be integrated into Integrated Assessment Models (IAMs) such as MEDEAS (Modelling the Energy Development under Environmental and Socioeconomic constraints). MEDEAS is designed to assess the feasibility and impacts of transitioning to a low-carbon economy by analyzing the interactions between energy systems, the economy, and the environment. Incorporating the surrogate model into MEDEAS will enhance its computational efficiency and accuracy, enabling more robust long-term energy scenario analysis. This will ultimately support the formulation of informed and effective energy policies for a sustainable future in Europe.

1.5 Contributions

In this work, personal contributions are described below :

- Adding my energetic point of view to the code previously done by STRAET F. [1] allowing variations of a reference simulation on Dispa-SET.

- Adaptations have been made to use these codes on Windows and also to deal with new versions of GAMS [5].
- Some energy-oriented corrections have been realized concerning the definition of the adimensional inputs for the surrogate model.
- Research have been realized based on different existing models in order to modify/verify bounds for the space design, the six inputs, of the model used.
- Modifying some of the adjusting functions and the code linked to it allowing to vary the inputs.
- Considering the model Dispa-SET, an analysis describing the difference between the two model formulations has been made: *Linear Programming (LP)* and *Mixed-Integer Linear Programming (MILP)*.
- A sensitivity analysis is conducted based on a reference case and different selection parameters to understand better the behavior of these inputs on the system and on the model.
- Adding theoretical concepts linked to ANN and RF models.
- Development of `models.ipynb` for Curtailment and Load shedding.
- Adding corrected results and energy analysis for the different models.

The resulting work is available in the GitHub repository: https://github.com/PtitVerredeRhum/Master_Thesis_Cloux

1.6 Outline

The thesis outline is described as :

- **Section 2:** Description of the Dispa-SET model.
- **Section 3:** Results of the Dispa-SET model.
- **Section 4:** Dataset generation and adjusting functions.
- **Section 5:** Development of the surrogate models.
- **Section 6:** Conclusion.

2 Dispa-SET

The Dispa-SET model [4] is an open-source unit commitment and optimal dispatch model focused on the balancing and flexibility problems in European grids. This optimization model aims at representing the short-term operation of large-scale power systems solving the unit commitment problem with a high level of detail . It consists of two main parts :

- **Unit scheduling:** This part involves making decisions about when to start-up, operate and shut down units over multiple periods of time. Binary variables are used to describe their on/off status as well as different constraints to ensure consistency over that period.
- **The economic dispatch :** It focuses on allocating the total demand among the available generation units over each period to minimize the overall system costs. This involves determining the continuous power output of each generation unit.

Given the problem features, it can be expressed as a mixed-integer linear programming program (MILP) or also relaxed to a linear program (LP). Both MILP and LP will be described in section 2.5.

A schematic of the Dispa-SET architecture is displayed in Figure 2.1.

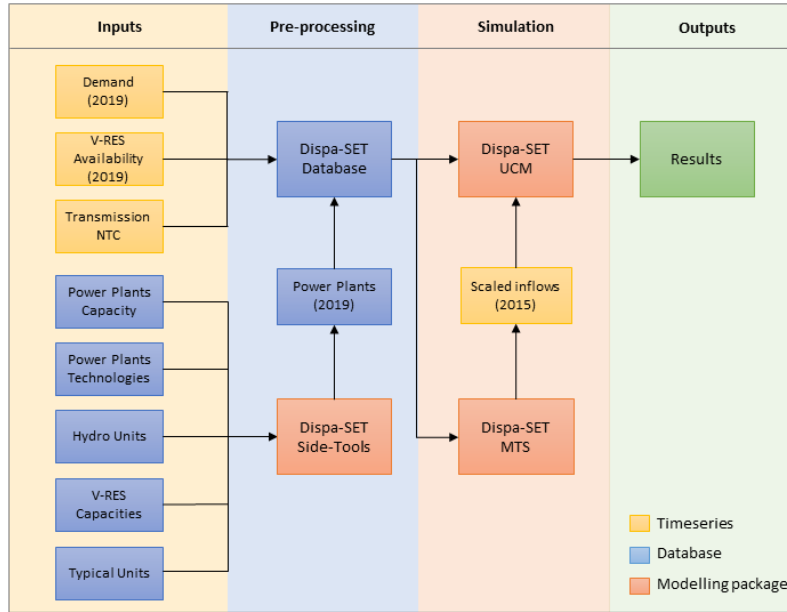


Figure 2.1: Block diagram of the architecture of the Dispa-SET model.

The interface is written using Python programming language and calls a General Algebraic Modeling System (GAMS) [5] as main solver engine.

The objective of GAMS is to provide a modeling environment for solving mathematical optimization problems. It supports a wide range of optimization problems, including linear programming, nonlinear programming, integer programming, and mixed-integer programming.

2.1 Objective function

The unit commitment problem aims to minimize the total power system cost defined as the sum of the different cost items (See Equation 2.1). Those different costs are listed below [4] :

- Fixed : depending on whether the unit is on or off.
- Variable : stemming from the power output of the units.
- Ramp-up : emerging from the ramping up of a unit.
- Ramp-down : emerging from the ramping down of a unit.
- Load shed : due to necessary load shedding.
- Transmission : depending of the flow transmitted through the lines.
- Loss of load: power exceeding the demand or not matching it, ramping and reserve.
- Spillage: due to spillage in storage.
- H2: cost of unsatisfied hydrogen by production from electrolyzers
- Water : cost of water coming from unsatisfied water level at the end of the optimization period.

$Min_{TotalSystemCost} =$

$$\begin{aligned}
& \sum_{u,i} (CostRampUp_{u,i} + CostRampDown_{u,i}) + \\
& \sum_{u,i} CostFixed_u \cdot TimeStep + \\
& \sum_{u,i} CostVariable_{u,i} \cdot Power_{u,i} \cdot TimeStep + \\
& \sum_{n,i} CostLoadShedding_{n,i} \cdot ShedLoad_{n,i} \cdot TimeStep + \\
& \sum_n VOLL \cdot LostLoad \cdot TimeStep \\
& \sum_{s,i} CostOfSpillage \cdot Spillage_{s,i}
\end{aligned} \tag{2.1}$$

Equation 2.1: Objective function of the Dispa-SET model.

2.2 Constraints

2.2.1 Supply-demand balance

The main constraint to satisfy is the balance between supply and demand, observed for every period and zone within the day-ahead market. This constraint stipulates that the total

power generation from all units within the node, including energy generated by storage units, alongside with power inflow from neighboring nodes and the curtailed power from intermittent sources must be equal to the local demand in that node. This demand is further adjusted by the power utilized for energy storage reduced by any interrupted load and adjusted for any load shedding (See equation 2.2).

$$\begin{aligned}
& \sum_u (Power_{u,i} \cdot Location_{u,n}) + \sum_l Flow_{l,n,i} \\
& = Demand_{n,i} + \sum_s (StorageInput_{s,i} \cdot Location_{s,n}) + \\
& \quad - ShedLoad_{n,i} - LostLoad
\end{aligned} \tag{2.2}$$

Equation 2.2: Supply-demand balance constraint in Dispa-SET.

2.2.2 Other constraints

Furthermore, there are other important constraints that need to be taken into account, as described in the Dispa-SET documentation [4]. Those are :

- Reserve constraints and requirements.
- Power output bounds
- Ramping constraints
- Storage constraints
- Heat constraints
- Emission limits
- Network-related constraints
- Load shedding constraints

2.3 Rolling horizon

The mathematical problem discussed earlier could be computationally demanding if solved for an entire year. To manage that issue, the problem is split into smaller optimization problems that run recursively throughout the year. An example depicted below shows a two-day optimization horizon with a one-day look-ahead period. Initial values for day j are derived from the previous day's final values. The look-ahead period prevents issues at the optimization period's end. Only the results of the first 24 hours are retained.

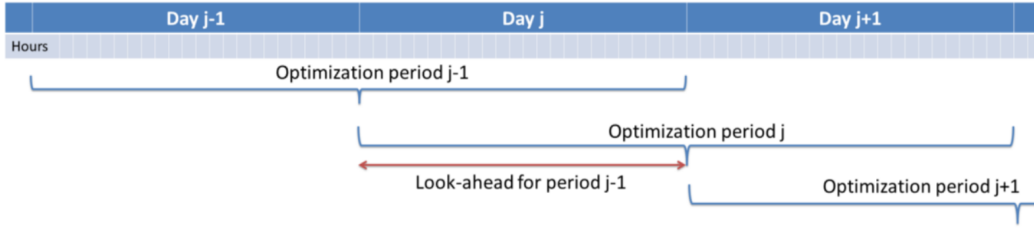


Figure 2.2: Rolling horizon illustration.

2.4 Mid-Term Scheduling

Accurately collecting historical storage levels as hourly time-series is often challenging or nearly impossible. This issue affects the reliability of future forecasts based on this data. In systems heavily reliant on hydro dams (HDAM) and pumped hydro storage (HPHS), like in Norway, the absence of precise historical data can also significantly impact simulation results.

To prevent this, the Dispa-SET model must operate in Mid-Term Scheduling (MTS) mode. In this mode, the initial and final levels of storage units (especially pumped hydro storage units) are specified as external inputs to the model and enforced through additional constraints.

However, this method has specific requirements. The model must use Linear Programming (LP) and the time resolution has to be increased to one day. In this mode, all equations related to unit commitment are excluded, and the binary variables Committed StartUp, and ShutDown are not defined. Consequently, some constraints are ignored.

The MTS allows different options:

- **No-MTS**, using historical curves,
- **Zonal-MTS**, where MTS is applied to each Zone individually,
- **Regional-MTS**, where MTS is applied to multiple Zones simultaneously within a region.

The third one is selected meaning that all zones are selected as a "region". In this formulation, MTS will determine reservoir levels using additional criteria like available net transfer capacities (NTC), rather than relying on historical cross-border flows (CBF).

In this work, MTS mode is used while setting half of the storage capacity for the initial and final levels.

2.5 Model formulations

In Dispa-SET, several formulation types can be used and are implying different hypotheses and different levels of complexity. This thesis utilizes the LP formulation as it requires less computational resources than the MILP formulation. However, a comparison between the two is conducted to understand their influence on the Dispa-SET results.

2.5.1 Integer Clustering

The MILP formulation is suitable for that kind of problem because it involves both discrete, i.e. on/off status, and continuous, i.e. output generation units, variables. Indeed, the MILP formulation can deal with the two types of variables.

Moreover, all units from the same zone that use the same fuel and technology are clustered together. While the total number of units is preserved, the power capacity is averaged across all units within the cluster.

2.5.2 Linear Programming Clustering

This problem can also be relaxed to a LP problem by ignoring the binary variables representing on/off statuses, using only continuous variables instead.

Since the binary variables can no longer be expressed, the constraints listed below are removed from the problem :

- Minimum up and down times.
- Start-up costs.
- Minimum stable load.

As a result, the clustering is different from the MILP formulation.

The clustering is made by merging all the units of similar technology, with the same fuel and from the same zone. Indeed, keeping the total number of units is not useful anymore since the start-up constraints are not taking into account.

2.5.3 Comparison

Trivially, the difference between both formulations is a trade-off between accuracy and computational efficiency. The MILP formulation is more accurate but requires much more computational resources in contrast to the LP formulation.

2.6 Reference case simulation

A reference simulation is first built for the entire year 2019, serving as the basis for this work. Initially, this base scenario is simulated using both LP and MILP formulations for comparison purposes. Then, a sensitivity analysis is conducted separately on the six features of the surrogate model to better understand the role of each one.

2.6.1 MILP vs LP formulation

Here is an example of the power dispatch in France simulated for October 2019. As it can be observed on Figure 2.3, the power generated by the nuclear sources obtained with the LP formulation varies a lot compared to the one simulated with the MILP formulation, see Figure 2.4. By relaxing the constraints about unit scheduling in LP, the production is no longer subject to the start-up costs of nuclear. So, the problem is relaxed and the system can freely switch on and off nuclear power plants which, in reality, is not happening. Moreover, the constraints about minimum up and down times and minimum stable load no longer apply in LP, allowing much more variation in the production coming from those units compared to the MILP formulation.

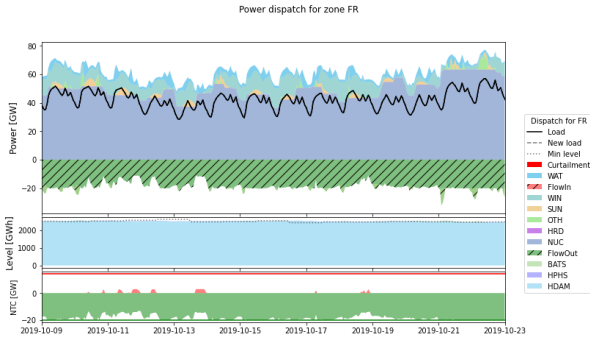


Figure 2.3: Reference case: Power Dispatch - two weeks observation - France - LP formulation.

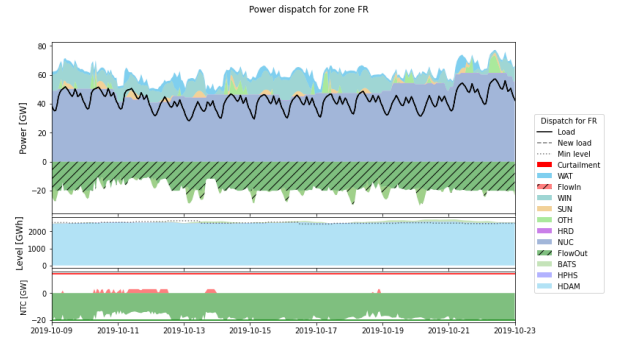


Figure 2.4: Reference case: Power Dispatch - two weeks observation - France - MILP formulation.

Another example is taken from Norway, a country with a high level of renewable energy generation. As one can see in Figures 2.5 and 2.6, during the 4th of February, the LP formulation shows the presence of power produced by wind turbines offshore while the MILP does not. This can be explained the same way as in the previous example : by the fact that the MILP problem is more constrained. By minimizing the objective cost function, the system realizes that, in MILP, switching off the power from wind turbines offshore is more cost-effective than letting the wind produce electricity and that leads to curtailment during this period as it can be observed in Figure 2.8 compared to Figure 2.7.

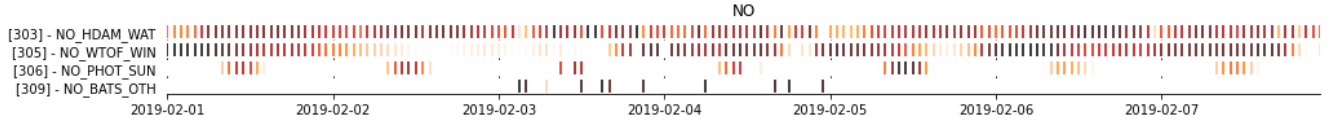


Figure 2.5: Reference case: Generation Units - one-week observation - Norway - LP formulation.

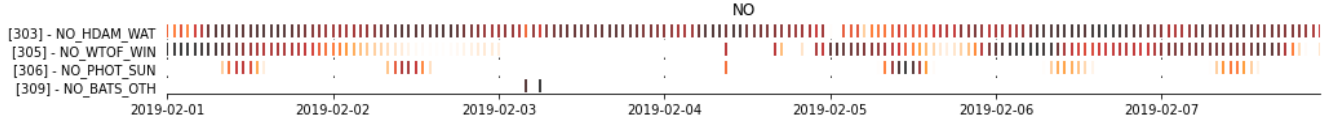


Figure 2.6: Reference case: Generation Units - one-week observation - Norway - MILP formulation.

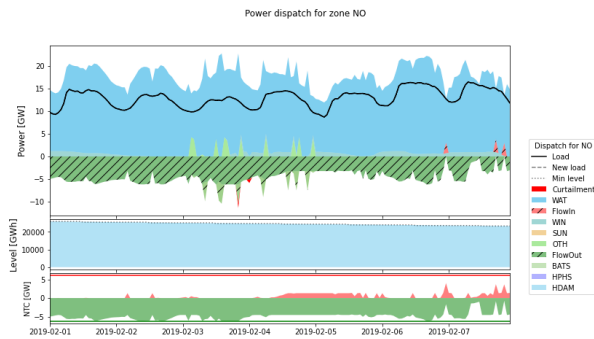


Figure 2.7: Reference case: Power Dispatch - one-week observation - Norway - LP formulation.

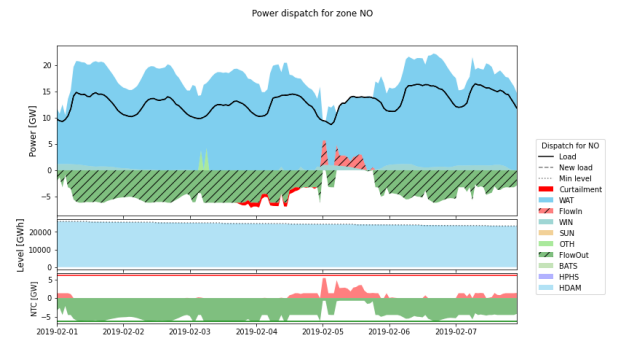


Figure 2.8: Reference case: Power Dispatch - one-week observation - Norway - MILP formulation.

These two examples show that the MILP is suitable for the unit commitment problem because it can deal with both continuous and integer variables. However, this formulation requires much more computational costs than the LP one. Conversely, the unit commitment problem can be relaxed to the LP formulation. But because this formulation can not deal with the binary variables, it doesn't take into account some of the constraints described in the previous section. This is much more computationally effective but less accurate, and so less realistic than the MILP formulation.

Here is a comparison table of LP vs MILP containing some interesting data about the reference simulation for one year :

	LP Formulation	MILP Formulation
Average Electricity cost (€/MWh)	90.2	91.14
Total Load Shedding (TWh)	0.0	0.412
Maximum Load Shedding (MW)	0.0	7127
Total Curtailed RES (TWh)	4.18	4.6
Maximum Curtailed RES (TW)	0.064	0.0627
Computation time (h)	1h20	14h25

Table 2.1: Comparison of quantitative outputs for 2019 simulation.

As it can be observed in the table above, the model formulation influences the results.

It can be seen that the system has no choice but to shed the load sometimes. Considering the high load-shedding cost fixed voluntarily at 1000€/MWh, it means that the system is forced to shed the load considering the additional three constraints specific to the MILP formulation. It can be explained by the fact that the units cannot be switched on/off at each time and also due to start-up costs. Meaning that it is less expensive to shed load than to switch on units in such a case.

Another observation is that the computation time gives a good understanding of the complexity of the MILP formulation. It requires about ten times more time than the LP formulation due to the fact that the problem is more constrained.

As a final observation, the same reasoning applies to the slight increase in curtailed energy. An hypothesis could be that due to some units which cannot vary the amount produced when they want, because of the three constraints, sometimes, units using nuclear as fuel will continuously generate at full capacity even when VRES is high, leading to the low-cost solution to curtail the energy coming from VRES.

2.7 MILP computational costs

To understand better how much computational resources and time are needed to run some MILP simulations, three parallel simulations have been carried out on the ULiège cluster.

First, due to time limits, none of the simulations were able to find solutions within the 2 days time limit given by the cluster. The cluster administrator is not allowed to increase time limits by users but only by jobs, i.e. by simulations.

Then, the three simulations were launched again, with a 8-day time limits. Two of them were successfully carried out after 63 hours while the other ran out of time.

Given that 64 CPUs can run simultaneously, and each of the 2,500 simulations takes an average of three days, this would result in 120 days of simulations, read four months, if the administrator allows an extension of the job duration to three days. However, the administrator is unable to increase the user's time limits in bulk. Instead, he must manually extend the time limits for each series of jobs one by one. In this example, with 40 series of 64 jobs, the administrator would need to manually increase the time limits every three day over the course of 120 days.

A more efficient scenario would involve running 128 CPUs in parallel, which would reduce the total simulation time to at most 51 days. This would require only 17 series of jobs, and consequently, the administrator would need to make only 17 manual adjustments to the time limits.

After discussion with the administrator, increasing the time limits for a considering number of jobs is not possible. Considering this information, two scenarios are emerging :

- First, finding a solution to decrease the simulation time.
- The second one is to try to use another cluster with higher times limit as *Hercule* and *Dragon* but allowing fewer CPUs simultaneous per user.

3 Dispa-SET model results

To better understand how one can interpret the results obtained after running a Dispa-SET simulation, a sensitivity analysis is performed on the six parameters used as inputs for the Dispa-SET. Those parameters are modified to become dimensionless, as per requested in this work. In the following part of the work, these six dimensionless inputs correspond to the features of the surrogate model.

After facing some computational issues with the GAMS solver on the cluster of ULiège *NIC5* in a one-year-long simulation, it has been decided that running the simulations over a one month period would be satisfactory regarding the nature of a sensitivity analysis. Thus, the simulation time frame has been set to October 2019 while knowing that one could miss out on some key points like seasonality analysis. Typically, the PV energy profile is higher during summer. Some of the concepts related to heating values,...

The starting point of the sensitivity analysis is to define the parameters and simulate a reference case. After that, based on the reference case over the selected month, one parameter will be modified at a time and new simulations run.

3.1 Surrogate model dimensionless features

1. CapacityRatio [.]

The ratio of maximum production over the maximum demand.

$$CapacityRatio = \frac{PowerCap_{flexunits} + PowerCap_{slowunits}}{PeakLoad} \quad (3.1)$$

2. ShareFlexibility [.]

The share of units that are flexible.

$$Share_{flex} = \frac{PowerCap_{flexunits}}{PowerCap_{flexunits} + PowerCap_{slowunits}} \quad (3.2)$$

3. ShareStorage [.]

The ratio of maximum power output of stationary batteries over the maximum demand.

$$Share_{storage} = \frac{PowerCap_{storageunits}}{PeakLoad} \quad (3.3)$$

4. ShareWind [-]

The ratio of electricity generated by wind onshore units over the total demand.

$$Share_{wind-on} = \frac{PowerCap_{windonunits} \cdot CF_{WTON}}{PeakLoad \cdot CF_{load}} \quad (3.4)$$

The ratio of electricity generated by wind offshore units over the total demand.

$$Share_{wind-off} = \frac{PowerCap_{windoffunits} \cdot CF_{WTOF}}{PeakLoad \cdot CF_{load}} \quad (3.5)$$

5. SharePV [-]

The ratio of electricity generated by PV units over the total demand.

$$Share_{PV} = \frac{PowerCap_{PVunits} \cdot CF_{PV}}{PeakLoad \cdot CF_{load}} \quad (3.6)$$

6. rNTC [-]

The net transfer capacity ratio reflects the maximum line capacities allowing electricity share between zones. Data provided are time-series of the power exchange between each pair of zones.

Firstly, the average, on time, net transfer capacity is computed for each zone z to any other zone x over each N_h hour of the data :

$$NTC_{z \rightarrow x} = \frac{1}{N_h} \sum_h NTC_{z \rightarrow x, h} \quad (3.7)$$

Secondly, Equation 3.8 gives a representative value of the transfer capacity by zones by summing all the transfers of one zone to the others.

$$NTC_z = \sum_x NTC_{z \rightarrow x} \quad (3.8)$$

Finally, the Net Transfer Capacity ratio is defined as the sum of each zonal NTC but weighted each on the sum of the peak load for each zones :

$$rNTC = \sum_z \frac{NTC_z}{\sum_x PeakLoad_x} \quad (3.9)$$

3.2 Reference Values : October 2019

The reference values presented in Table 3.1 are only used for the sensitivity analysis. Reference values for a simulation over one year are computed in another section of the work.¹ Additionally, all the Figures presented in this section are representing simulations over a period of one month, October 2019.

	Reference Value [-]	Adjusted Value [-]
Capacity Ratio	1.2	0.6
rNTC	0.32	0.64
share_{flex}	0.41	0.8
share_{pv}	0.03	0.15
share_{sto}	0.55	3.00
share_{wind-off}	0.06	0.3
share_{wind-on}	0.013	0.065

Table 3.1: Reference Values and scaling for the sensitivity analysis - October 2019

Different scaling (increase/decrease) of the parameters are used depending on how theses parameters will vary in the future. 2050 scenarios [9] can be an indicator on how installed capacities and transfer capacities will vary in the next few years.

3.3 Sensitivity analysis

3.3.1 Share of VRES : PV, wind onshore and offshore

VRES like solar and wind will increase in the coming years due to the global push to reduce carbon emissions, advancements making them more efficient and affordable, and supportive policies. As the world shifts towards cleaner energy, VRES are set to play a growing role in the energy mix, driven by both environmental and economic factors.

What is the influence of the share of each of the VRES if scaled by a factor 5 ?

Installed capacities

The Figures below depict how the installed capacity can be influenced by the increase of the share of the different variable renewable energy sources by comparing the base scenario (left) and the modified one (right).

¹In 2019, no stationary batteries were installed in EU. To perform the sensitivity analysis, it has been considered that some were installed due to EU TIMES JRC MODEL, but not for the rest of the work. That is why a low value of 10MW of power capacity installed for each BATS unit is considered for the rest of this work.

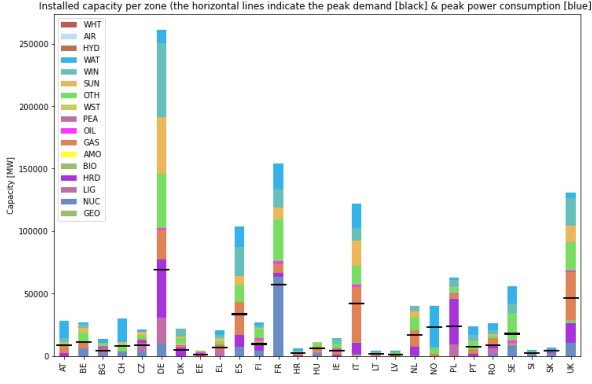


Figure 3.1: Reference case: Installed capacities by zones.

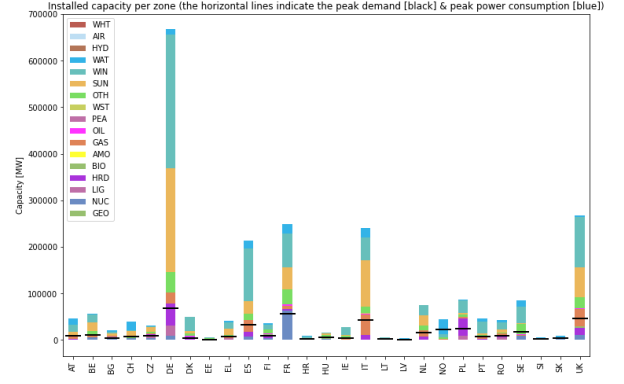


Figure 3.2: Adjusted share of VRES: Installed capacities by zones.

First, one can see that the scale of the installed capacities changes completely. The increase in VRES share impact directly the total VRES capacity installed (See Equation 3.4, 3.5 and 3.6) because the capacity factors stay unchanged and so does the peak demand.

Generation per zone

As the system wants to optimize a cost function, some variations are also observed in the total generation :

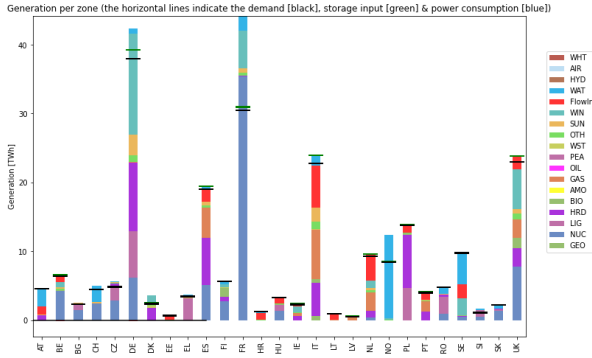


Figure 3.3: Reference case: Generation per zone.

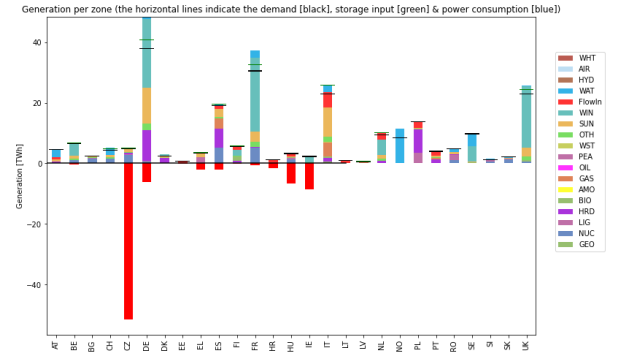


Figure 3.4: Adjusted share of VRES: Generation per zone.

In some countries, mainly in France (FR) and Germany (DE), where a larger part of the electricity is now generated by VRES, one can observe that the system chooses to switch off the most expensive energy sources such as nuclear in France and coal in Germany.

In Figure 3.4 and 3.6, one can see that due to the increase in VRES generated energy in Germany, the country would be forced to curtail huge amounts of that energy. As curtailment is a waste of energy, the option of exporting energy to neighbouring countries is preferred. In this particular example, Czech Republic (CZ) is importing most of its energy from Germany. This phenomenon can be observed in details in the results variables.

Due to the higher capacity installed of photovoltaic panels in Czech Republic, once the demand is fulfilled during peak solar period, the remaining electricity is either exported or stored using stationary batteries. This explain the different peaks displayed in Figure 3.8 in stored and exported energy.

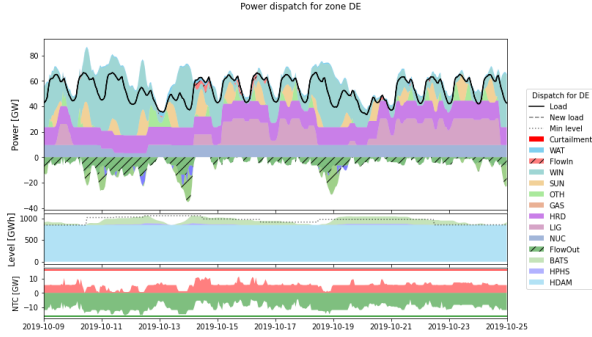


Figure 3.5: Reference case : Germany - Power dispatch.

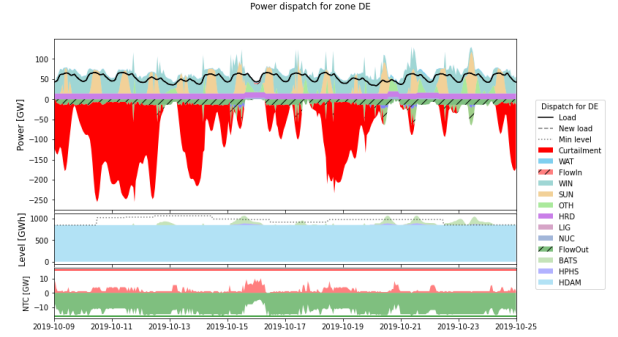


Figure 3.6: Adjusted share of VRES : Germany - Power dispatch.

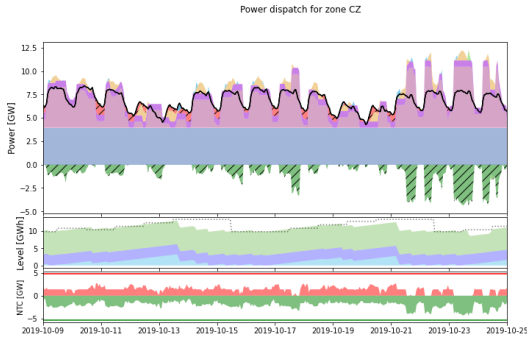


Figure 3.7: Reference case: Czech Republic - Power dispatch.

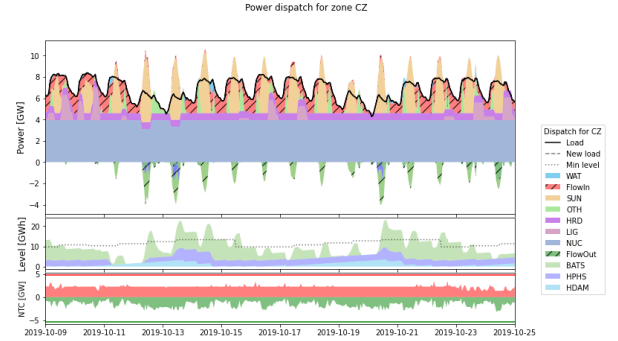


Figure 3.8: Adjusted share of VRES: Czech Republic - Power dispatch.

CO2 Emissions

As it could be expected, the tendency is a decrease in carbon dioxide emissions for countries where there is a decrease in the energy generation coming from Anthracite, coal, lignites,... Indeed, these sources are generally linked with steam turbine technologies which produce more CO₂ than the others. Good examples are Bulgaria and the Czech Republic in Figures 3.9 and 3.10.

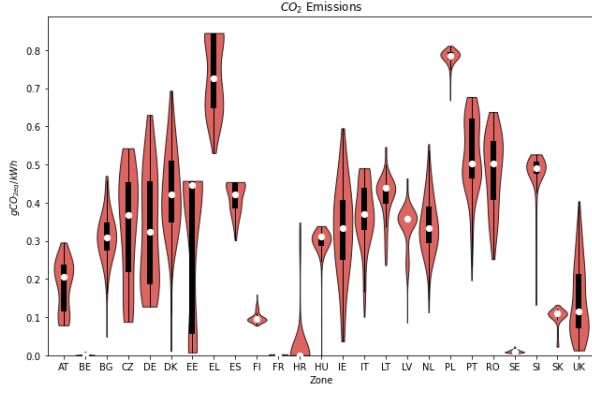


Figure 3.9: Reference case: Carbon dioxide emissions.

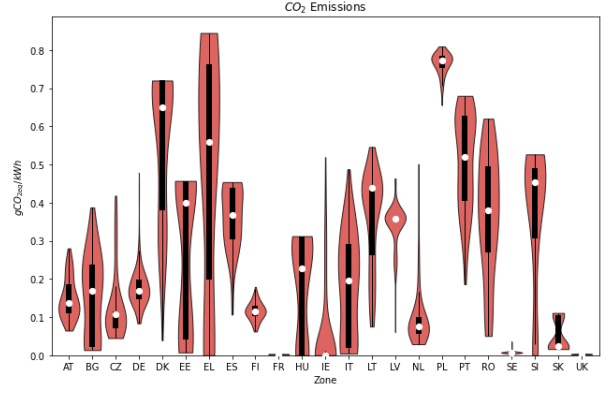


Figure 3.10: Adjusted share of VRES: Carbon dioxide Emissions.

Interconnection

In the two Figures below, it can be observed that a change in the capacity of VRES installed influences the import/export between the zones. Some of the countries with a big VRES potential like the United Kingdom and Sweden will increase their generation in VRES and so become "exporting countries" while in the reference case they were considered as "importing countries".

The same analysis with the fluxes, depicted by arrow on the Figures hereunder, can be conducted as the generation in each zones changed. This implies that the fluxes have to adapt their way. In general, it is observed that an increase in the VRES leads to an increase in the fluxes between each zones. So, the countries will have a tendency to export the energy rather than consuming it.

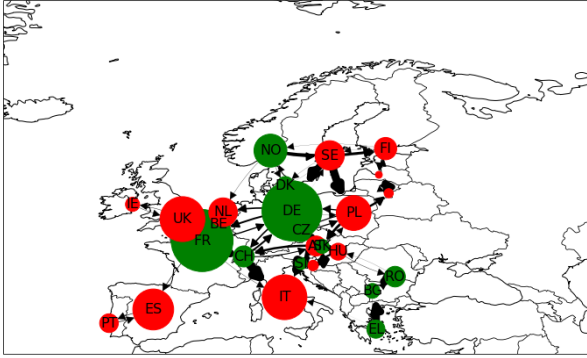


Figure 3.11: Reference case: Interconnections.

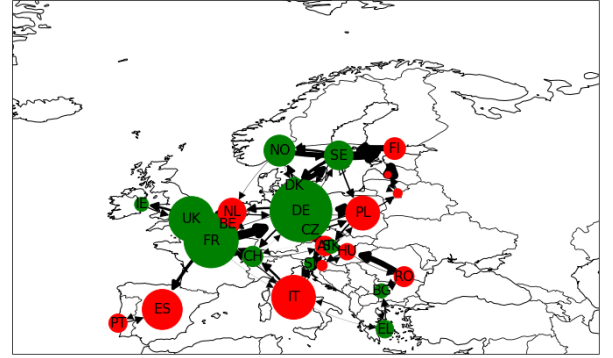


Figure 3.12: Adjusted share of VRES: Interconnections.

Conclusions

An increase in the share of VRES implies a bigger capacity installed of PV, wind onshore and offshore technologies. On the one hand, it reduces carbon dioxide emissions because more electricity is generated thanks to renewable sources. On the other hand, very high capacities

installed lead to high amount of curtailment due to the high amount of energy generated during peak periods.

Two solutions can be used to avoid the curtailment :

- Storage technologies : which can be used to store the extra amount of energy generated during peak period when the production exceeds the demand.
- Transport between zones : which is also a really important point, is to distribute this excess energy in other regions where VRES installed capacities are lower or where the demand is higher. So, sharing, through import/export, energy between each zones is very important to avoid curtailment.

3.3.2 Share of flexible units

Flexible power plants, such as those fueled by natural gas, are likely to see increased deployment in the coming years due to their ability to quickly ramp up and down production, complementing the intermittent nature of renewable energy sources like wind and solar. As the share of renewables in the energy mix grows, the need for reliable, on-demand power that can fill gaps during periods of low renewable output becomes crucial.

What is the influence of the share of flexible units doubles?

Installed capacities - Power and Heat dispatch

The total installed capacity of each country remains similar (Figure 3.13 and 3.14) but it can be observed that the flexible units are much more present and used than the slow ones, namely nuclear, lignite and coal. This is easily explained since the capacity ratio is an input of the simulation and has to remain unchanged. So, if the power capacity of the flexible units increases, the one of the slow units has to decrease as per Equation 3.1.

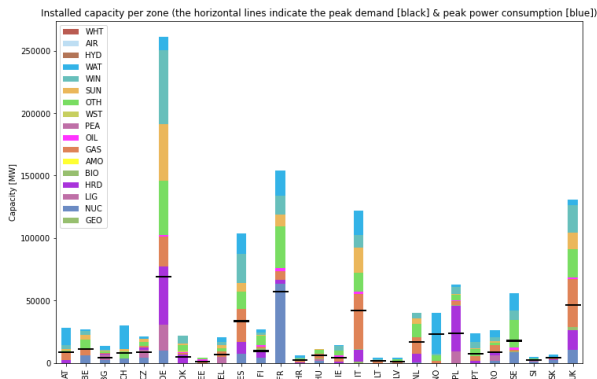


Figure 3.13: Reference case: Installed capacities by zones.

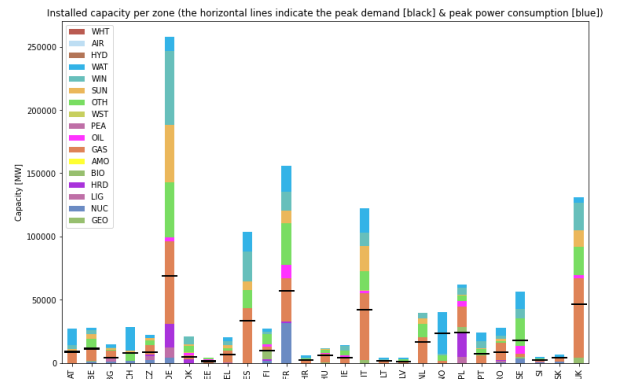


Figure 3.14: Adjusted share of flexible units: Installed capacities by zones.

Taking again the example of Germany in Figure 3.16 and 3.20, it can be seen that the

flexible units, mainly gas, are used intensively to fulfill the demand when the VRES are not present enough and when the slow units are already working at high capacity. One should note that the capacity installed of slow units is low due to the constant capacity ratio and so they are constantly producing energy close to their peak power, making it difficult to adapt their production to meet swift demand changes.

To go one step further, a change in the share of flexibility implies a change in heat generation. Indeed, in these flexible units, combined heat and power can be found. Compared to the reference case where only heat from coal units were generated, now heat from "combined heat and power (CHP) gas units is available in the electrical mix.

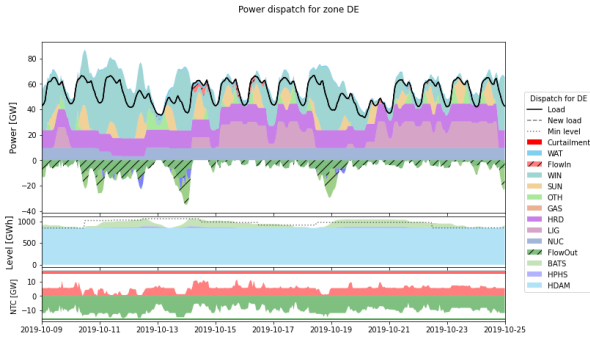


Figure 3.15: Reference case: Germany - Power dispatch.

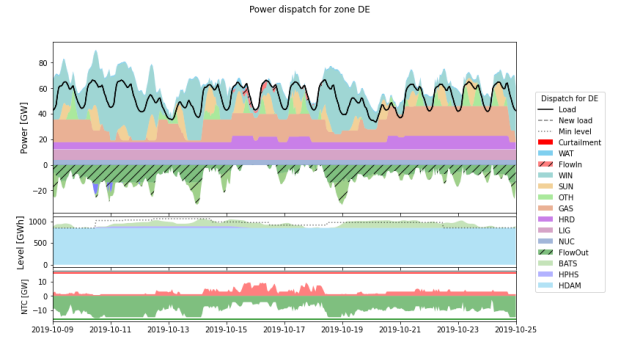


Figure 3.16: Adjusted flexible units: Germany - Power dispatch.

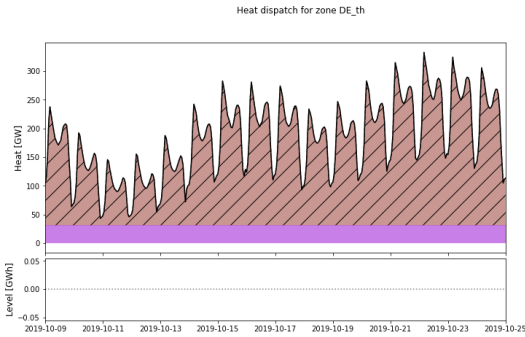


Figure 3.17: Reference case: Germany - Heat dispatch.

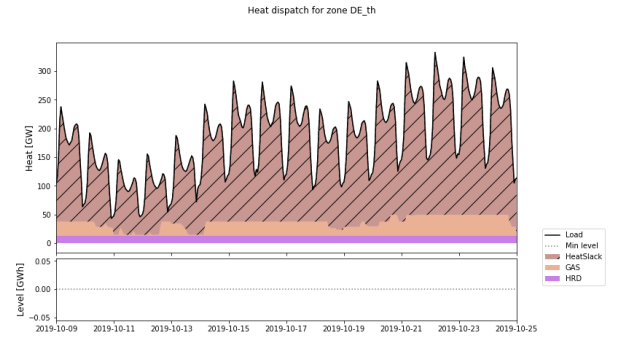


Figure 3.18: Adjusted flexible units: Germany - Heat dispatch.

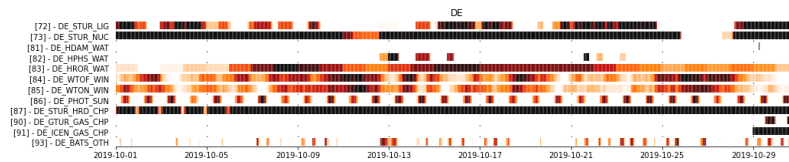


Figure 3.19: Reference case: Germany - Generation Units.

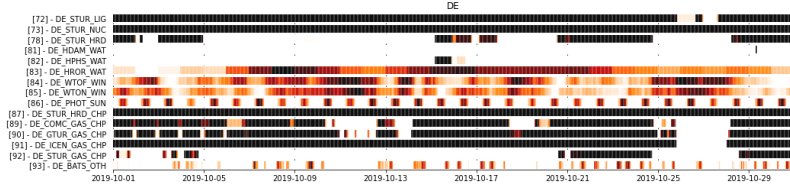


Figure 3.20: Adjusted flexible units: DE - Generation Units.

Generation per zone - CO2 Emissions

Some of the electricity generated in the reference case by the slow units is now generated by the flexible units. Two clusters can be made to explain the evolution of carbon dioxide production in function of the generation per zone (Figure 3.21, 3.22, 3.23 and 3.24).

Taking a closer look at Greece (EL) that sees a great part of its electricity production coming from lignite units that are converted into gas units in the new scenario. Since gas units produce less carbon dioxide than lignite units, a noticeable decrease in carbon dioxide emission can be observed for this country.

An opposite scenario is to look at France where its generation from nuclear decreases in favor of gas units. The latter produce more carbon dioxide than the nuclear field which leads to an increase in carbon dioxide emissions.

As a conclusion, carbon dioxide emissions are heavily depending on unit type and on which kind of fuel is used to generate the electricity.

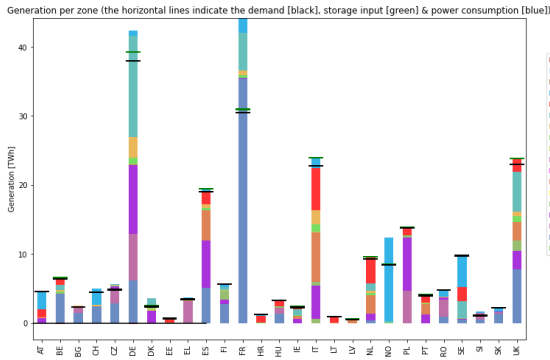


Figure 3.21: Reference case: Generation per zone.

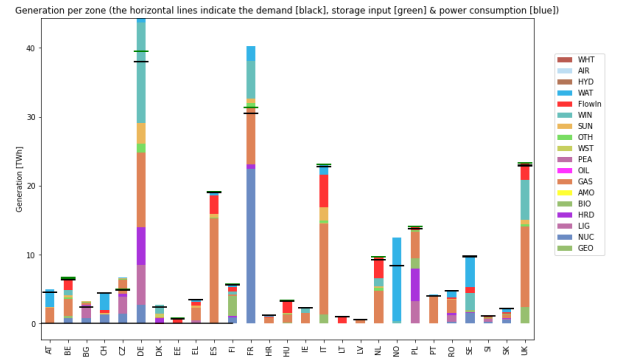


Figure 3.22: Adjusted flexible units: Generation per zone.

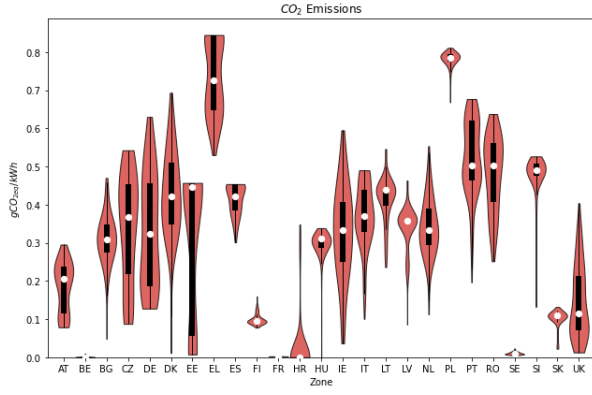


Figure 3.23: Reference case: Carbon dioxide emissions.

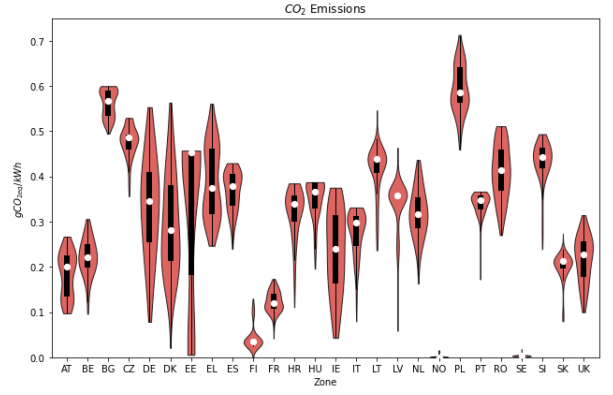


Figure 3.24: Adjusted flexible units: Carbon dioxide emissions.

Conclusions

Increasing the share of flexibility implies in this analysis a decrease in the share of slow units. But, they work in synergy. While the slow units provide a stable and consistent source of energy, if the nuclear example is taken, it cannot adapt quickly to electricity demand changes. But its really low fuel consumption makes it an interesting choice to exploit on the long term. Conversely, flexible units can adapt quickly to fulfill the lack of VRES which makes them also important in the electricity mix.

Regarding carbon dioxide, an increase in the share of flexibility will not lead to a neutral carbon future. The important thing to understand is that flexible units linked to gas produce more carbon dioxide than VRES or nuclear units but much less than slow units using coal and lignite.

To conclude, keeping a good balance between flexible and slow units is the key to a sane and as green as possible electricity network.

3.3.3 Ratio of Net transfer capacity (rNTC)

Sharing more energy amongst big producers and smaller producers, whether it is between zones, regions or countries, has become of high importance. Enabling such exchanges is a key factor to preventing curtailment when a high potential in VRES zone is producing in excess.

What is the influence of the ratio of net transfer capacities doubles?

Generation per zone

The power capacities installed are not influenced by the ratio of net transfer capacity. However, the ration of net transfer capacity can influence countries in the amount of electricity generated and in the choice of the source producing that electricity.

Two major scenarios are emerging from this as a better rNTC leads to an increase in the exchanges of electricty between zones :

1. The production increases in countries with a high production capacity : as it can be seen in Figures 3.25 and 3.26, only France and Germany see their energy generation increase as they have the capacity to do so at a low cost. For instance, in Figure 3.28 France is using all its nuclear power to produce low cost and decarbonized electricity to sell to neighbouring countries.

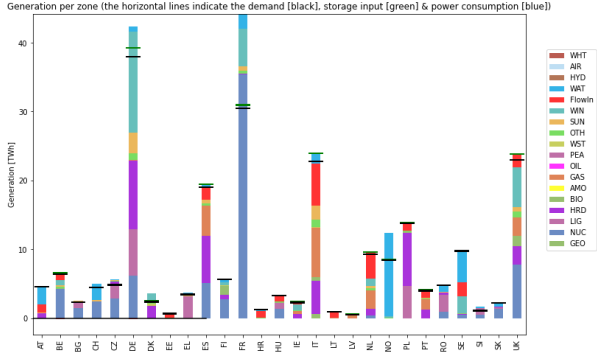


Figure 3.25: Reference case: Generation by zones.

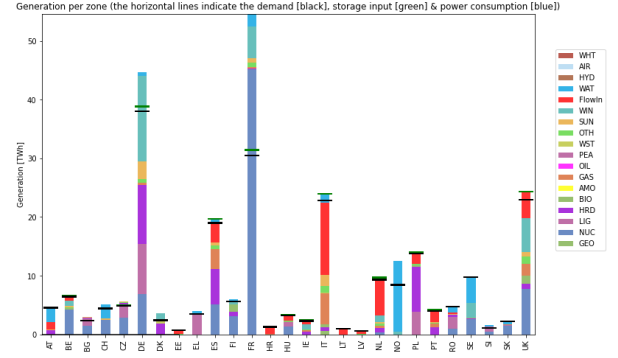


Figure 3.26: Adjusted rNTC: Generation by zones.

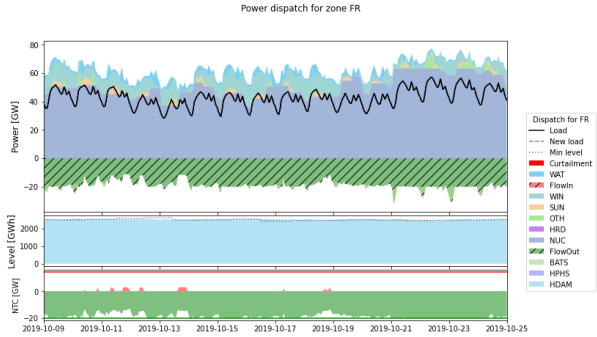


Figure 3.27: Reference case: France - Power dispatch by zones.

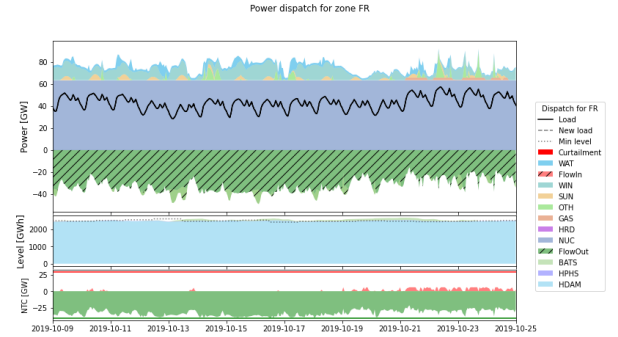


Figure 3.28: Adjusted rNTC: France - Power dispatch by zones.

2. Some countries relying on gas see the gas use drop and replace by electricity coming from other zones : as it is also cheaper to buy greener electricity from another country and due to the high price of the gas, gas units are generating less electricity. That is the case for Italy, Latvia, Netherlands and Portugal as can be seen in Figure 3.29 and 3.30. The share of electricity generated by the gas decrease while the exported energy increase.

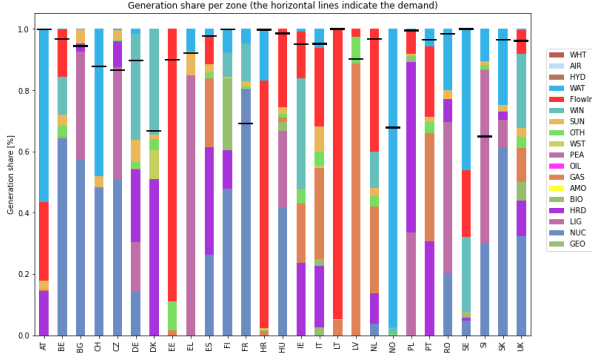


Figure 3.29: Reference case: Generation share per zone.

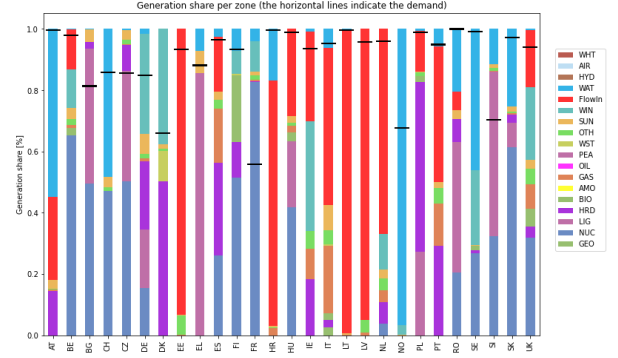


Figure 3.30: Adjusted rNTC : Generation share per zone.

Conclusion

Increasing the ratio of net transfer capacities changes some characteristics of the system. First, the generation locations will be the ones where sources cost less in fuel considering that the total capacity installed does not change. It implies, that some of the locations generating the most electricity will distribute the excess energy produced around the different neighbouring zones, leading to an increase in transfers between zones. In the present case, considering the UE, it can be seen that France and Germany are the "key generation points" and are located in the middle of the other zones which eases the distribution of electricity to the surrounding areas. Then, the system will preferentially diminish the generation coming from gas units due to its high fuel price leading obviously to more transfers to fill the demand.

3.3.4 Share of storage

Energy storage will grow because it is crucial for balancing the intermittent supply of renewable like solar and wind. As these sources increase, storage solutions are needed to store excess energy and provide it when needed, ensuring grid stability. Advances in technology and falling costs make storage more viable, driving its expansion.

What is the influence of the share of storage if it is set to 3.00 leading to a stationary battery power capacity installed of three times the peak demand?

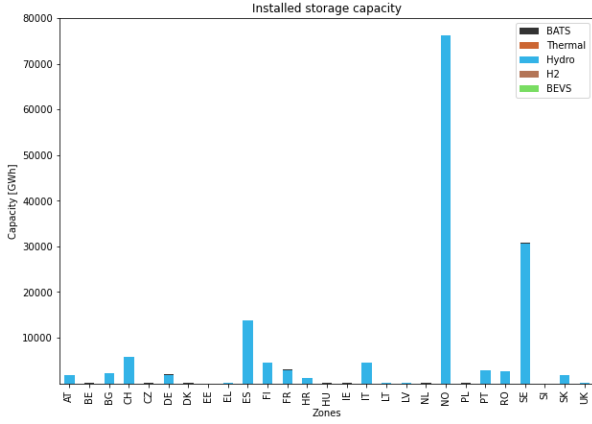


Figure 3.31: Reference case: Storage capacity installed by zones.

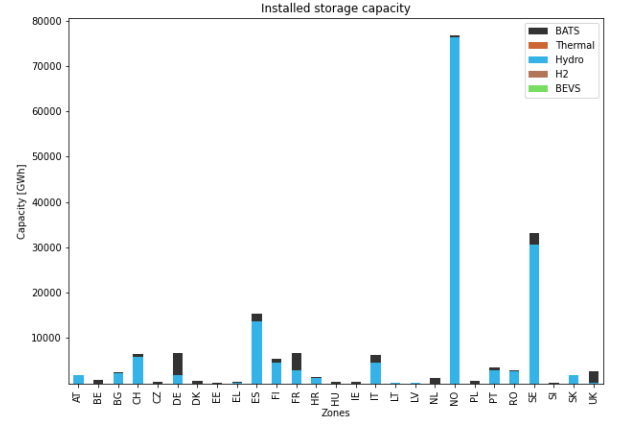


Figure 3.32: Adjusted storage: Storage capacity installed by zones.

Increasing the installed capacity of stationary batteries to three times the peak demand may not have a significant impact compared to hydro dams because the duration time of batteries is substantially shorter. While batteries can quickly respond to high demand with their large installed power, their storage duration is typically limited to a few hours. In contrast, hydro dams have the ability to store energy over much longer periods due to their large reservoirs, allowing them to manage energy supply more effectively across varying demand levels throughout the year. Therefore, even with a high installed capacity, the short discharge duration of batteries means they might not provide the same level of long-term energy storage and stability as hydro dams. That is why, as it can be seen in Figure 3.32, the storage capacity is low compared to hydro-dams.

However, it can be noted that the use of stationary batteries makes sense. These batteries are characterized by a flexible location and adaptable sizing making them more interesting than hydro dams which are restricted to locations with significant elevation. The use of this storage rather than using hydro dams can be observed in the following Figures, comparing power dispatch for Germany and Netherlands with the reference simulation.

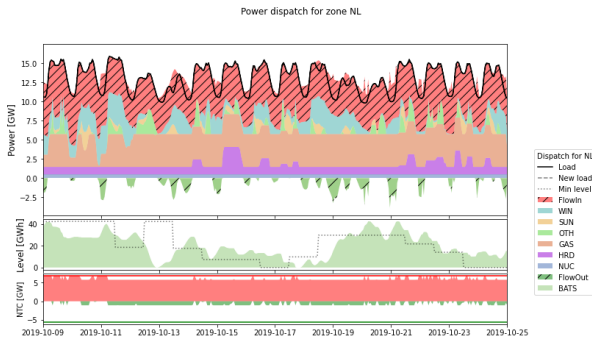


Figure 3.33: Reference case: Netherlands - Power dispatch by zones.

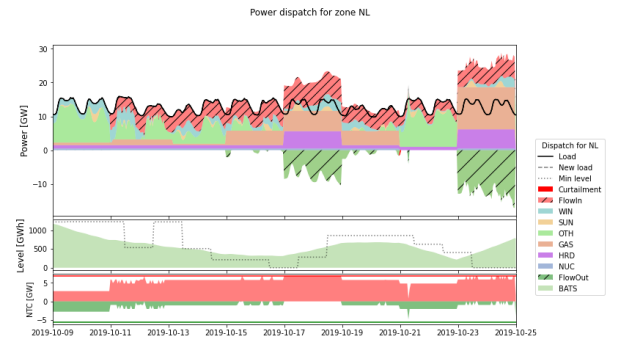


Figure 3.34: Adjusted share of storage: Netherlands - Power dispatch by zones.

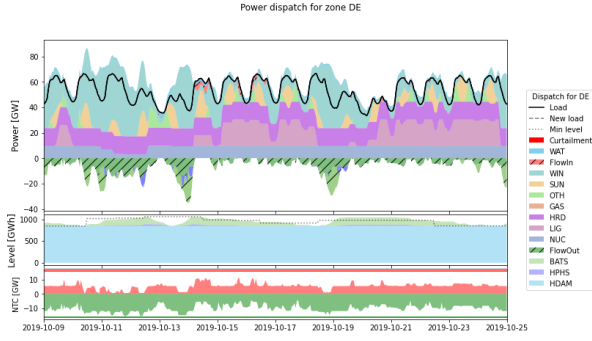


Figure 3.35: Reference case: Germany - Power dispatch by zones.

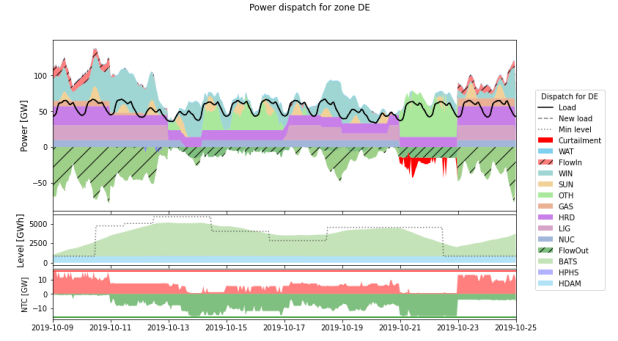


Figure 3.36: Adjusted share of storage: Germany - Power dispatch by zones.

Figure 3.33 and 3.34 are showing how batteries are used to fulfill the demand. Moreover, it can be seen that using this kind of storage is cheaper than importing energy from neighboring countries.

Figure 3.35 and 3.36 are showing another advantage of the stationary batteries. They allow to generate more electricity than the instantaneous demand when storage is not full. Therefore using this stored electricity when the demand is not entirely fulfilled. However, as it can be observed in Figure 3.36, the Dispa-SET constraints show that, sometimes, it is cheaper to discharge the batteries to fulfill the demand and so curtail the excess coming from VRES rather than using slow units and VRES to fill the demand.

The primary goal of stationary batteries is to store energy generated by Variable Renewable Energy Sources (VRES) during peak solar and wind periods to prevent curtailment. This approach ensures that excess renewable energy is not wasted and can be used later when demand is higher or renewable generation is lower. However, this objective seems to conflict with the previous analysis, where Dispa-SET discharges batteries and allows some curtailment to happen.

A hypothesis could be that, in the Dispa-SET model, the cost of curtailment is set to zero. As a result, the model is not incited to minimize curtailment and instead may choose to curtail energy without penalty. Consequently, Dispa-SET might prioritize discharging batteries to meet demand even when curtailment occurs.

Conclusion

While the primary purpose of these batteries is to store excess renewable energy and prevent curtailment, the analysis shows that they also play a key role in reducing reliance on cross-border imports by optimizing local energy use. This finding also highlights the importance of accurately setting the parameters of the simulation, such as the cost of curtailment, to avoid misleading results in the analysis.

3.3.5 Capacity Ratio

The capacity ratio of slow and flexible units is important for ensuring grid stability, supporting the integration of intermittent renewables, and providing operational flexibility. However, as renewable energy sources like wind and solar continue to expand and improve, and as energy storage technologies advance, the proportion of installed capacity from these traditional units is expected to decrease. This shift reflects a transition towards a cleaner and more renewable-centric energy system, with greater reliance on variable renewable sources and advanced storage solutions.

What is the influence of the capacity ratio if divided by 2 ?

Generation share

As the capacity of slow and flexible units are reduced, some of the countries cannot fulfill their demand. Indeed, these two types of units represent a great part of the generation of the total energy demand.

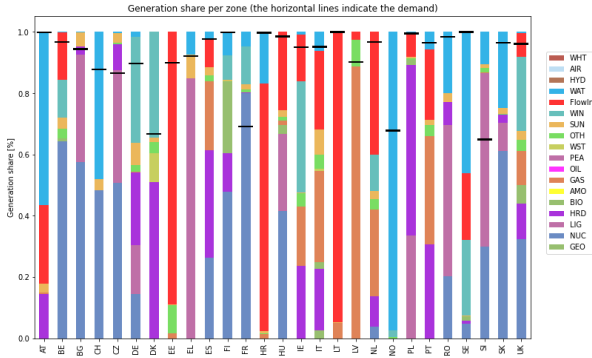


Figure 3.37: Reference case: Generation share by zones.

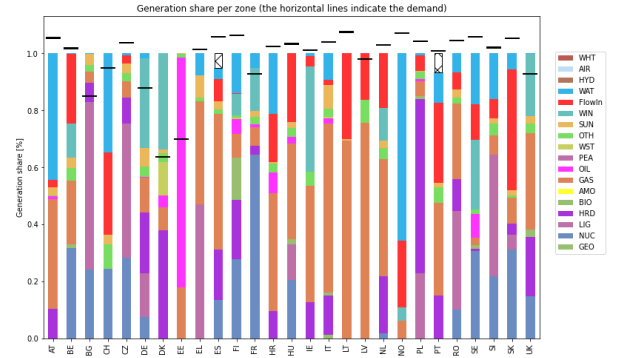


Figure 3.38: Adjusted Capacity Ratio: Generation share by zones.

One can see the trivial apparition of much more gas in the mix. The explanation behind that phenomenon is, that in the reference case, because of the objective cost function, the generation is made depending on the fuel cost. Knowing that nuclear, lignite, and black coal are much less expensive than gas, only the gas units appears in the reference case as a flexible option when needed. In this particular case, where there is a lack of slow unit capacities installed, the gas units are used to fulfill the demand.

Slovakia example

A good example is shown below with Slovakia, in Figure 3.40. Slow and flexible units work at full power, VRES generate electricity when conditions are favorable and the rest of the demand has to be fulfilled by others zones. Flexible units are, in this case, playing the main role of slow units. Indeed, they are used to provide constant and stable generation.

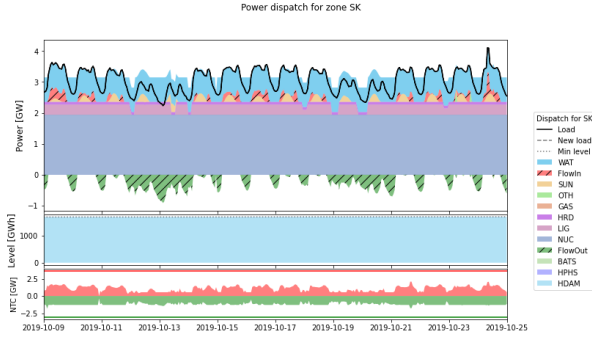


Figure 3.39: Reference case: Slovakia - Power dispatch by zones.

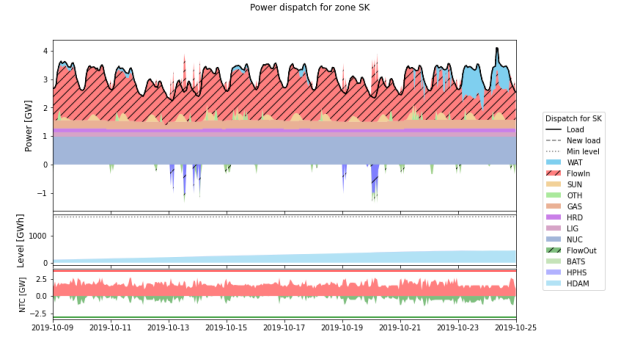


Figure 3.40: Adjusted Capacity Ratio: Slovakia - Power dispatch by zones.

Conclusion

A decrease in the capacity ratio of slow and flexible units typically accompanies an increase in installed wind and solar capacity. This shift is necessary to prevent flexible units, designed primarily for their ability to quickly adjust output and balance the grid, from being forced to take on the role of slow units, which are intended to provide a stable, continuous base load. The gas units being more expensive remain necessary to the energy mix.

4 Sampling - Dispa-SET simulation - Dataset generation

4.1 Overview

In this section, the work conducted is highly related to the conclusions obtained previously by STRAET F. (Computer student engineer at ULiège) in 2023 where he was aiming at creating the dataset.

For instance, following IRENA’s classification, the unit grouping remains unchanged as well as the shape of the design space that remains a Latin hypercube sampling with six dimensions, each one corresponding to one feature of the problem and using an arbitrarily set number of around 3000 points, giving 3.75 points per dimension.

As developed in the previous section, the equations governing those six input parameters have been corrected and modified to account for personal choices and dimension issues. Additionally, some new bounds on the parameters have been implemented using the PRIMES 2050 and the PyPSA-Eur 2050 scenarios.

The generation of the dataset remains mostly unchanged compared to previous work, corrections have been made regarding the use of the adjustment functions developed in Dispa-SET.

4.2 Data preparation and initial parameters

4.2.1 Unit groupings

In this context, the specific technologies and fuel types associated with each power plant are irrelevant since they do not affect the input characteristics of our dataset. With fewer input features than technology-fuel pairs, only the former are significant for training the surrogate model. Therefore, the units are grouped into five categories: flexible units, slow units, storage units, PV units, and wind units.

IRENA’s classification[10] defines flexible units as those capable of rapid ramping, maintaining a low minimum operating level, and exhibiting quick start-up and shutdown times. This distinction is utilized to differentiate between slow and flexible units, with their respective proportions determining the $Share_{flex}$ input. The criteria for this classification are outlined in Table 4.1.

Remark: Due to the number of hydroelectric units being almost at saturation in UE, storage units will take her into account only stationary batteries.

Units	Fuel	Condition
$Flex_{units}$	GAS, HRD, OIL, BIO, LIG,	$PartLoadMin < 0.5$ and $TimeUpMin < 5$ and $RampUpRate > 0.01$
$Slow_{units}$	PEA, NUC, GEO	$PartLoadMin \geq 0.5$ or $TimeUpMin \geq 5$ or $RampUpRate \leq 0.01$

Table 4.1: Classification of flexible and slow units [1].

The three last groups are defined as :

- $Storage_{units}$ with (OTH, BATS)
- PV_{units} with (SUN, PHOT)
- $Wind_{units}$ with either (WIN, WTON) or (WIN, WTOF)

4.2.2 Parameters estimated [1]

The availability factors of PHOT, WTON, and WTOF, along with the peak load, are essential parameters. These values are provided as time-series inputs for the 2019 simulations. To provide a sense of scale, their average values are presented in Table 4.2.

	Value
CF_{LOAD}	0.736 [.]
CF_{PV}	0.13 [.]
CF_{WTON}	0.26 [.]
CF_{WTOF}	0.38 [.]
PeakLoad	441000 MW

Table 4.2: Average values of the capacity factor over all zones, and maximum demand at time t.

4.3 Design space

The design space is crucial for data sampling, as its shape, dimensions, and the chosen sampling strategy directly affect the balance of the dataset, potentially introducing bias.

Regarding the shape of the design space, the analysis made by STREAET F. is considered good as this work does need to meet the same requirements. As a reminder, he showed that using a hypercube with 6 dimensions is the best compromise between the complexity of the input space and the usability of the results as input to another model. As a result, the time, cost and complexity of creating a more general input space is low compared to creating a strictly required input space.

4.3.1 Six dimensionless features of the surrogate model.

The six important features were described in Section 3.1. Three adaptations on these dimensionless features were made.

First, an adaptation for the capacity ratio has been realized to eliminate the dependency between the *Capacity Ratio* and the *Share of storage*. The power capacity of stationary batteries was removed from Eq. 3.1.

Then, the share of wind offshore was added and included in the share of wind feature by making some considerations :

$$Share_{wind-on} = Share_{wind} \cdot C_{onshore} \quad (4.1)$$

$$Share_{wind-off} = Share_{wind} \cdot C_{offshore} \quad (4.2)$$

with $C_{onshore} = 0.7$ and $C_{offshore} = 0.3$, computed with the capacity installed of wind onshore/offshore for 2050 PRIMES Scenarios [9].

Finally, corrections have been made to meet the dimensionless characteristics of the features. The capacity factor of the load was added in the equations related to the share of wind onshore (3.4), wind offshore (3.5), and PV (3.6). It was calculated following the reference simulation for one year :

$$CF_{load} = \frac{\text{Total Annual Energy Demand (MWh)}}{\text{Peak Demand (MW)} \times 8760 \text{ hours}} = 0.736 \quad (4.3)$$

4.3.2 Reference simulation and bounds

A reference simulation is run for the year 2019. This simulation is used as a base to adjust the features and realize all the simulations corresponding to the sampling. The input values for this reference are represented in Table 4.3.

Moreover, bounding the input space is important. That is why research has been made to select properly these bounds for each variable. Two models are used to collect data :

- PRIMES 2050 is a more policy-oriented modeling tool, primarily used to evaluate **the impacts of policy decisions** on the overall energy system of the EU. It considers interactions between different energy sectors and is used to inform policymakers.
- PyPSA-Eur 2050 is particularly useful for academic research and detailed planning of electrical systems, offering great flexibility to test different technological and policy scenarios. It **explores various decarbonization scenarios**.

Considering the different objectives of the models but also the available data and methodology, it is clear that the results will not be the same. That is why most of the data are taken from PRIMES because of its objective to evaluate the impacts of policy decisions. When data is missing, PyPSA-EUR will be used to fill the gaps.

Bounds are defined considering the analysis below. Data and bounds are shown in Table 4.3.

Capacity Ratio

The capacity ratio tends to decrease. Indeed, generation from slow units, i.e fossil fuels, solid fossil fuels, petroleum products, nuclear, decreases while the total demand increases. Considering the definition, the main tendency is a decrease in the capacity ratio from 1.16 in 2019 to 0.74 in 2050. This could be explained by a bigger use of the VRES, and especially storage capacities. The usage of slow units has to decrease to reach carbon neutrality as soon as possible.

The upper bound is set to **1.3** while the lower bound will be lower than for the scenarios and set to **0.4**.

Share of flexibility

It is difficult to predict how the share of flexibility will vary until 2050. Making the calculations with power capacity values gives a similar value to the reference one (around 0.45). This value is obtained considering that all of the current slow units are still defined as slow units. Given the example of small modular reactor (SMR) [11], using nuclear fuel. These kinds of technologies are considered flexible while now in 2024, each of the technologies associated with nuclear are still defined as slow units.

Technologies tend to be more and more flexible, that is why the upper bound is put to a high value: **0.9**² and the lower bound is fixed to : **0.25** considering that technologies tend to develop more and more flexible processes.

Share of PV and Wind

Still using PRIMES scenarios [9], one can see a great increase in the share of PV and wind, as per Table 4.3. Upper bounds are chosen slightly higher than the scenarios values.

Design space for the share of PV is defined between: **0.0** and **0.35**

Design space for the share of Wind is defined between: **0.0** and **0.55**

Share of storage

Data about stationary batteries are also available in the PRIMES scenarios [9]. And can be visualized in Figure 4.1 :

²Considering that for unknown reasons, simulations don't work with a value higher than 0.9

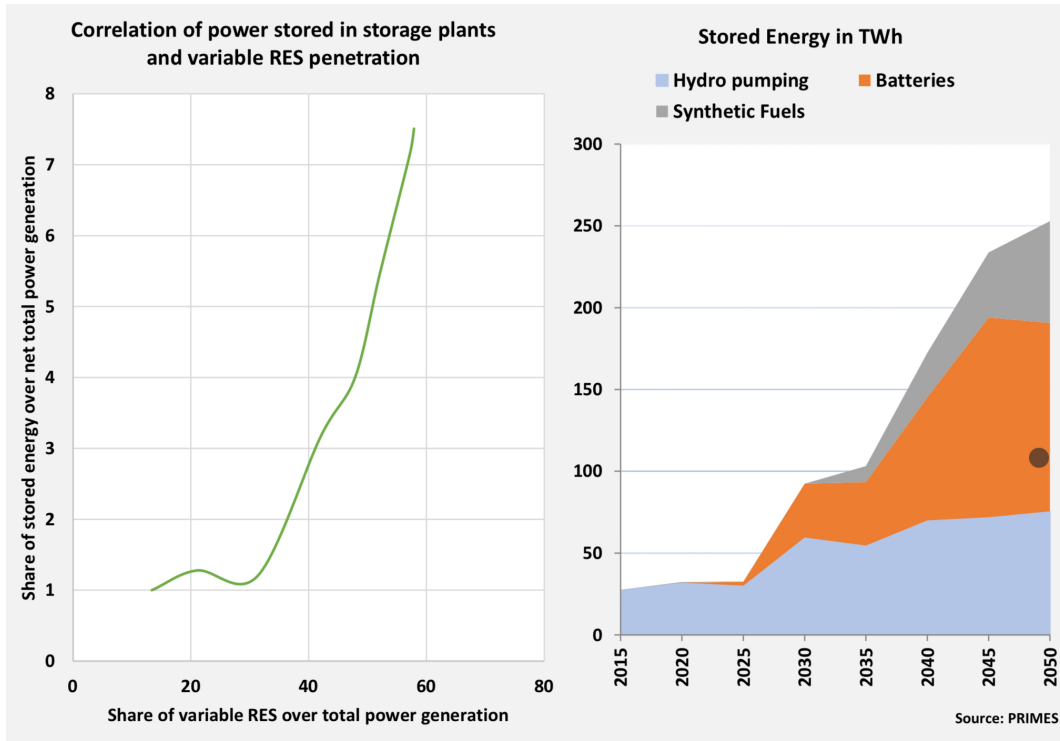


Figure 4.1: Storage plants and energy - PRIMES [9]

For 2050, 110 TWh of stored energy over one year by stationary batteries will be observed, leading to 300000 MWh of storage capacity. Considering a storage duration of 4 hours, it leads to 75GW of power capacity installed for stationary batteries. Which gives us a share of storage of 0.16.

These values could be compared using the Pypsa-EUR scenarios [12] leading to a value of 1 TW of power capacity which gives a share of storage of 2.3.

A third source have been used to select properly the bounds. It says that 1.2 TW should be installed in order to reach carbon neutrality in 2050 (International Energy Agency [13]) leading to a share of storage of 2.6.

It is interesting to observe that projected installed capacity of stationary batteries in 2050 varies between the PRIMES scenarios and the PyPSA-EU model due to their different focuses. It looks trivial that the second model predicts more power capacity with a decarbonization scenario. These differing objectives are resulting in different predictions for battery capacity in 2050.

The upper bound is defined above the maximum value found within the three sources.

That is why lower and upper bounds are defined as : **0.0** and **3.0**

rNTC

Results from Pypsa-EUR [12] are extracted from 2050 scenarios, maximum net transfer capacities between zones, leading to a value of rNTC of 0.55 compared to a value of 0.28 for the reference simulation.

Transfer capacities between European zones will increase to balance the growing share of renewable energy, allowing surplus power from high-generation areas to reach regions with higher demand. This enhances grid stability and supports the integration of renewable sources across Europe.

Design space for the rNTC is defined between: **0.0** and **0.75**, once again a higher upper limit is chosen than for the reference scenario.

	Ref. Values [4]	2050 S.V. ³ [9] [12]	Bounds
Capacity Ratio	1.16	0.74	[0.4 - 1.3]
ShareFlexibility	0.42	0.46	[0.25 - 0.9]
ShareStorage	0.001	2.6	[0.00 - 3.00]
ShareWind⁴	0.2	0.45	[0.0 - 0.55]
SharePV	0.047	0.2	[0.0 - 0.35]
rNTC	0.28	0.55	[0.0 - 0.75]

Table 4.3: Values of the different features of the surrogate model: References - Scenarios - Bounds.

4.3.3 Targets of the surrogate model.[1]

The target outputs of the system are the curtailment and load shedding. To improve scalability, it is required to normalize these targets relative to the maximum possible RES generation, calculated as the sum of availability factors times each unit’s power capacity, and the total demand respectively.

This normalization ensures that the targets remain scalable and applicable across different system sizes. By normalizing, the generalization of the model is also enhanced for diverse future applications.

1. Curtailment

Total energy curtailed to the maximum VRES generation from all units [1]:

$$Curtailment = 100 \times \frac{EnergyCurtailed}{8760 \sum_{units} AF * PowerCap_u} \quad (4.4)$$

Where *EnergyCurtailed* stands in [TWh] for all zones considered. *AF* corresponds to the average capacity factor for one year and *PowerCap* stands for the installed capacities of the units [TW].

2. Load Shedding

It corresponds to the load that can not be fulfilled divided by the total demand [1]:

$$LoadShedding = 100 \times \frac{\sum ShedLoad}{\sum Demand} \quad (4.5)$$

4.4 Generation of the dataset

Before generating the dataset, one has to choose the sample points. As outlined in the previous work, several strategies could be implemented. Such as a "naive" sampling strategy, a Monte-Carlo strategy or the Latin-hypercube strategy. The later one has been chosen as it does not create too much regularities compared to the "naive" one and also avoids creating sample points that are too close to each other, which is more likely to happen when using the Monte-Carlo strategy. That would be equivalent to running multiple times the same simulation, resulting in a waste of computational resources and time.

In addition, based on the reference simulation, key parameters (corresponding to the surrogate model features) are modified following the sampling. To that extent, some functions are available in Dispa-SET. The use of these functions has been corrected.

4.4.1 Adjusting ShareFlex

`adjust_flexibility` is used to modify the capacity installed rate of flexible units compared to the one of slow units to reach the desired $Share_{flex}$.

The script computes the target capacity by multiplying the total capacity by the desired share of flexibility. Then, it adjusts the flexible unit power capacity for each zone. This adjustment is weighted by the total capacity of each zone to ensure a proportional allocation.

4.4.2 Adjusting Capacity Ratio, share of Storage/PV/Wind

`adjust_capacity` applies a linear scaling to the power output of a specified set of units. It can also directly use the new value of capacity installed concerned. This process is particularly useful in adjusting : the capacity ratio, the share of PV, share of storage, share of wind offshore and share of wind onshore.

Several errors were made in the previous work concerning the scale factors and are now corrected as shown in Table 4.4.

Because the capacity ratio was not taken correctly into account in the various simulations, the code display below was implemented.

	Capacity Installed modified	Method	Factor/Value
Capacity Ratio	$PowerCap_{flex+slow}$	Scaling	$\frac{CapacityRatio_{sampling}}{CapacityRatio_{ref}}$
Share PV	$PowerCap_{PVunits}$	New Value	$\frac{Share_{PV} \cdot Peak_{Load} \cdot CF_{load}}{CF_{PV}}$
Share Storage	$PowerCap_{storageunits}$	New Value	$Share_{sto} \cdot Peak_{Load}$
Share WTON	$PowerCap_{wind-on}$	New Value	$\frac{(C_{onshore} \cdot Share_{wind}) \cdot Peak_{Load} \cdot CF_{load}}{CF_{wton}}$
Share WTOF	$PowerCap_{wind-off}$	New Value	$\frac{(C_{offshore} \cdot Share_{wind}) \cdot Peak_{Load} \cdot CF_{load}}{CF_{wtof}}$

Table 4.4: Scaling factors / Values applied to the adjusting functions in order to vary the installed capacities.

```

1 for index in base_units:
2     tech = index.split('_')[1]
3     fuel = index.split('_')[2]
4     tuple_actuel = (tech, fuel)
5     if tuple_actuel not in resultat:
6         resultat.append(tuple_actuel)
7         data = ds.adjust_capacity(data, tuple_actuel,
                                   scaling=(capacity_ratio)/(ref_values['overcapacity']),
                                   singleunit=True)

```

Listing 1: Code used to modify the Capacity Ratio for each of the simulations

`adjust_capacity` takes a tuple as argument to modify the capacities installed. Due to the fact that the capacity ratio implies several combinations of tuples *tech/fuel*, the code implemented has to collect all the tuples linked to flexible and slow units. Finally, it loops over all the tuples to adapt the capacity ratio.

4.4.3 Adjusting rNTC

`adjust_rntc` applies a linear scaling to each zonal NTC time series. Originally, this function was intended to simply replace the value or rNTC by another value given as an argument. Upon deeper look into its source code, it has been discovered that this argument was in fact used as scaling value.

	Method	Factor/Value
rNTC	Scaling	$\frac{rNTC_{sampling}}{rNTC_{ref}}$

Table 4.5: Scaling factor applied to the adjusting function to vary the rNTC.

4.4.4 Extracted outputs and Datsaet features

Table 4.6 is regrouping extracted ouptuts as well as dataset features, taken from [1].

Parameter	Unit	Parameter	Unit
Cost	€/MWh	MaxLoadSheddingShare	%
Congestion	h	CurtailmentToRESGeneration	%
PeakLoad	MW	CF gas	[.]
MaxCurtailment	MW	CF nuc	[.]
MaxLoadShedding	MW	CF wat	[.]
Demand	TWh	CF win	[.]
NetImports	TWh	CF sun	[.]
Curtailment	%	Capacity ratio	[.]
Shedding	%	Share flex	[.]
LostLoad	TWh	Share sto	[.]
MaxRESGeneration	TWh	Share Wind	[.]
TotalGeneration	TWh	Share PV	[.]
ShareRESGeneration	%	rNTC	[.]

Table 4.6: Elements in green correspond to the six features, and elements in blue are the two target outputs.

Some of the outputs are not defined and not explained because they will not be used in this work, but are still present for future working purposes.

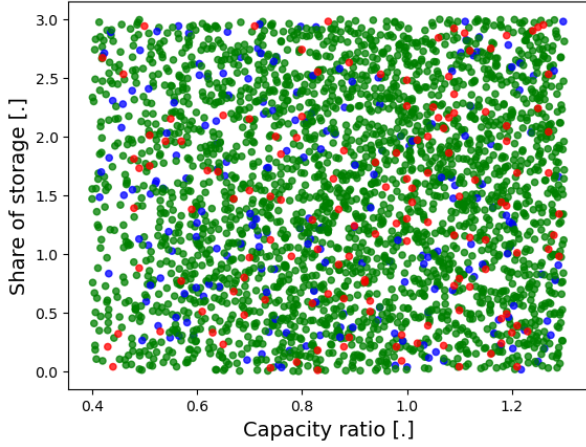
As explained before, the two targets are made dimensionless (See Section 4.3.3).

4.5 Simulations failed

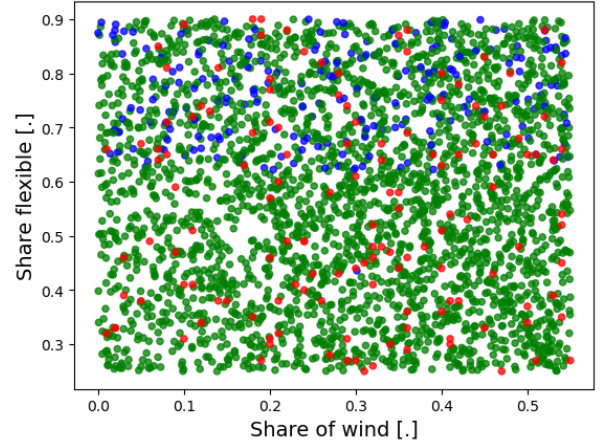
Out of the 3000 simulations launched in series, 2754 have given successful results.

140 simulations are not written in the dataset while 206 simulations ran out of time. Indeed, a time limit was fixed for the GAMS execution based on the mean-time execution of the other simulations. These 206 simulations, no matter the time limit fixed, reach the limit every time.

To better understand the behavior of the unsuccessful simulations, all the samples are displayed on the features space two by two.



(a) Capacity Ratio with share of storage.



(b) Share of flexible with share wind.

Figure 4.2: Simulations sampling of the features. Points displayed in green correspond to successful simulations, in red are the unwritten dataset simulations, in blue are the simulations that reach the time limit.

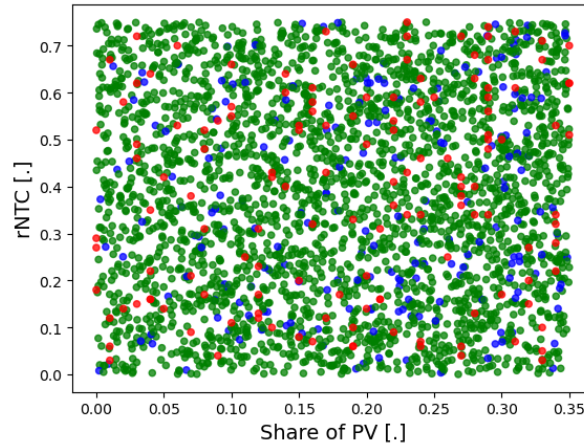


Figure 4.3: rNTC with share of PV. Simulations sampling of the features. Points displayed in green correspond to successful simulations, in red are the unwritten dataset simulations, in blue are the simulations reach the time limit.

First, all the figures are not showing any correlations regarding the unwritten simulations. So, these simulations failed for unknown reasons. On the ULiège cluster, several jobs can be run simultaneously depending on the number of CPUs available at the same time. Due to this limitations, one script written by STREAT F. [1] deals with launching simulations in smaller series. One hypothesis for these failed simulations is that the script responsible for launching the jobs reached its time limit before each of the simulations of the concerned series find a solution. Because, for unknown reasons, all the parallel simulations were not launched exactly at the same time. They were sometimes waiting minutes or hours before launching the following ones. That could explain that some of theses results are not written in the extracted dataset.

In Figure 4.2b, simulations failed have a large share of flexibility, most of them between 0.6 and 0.9. By using 3-dimensional plots, no other results nor correlations are emerging between features. By observing some of the failed simulations in the resulting dataset, some parameters have identical values on all of the simulations. For instance the total generation while some other outputs have complete different values.

The simulations seem to fail each time at the same moment. At this moment the total generation is equal to 119 TWh which corresponds to 4% of the average annual generation for the successful simulations. By checking the results of the simulation, one can find that it stopped on the 16th of January at 5am, i.e. that the generation of the OutputPower time-series dataframe produces only zeros after that moment. This is around this period that the demand reaches its higher values. The reason is still unknown. This is perhaps due to a numerical error or the solver is looking for a dispatch solution but cannot find it until the time limit.

5 The surrogate Models

5.1 Overview

This section aims at building two surrogate models using machine learning methods.

The learning method is **supervised** because it has targets. The two targets are defined in the previous section and are the curtailment and the load shedding. These two targets are processed independently for several reasons. Some of them are briefly described.

First, interdependencies between targets can unnecessarily complicate the model training. Secondly, training models for multiple targets requires more computational resources and complex optimizations. Also, assessing model performance would be more challenging and may result in trade-offs between the targets.

The data inputs processed are quantitative continuous variables. So, one can expect the problem to solve would be a regression problem.

Several steps have to be followed to accurately build the model, this is the purpose of this section. A data preparation phase has to be done, then the learning is separated into training, testing, and validating the data. In the end, the model is built and evaluated by adjusting some hyperparameters.

5.2 Data preparation and correlations

5.2.1 Cleaning

The data are cleaned following these points :

- Checks are made about NaN values for the six features and the two targets.
- Failed simulations are deleted from the dataset.

5.2.2 Correlations

Correlations between features and targets are observed thanks to the `corr()` method from *Pandas* library and collected in Table 5.1:

The correlations computed are used to have a first idea of the correlations between features and targets. Once the models are built, the importance of the features, available in the random forest regressor model, will be used to better understand the behavior of the system and the impact of the different features.

As it can be seen, curtailment looks highly correlated with the share of wind. During peak periods of wind, too much electricity is generated and if curtailment is not applied, it could damage

	Curtailment_ [TWh]	Shedding_ [MWh]
CapacityRatio	-0.09	-0.4
ShareFlex	-0.06	-0.02
ShareStorage	-0.07	-0.04
ShareWind	0.82	0.14
SharePV	0.28	-0.06
rNTC	-0.26	-0.2
Curtailment_ [TWh]	1	0.13
Shedding_ [MWh]	0.13	1

Table 5.1: Correlations between features and targets.

and overload distribution and transmission lines. The share of PV and the rNTC look to be also correlated with the curtailment.

Regarding load shedding, it is inversely proportional to the capacity ratio. Trivially, the capacity ratio is a reflection of the capacities installed for slow/flexible units which represent the major part of the demand management. A too-low capacity ratio, leads to too-low capacities installed of flex/slow units, which leads to the fact that the system cannot reach the demand and one of the solutions is to shed the load. The load shedding looks also slightly correlated with the rNTC. Having bigger capacity to transfer energy between zones can help during high demand period and avoid load shedding.

Further explanations will be developed in the section about the features' importance.

5.3 Scaling and Splitting

5.3.1 Splitting

Splitting is an essential part of the machine-learning process. Here is how the data are split.

- **Train data** : used to train the model by adjusting its parameters to minimize error on these data.
- **Validation data** : used to evaluate the model's performance during training and to adjust hyperparameters, helping to prevent overfitting on the training data. The validation set is derived from the training data.
- **Test data**: used to evaluate the final performance of the model after complete training, providing an estimate of the model's performance on unseen data.

30 % of the data are reserved for the test while the remaining 70% are used for the training. Additionally, 20 % of the training data are also used to validate the model.

5.3.2 Scaling

It is important to scale the inputs and outputs data. Some algorithms, like neural networks, support vector machines (SVMs), and k-means, are sensitive to the scale of data. Without scaling, features with larger ranges could dominate the learning process, leading to biased model performance. Additionally, scaling improves convergence speed during training, particularly for models like neural networks and support vector machines.

To that extend, the function `MinMaxScaler` from the Scikit-learn [14] library was used to scale the data between 0 and 1. This is a normalization technique.

5.4 Model selection

For this kind of problem (Supervised - Regression), several models exist. Two of them, namely MLP Regressor and Random Forest, will be used and compared in order to select which one fits the best.

5.4.1 Multi-layer Perceptron (MLP) Regressor

A Multi-layer Perceptron is a type of artificial neural network, ANN, that is a computational model inspired by the neural structure of the human brain. It consists of interconnected nodes, or artificial neurons, organized into layers. Information flows through these nodes, and during training, the network adjusts the connection strengths, also called weights, to learn from data. This enables the ANN to recognize patterns, make predictions, and solve various tasks in machine learning and artificial intelligence. ANNs can range from a single layer to multiple layers of interconnected neurons.

Three layers are present in the architecture: The input layer, the hidden layer (composed of more than one layer) and the output layer as it can be seen in Figure 5.1:

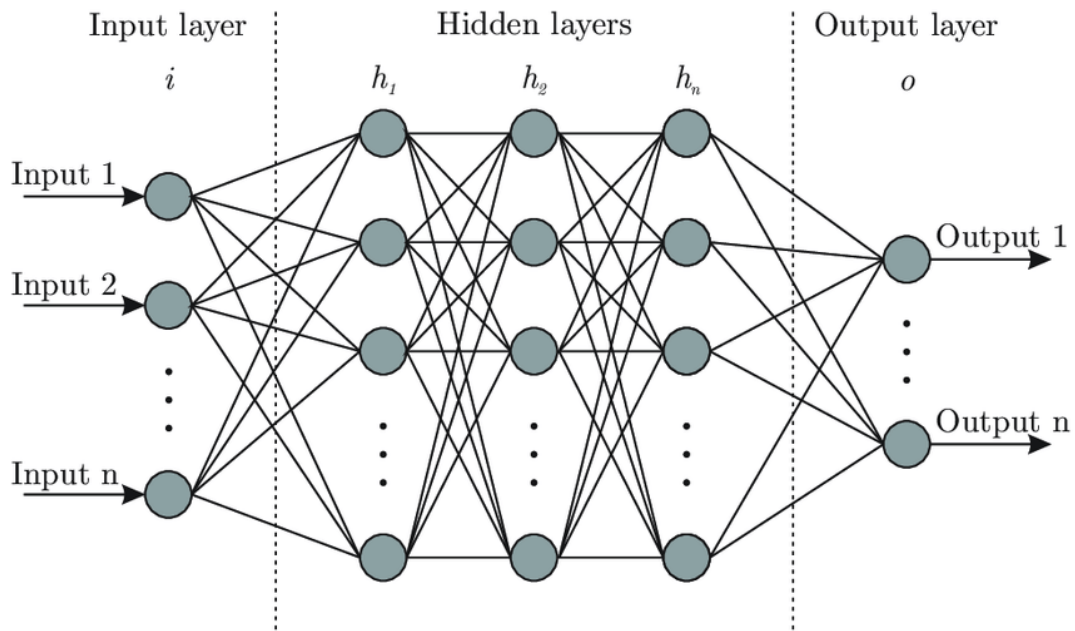


Figure 5.1: Representation of an Artificial Neural Network architecture.

If a zoom is made on one node, or neuron, on the first layer of the hidden layer, this is what can be observed :

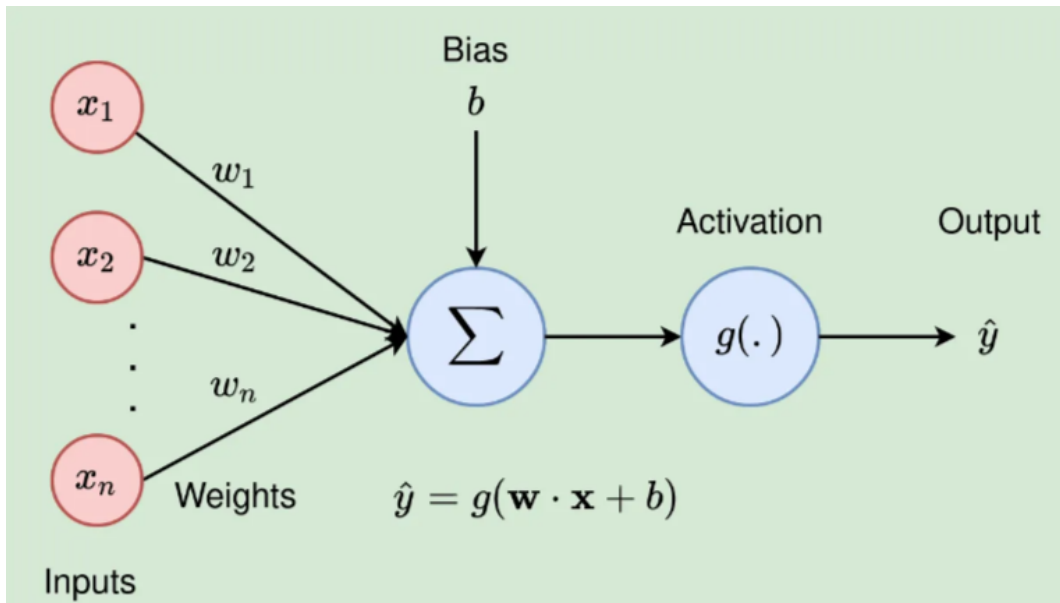


Figure 5.2: Hidden first layer of an Artificial Neural Network architecture.

First, the six features, so there is six neurons, representing the input layer are multiplied by the respective weights and summed. Then, a bias is added to this result as can be seen in Equation 5.1 :

$$\sum_{i=0}^6 w_i x_i + b = \mathbf{w} \cdot \mathbf{x} + b \quad (5.1)$$

Finally, an activation function is applied such as Sigmoid or Tanh function :

$$\hat{y} = g(\mathbf{w} \cdot \mathbf{x} + b) \quad (5.2)$$

Activation functions are crucial for neural networks to work effectively. They allow the network to capture complex relationships, like non-linearity, facilitate gradient propagation for learning, and make informed decisions based on the data. Selecting the appropriate activation function can greatly influence the performance and convergence of a neural network model.

The following step is the output of one neuron is used as the input of the next neuron and so on for each layer until reaching the last layer of the hidden layer where the final prediction is observed.

In summary, the weights and biases are the adjustable parameters of the neural network that determine its behavior. The network's structure, defined by the number of layers and the number of neurons per layer, determines the network's ability to learn and model complex relationships in the data.

5.4.2 MLP - hyperparameters

The hyperparameters are important parameters to tune in order to build the model correctly. The Python library used for building the network is Scikit-learn [14] and references the following hyperparameters in the documentation :

- **hidden_layer_sizes**: This parameter defines the structure of the neural network's hidden layers, crucial for determining the model's complexity and capacity. Each tuple in **hidden_layer_sizes** specifies the number of neurons in each hidden layer. Adjusting this parameter allows for customization of the network's depth and width.
- **activation**: The **activation** parameter dictates the activation function used in the neurons throughout the neural network. It impacts how non-linearities are introduced into the model.
- **solver**: This parameter determines the optimization algorithm used to train the neural network by adjusting its weights. Options such as '**adam**' and '**sgd**' offer distinct approaches to gradient-based optimization, each suitable for different problem domains.

- **alpha**: The **alpha** parameter serves as the regularization term, L2 penalty, applied during training to prevent overfitting by penalizing large weights.
- **learning_rate**: This parameter governs how much the model adjusts its weights in response to the estimated error during training. Choosing between 'constant' and 'adaptive' alters the behavior of the learning rate throughout the training process.

5.4.3 Random Forest

The bootstrapping **Random Forest** algorithm leverages ensemble learning methods within the decision tree framework to generate multiple decision trees from randomly sampled subsets of the data. By averaging the results of these trees, it produces a final output that typically leads to more robust predictions than for only one decision tree :

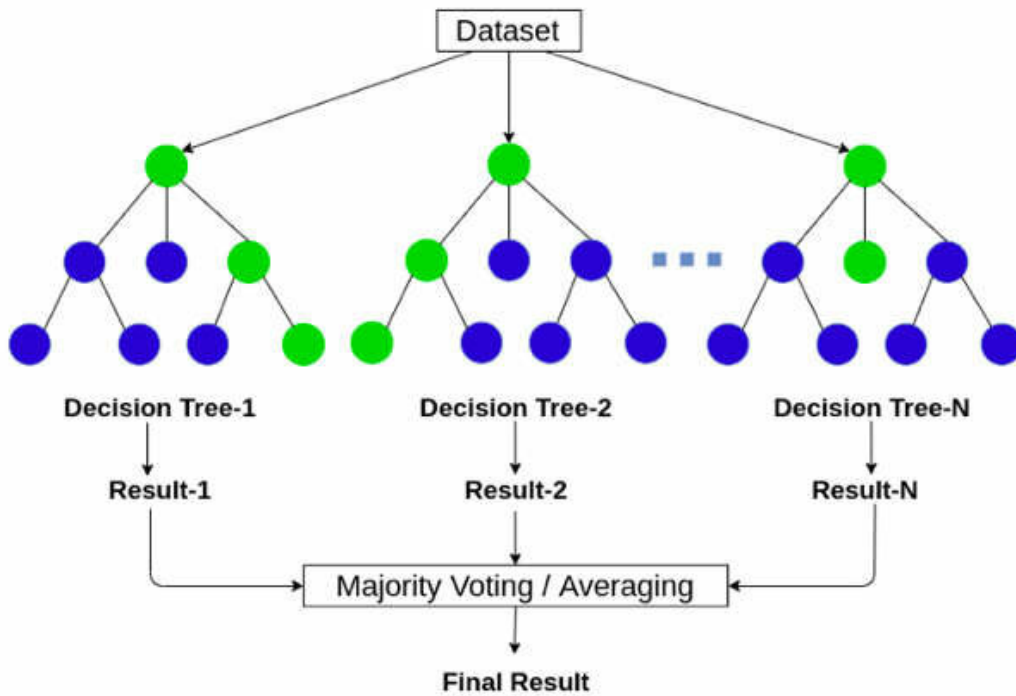


Figure 5.3: Random Forest architecture representation.

A **decision tree** work as represented below :

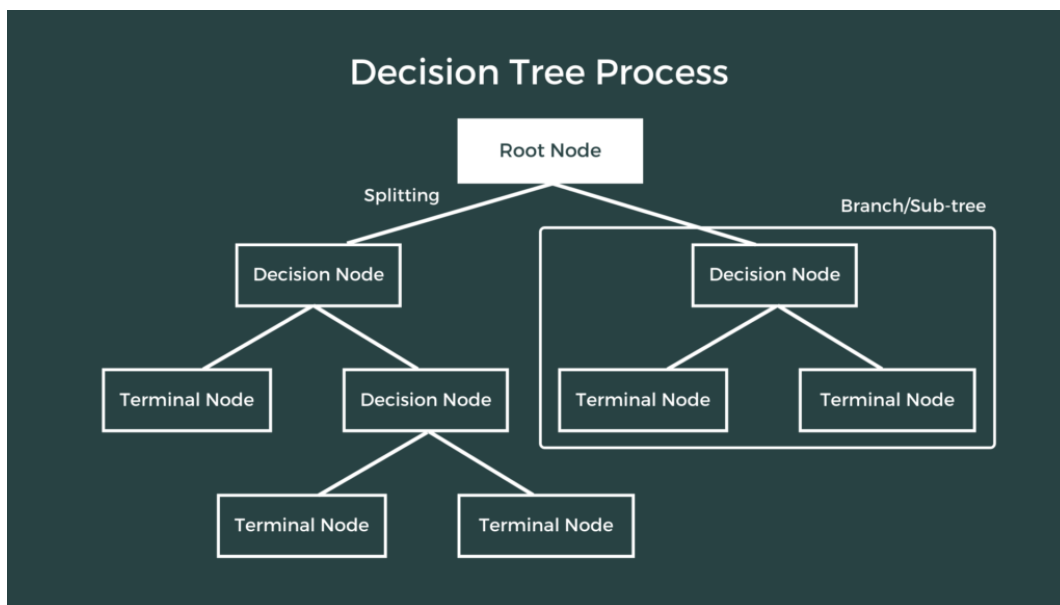


Figure 5.4: Decision tree architecture representation.

First, bootstrapping is processed. It involves randomly sampling subsets of a dataset across multiple iterations, selecting a specified number of variables each time. The results from these iterations are then averaged to achieve a more accurate and robust outcome.

Each of the subsets is used as the base for one decision tree. At each of the decision nodes and root node, one of the variables is used as a condition and, depending if this condition is true or false, the algorithm goes to one or the other branch and so on.. until reaching a terminal node where an estimator of the output is given.

For understanding purposes, an example is shown in Figure 5.5.

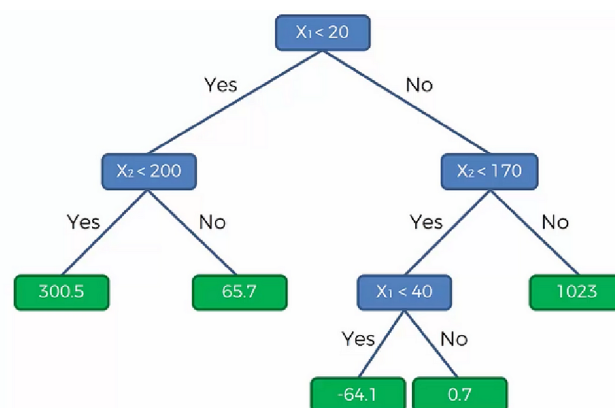


Figure 5.5: Decision tree architecture example.

5.4.4 RF - hyperparameters

Here are the important hyper-parameters to evaluate to build the model correctly, also coming from the documentation of Scikit-learn [14] :

- **n_estimators**: This parameter determines the number of trees in the forest. Increasing the number of estimators generally improves the performance of the model.
- **max_features**: Specifies the number of features to consider when looking for the best split. 'auto', 'sqrt', 'log2', or an integer value can be chosen. This parameter impacts the diversity and performance of each tree in the forest.
- **max_depth**: Controls the maximum depth of each tree in the forest. Deeper trees can model more complex relationships in the data but are more prone to overfitting.
- **min_samples_split**: The minimum number of samples required to split an internal node. Higher values prevent the model from learning overly specific patterns, potentially improving generalization.
- **min_samples_leaf**: The minimum number of samples required to be at a leaf node. This parameter controls the size of the leaves and impacts the complexity of the model.
- **bootstrap**: Whether bootstrap samples are used when building trees. If set to **True**, each tree is built on a bootstrap sample of the training data, introducing randomness and improving generalization.

5.4.5 Comparison between RF and ANN

In order to understand better which of the two algorithms could suit the best, here are collected characteristics of them previously described:

	Artificial Neural Networks (ANN)	Random Forests (RF)
Linearity	Capable of learning complex non-linear relationships	Naturally handles both linear and non-linear relationships
Data Requirements	Requires a large amount of data	Can work well with smaller datasets
Computational Power	Computationally intensive	Can become computationally expensive with a large number of trees
Training Time	Longer due to more complex calculations	Shorter due to parallel training
Prediction Time	Quick predictions after training	Slow real-time prediction
Overfitting	Prone to overfitting with small datasets	Resistant to overfitting due to ensemble method
Feature Selection		Can identify the most significant features

Table 5.2: Comparative Advantages and Disadvantages of Artificial Neural Networks and Random Forests for Regression.

5.5 Model Building

As explained above, two models are built, one for each output: Curtailment and Load Shedding.

First, the best combination of hyperparameters defined previously is searched for each of the two algorithms : MLP and RF. This is done thanks to the `GridSearchCV` function. It uses a score metric with a KFold ($K = 10$) cross-validation with the training fold.

The 10-fold cross-validation is a technique used to assess machine learning model performance with limited data. It involves splitting the data into subsets, training and testing the model multiple times, and then averaging the results. This approach provides a more reliable estimate of how well the model will perform on new, unseen data compared to a single train/test split. Visual representation is shown in Figure 5.6.

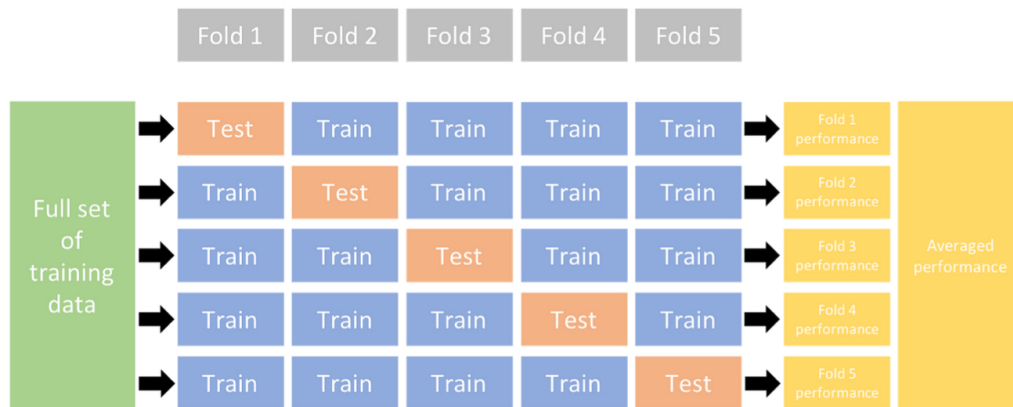


Figure 5.6: K-Fold Cross Validation representation.

Then, once the hyperparameters are selected for each of the two algorithms, they are compared to each other to know which one has the best validation score, i.e. the lowest mean absolute error, the lowest mean squared error or the higher r2 score. Comparing boxplots and hyper-parameters values are represented in the next sections.

Finally, the models are evaluated on unseen data, test fold, to test the model's ability to generalize beyond the data it was trained on.

5.6 Curtailment - Hyperparameters and models comparison

The search for the best hyperparameters for the two models on the curtailment is yielding the following results :

	Values
n_estimators	500
max_depth	None
min_samples_split	2
min_samples_leaf	1
max_features	None
bootstrap	True

Table 5.3: Curtailment RF Hyperparameters

	Values
hidden_layer_sizes	(100, 50)
activation	relu
solver	adam
alpha	0.0001
learning_rate	constant

Table 5.4: Curtailment MLP Hyperparameters

Here are the cross-validation results :

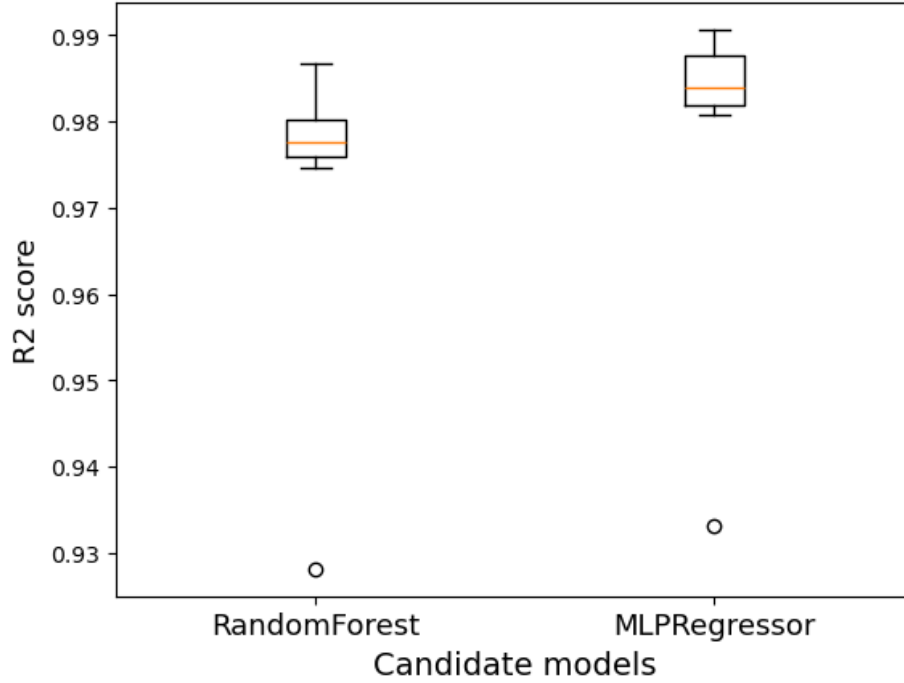


Figure 5.7: Performance evaluation by a cross-validation. Comparison between RF and MLP for R2 score metric - Curtailment target.

Both scores observed in Figure 5.7 are looking good. A slightly better score for the MLP algorithm is observed. The two other boxplots with the metrics MAE and RMSE are also generated and the same conclusions are drawn thus those Figures are omitted, the score obtained are still available in Table 5.5.

5.7 Curtailment - Results

Testing the models on unseen data is essential to evaluate its ability to extend to new data. This process helps detect overfitting, where the model might perform well on training data but poorly on unseen data. The performance metrics are presented in the following table :

	RF Regressor	MLP Regressor
Root Mean Squared Error (RMSE)	0.034	0.029
R2 Score	0.978	0.985
Mean Absolute Error (MAE)	0.023	0.021

Table 5.5: Performance metrics for RF Regressor and MLP Regressor - Test fold on the curtailment target.

A first observation is that the test set score is really close to the training set score with cross validation. Considering them good, it is a first indication of good generalization.

Other representative plots are shown in Figure 5.8 and 5.9 :

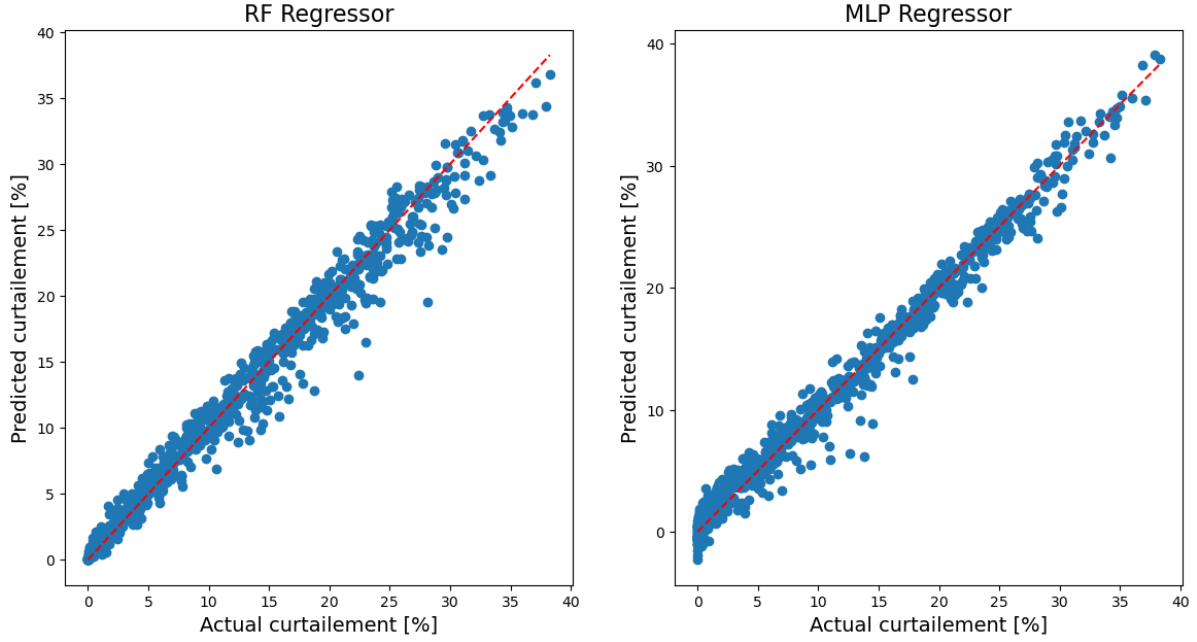


Figure 5.8: Scatter plots comparison between the two models for the curtailment target.

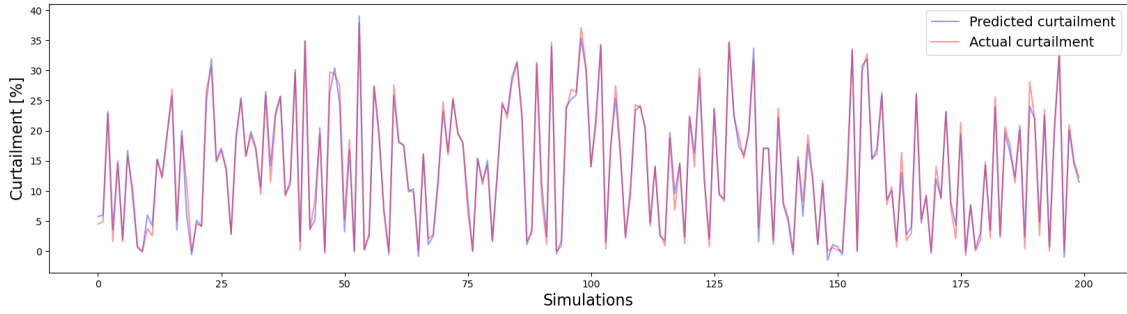


Figure 5.9: 200 first points on test fold for the MLP regressor model - Curtailment target.

In figure 5.8, one can observe that the MLP model predict sometimes slightly negatives values for curtailment while RF model not. This is due to the way the two model perform the regression.

The MLP regressor can produce any continuous value, including negative values, depending on the weights learned by the neurons to minimize a loss function. The RF regressor generally produces non-negative outputs. Indeed, its predictions are averages of decision tree outputs, which makes negative values less likely, particularly if they are not present in the training data.

As one can observe, the MLP model seems to have almost excellent predictions as previously explained. Nevertheless, the RF algorithm remains good on unseen data and due to its characteristics, it will be used to obtain information about features' importance in the next sections.

5.7.1 Execution Times

In Table 5.6, prediction and training time for the curtailment target are displayed :

	RF Regressor	MLP Regressor
Training Time [s]	57	2.7
Prediction Time [s]	0.235	0.036

Table 5.6: Training and Prediction Time Comparison between RF Regressor and MLP Regressor for curtailment target.

The training time for the MLP is significantly shorter than that for the RF, even though RF operates in parallel. This is due to the fact that the Random Forest model uses 500 trees, each of which must be individually trained, introducing considerable computational overhead despite the parallelization. In contrast, the MLP model has a simpler structure with only two layers of 100 and 50 neurons, requiring less computation and leading to a much shorter training time.

Moreover, the prediction time for the MLP is also shorter than that for the RF, which is expected. In RF, each prediction requires aggregating results from all 500 trees, which increases the time needed to generate predictions. The MLP, with its smaller and more straightforward structure, can produce predictions more quickly, further contributing to its efficiency in this scenario.

5.7.2 Learning Curve

Learning curves [15] show the training and validation loss as more training examples are gradually added. They help us determine whether adding more training examples would improve the validation score, i.e. the performance on unseen data. If a model generates overfitting, adding more training examples might enhance its performance on unseen data. Conversely, if a model is underfitting, adding more training examples is unlikely to help.

- **Overfitting** : happens when the model performs well on the training set but no so well on unseen data [16].
- **Underfitting** : happens when it neither performs well on the train set and the test set [16].

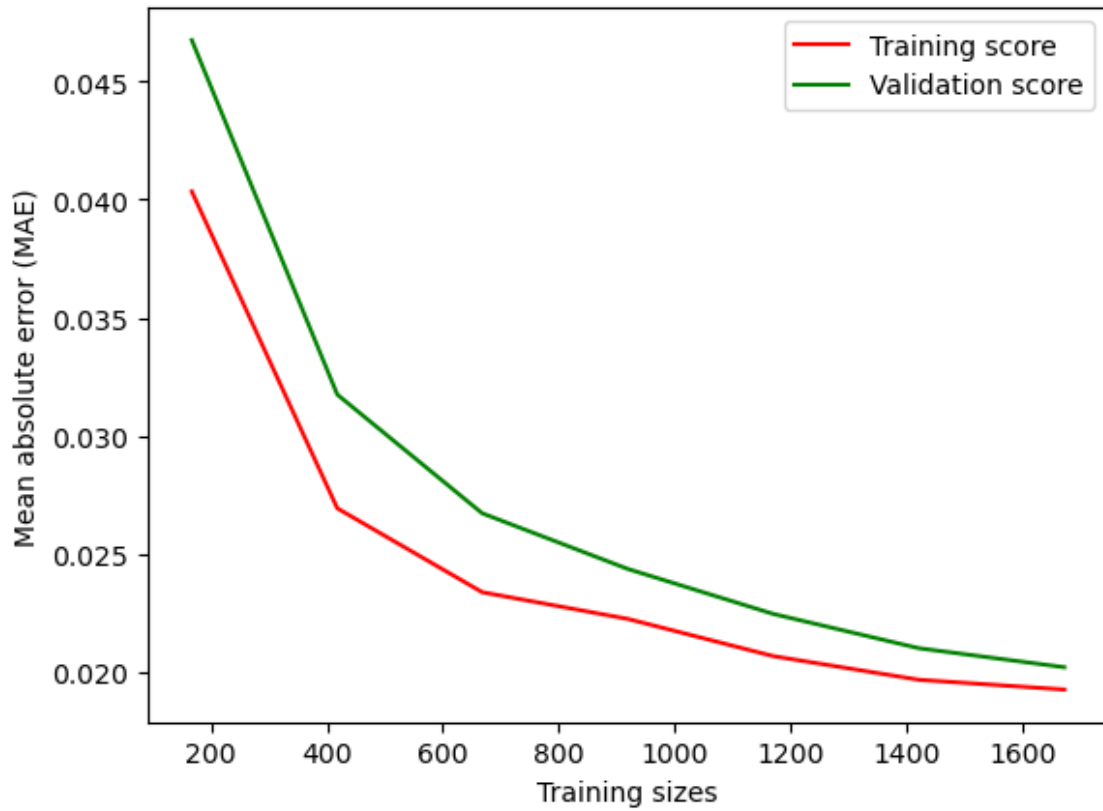


Figure 5.10: Learning curves on validation and training set - MLP Regressor model - Curtailment target.

Figure 5.10 represents the learning curves of the MLP model.

A well-fitted model's learning curve typically shows a moderately high training loss initially, which decreases and eventually flattens as more training examples are added, indicating that adding more data no longer improves performance on the training set. Similarly, the validation loss starts higher, decreases with more data, and then flattens, showing that additional training examples don't further enhance performance on unseen data. It is common to say that the validation loss is slightly greater than the training loss.

This analysis indicates that the model does generalizations well.

5.7.3 Loss curves

The loss curve is a crucial tool in machine learning for diagnosing overfitting and determining the optimal number of iterations for model training. By plotting the loss values over epochs, it allows you to observe how the model's performance on both training and validation datasets evolves. Persistent divergence between training and validation loss indicates overfitting, while a stable or converging curve suggests effective learning. Additionally, analyzing the curve helps in determining the maximum number of iterations needed, as it shows whether the model is still

improving or if it has reached convergence, guiding decisions on when to stop training to avoid unnecessary computation and potential overfitting.

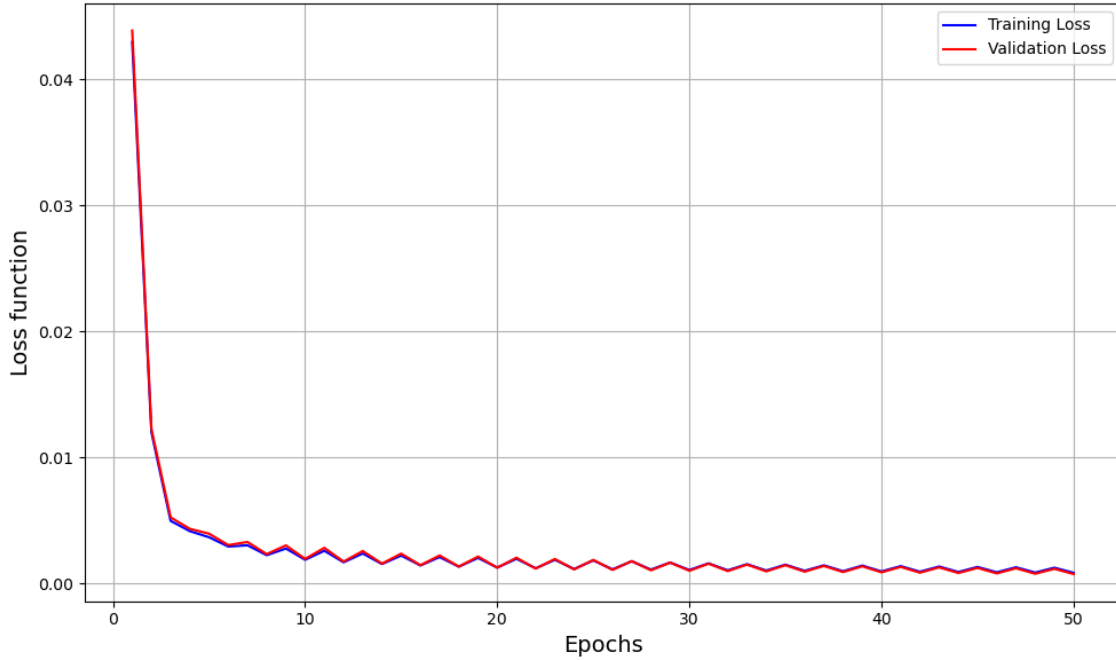


Figure 5.11: Loss curves on validation and training set - MLP Regressor model - Curtailment target.

As can be observed, the curves look to converge correctly, indicating that no overfitting occurs. Moreover, it shows that the maximum number of iterations can be limited to 50.

5.7.4 Feature importances

An interesting method implemented in the RF regression model is the feature importance. It reflects how much a feature reduces the variance in the dataset when used for splitting nodes, averaged across all trees in the forest. Feature importances are represented in Figure 5.12.

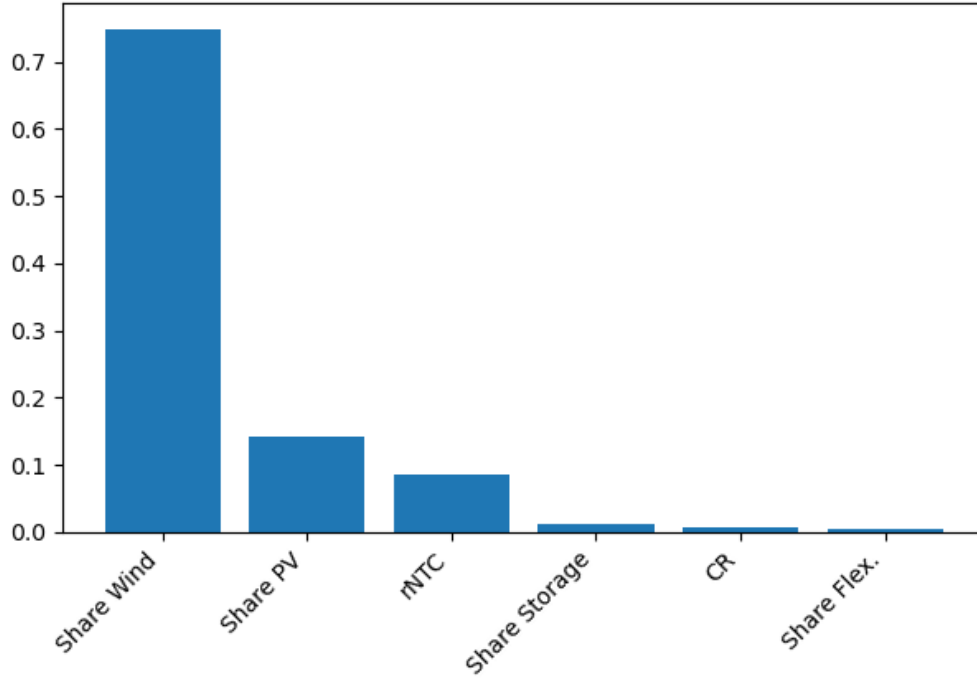


Figure 5.12: Features importances - RF Regressor Model - Curtailment target.

As previously observed, the curtailment is mainly driven by the share of wind. But also with the share of PV and the rNTC. One question is : "why does the share of wind influence more the curtailment than the share of PV despite the fact that they are both VRES ?"

Several reasons could explain the phenomenon. Wind turbines are often curtailed more than solar panels due to differences in variability and predictability. Wind energy production is more variable and less predictable, complicating grid management. While solar production is more consistent and produce energy only during the day, and so only during peak demand period. Additionally, wind turbines can generate electricity 24/7, including at night when demand is lower, which can lead to overproduction and grid overload, necessitating curtailment. To conclude, wind turbines offer greater control flexibility simply by stopping some of the them or by adjusting the blade angle, making curtailment easier to implement compared to solar systems.

A high rNTC leads to more exchange between zones which automatically leads to less energy curtailed. Developing or increasing it, is important to avoid spilling energy.

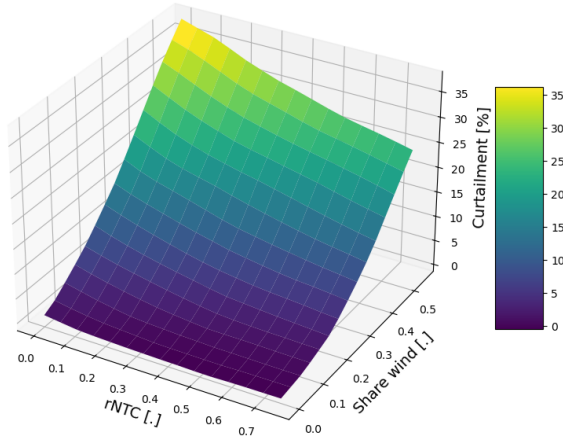
These results can be used as a base to observe the influence of some features on the target and will be developed more in details in the next section.

5.7.5 Surface plots - Curtailement

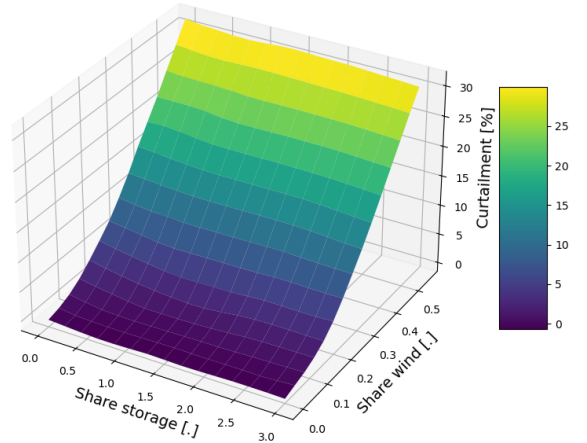
By keeping the best model, namely the MLP, and better visualizing the behavior of the output, some of the features are frozen, i.e. equal to the reference case values computed in Table

4.3, while two of them vary ⁵.

Due to the six features used for the surrogate model, 15 plots can be displayed. For understanding purposes and to avoid redundancies, only four are displayed. First, and based on the feature importance, the two most important features are used. Then, the last three are build on making one important feature vary and compared against one that is considered less important. The four figures : 5.13a, 5.13b, 5.14a and 5.14b are shown below :

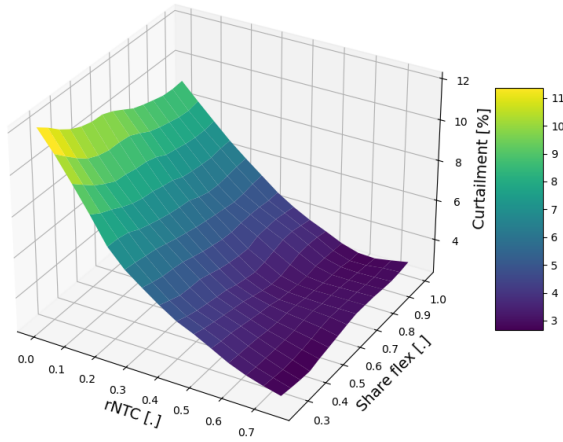


(a) rNTC with share of wind.

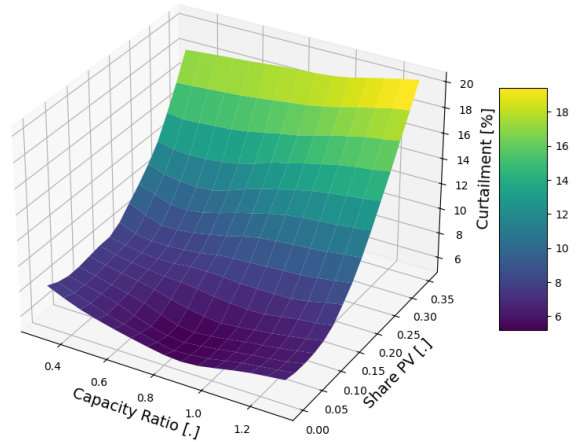


(b) Share of storage with share of wind.

Figure 5.13: Results of the surrogate model for the curtailment target - Surface plots with four constant and two varying features.



(a) rNTC with share of flexibility.



(b) Capacity ratio with share of PV.

Figure 5.14: Results of the surrogate model for the curtailment target - Surface plots with four constant and two varying features.

⁵In the following plots, all the features are re-scaled in the defining bounds of each features for understanding purposes

As it can be observed on the four surface plots, three features impact significantly the percentage of curtailment generated by year : The capacity installed of wind turbines, the net transfer capacities between zones and the capacity installed of photovoltaic panels.

It shows also that the three other features have a much lower impact on the results. Plotting them each with a more important feature show everytime smooth surface plot increasing or decreasing with the more important one.

One can also observe that the share of wind is the most sensible feature. A slightly increase will lead to a bigger increase in the curtailment.

Figure 5.13a show that if the capacity installed of wind turbines increases over the year, the share of electricity between zones in Europe have to increase also to reduce the curtailment and so the rNTC does. Doubling the rNTC for high share of wind could lower the curtailment by 10%, which is non-negligible.

Finally, in Figure 5.13b, it is observed that the share of storage has almost no influence on the prediction of the curtailment compared to the share of wind. After analyzing the configuration files and some of the simulations, a hypothesis is that the issue might be related to the cost associated with curtailment. Indeed, setting the curtailment cost to zero leads to the misconception that the installed capacity of stationary batteries is not connected to the amount of curtailment in the surrogate model. Because the model does not penalize curtailment, it does not reflect the true impact that an increase in battery capacity could have in reducing curtailment. As a result, the surrogate model may incorrectly suggest that adding more batteries has little effect on curtailment, even though, in reality, their primary purpose is to store excess renewable energy and reduce curtailment.

5.8 Load Shedding - Hyperparameters and models comparison

Other hyperparameters for the load shedding were found from GridSearchCV and are shown in Table 5.7.

	Values
n_estimators	50
max_depth	10
min_samples_split	2
min_samples_leaf	2
max_features	None
bootstrap	True

Table 5.7: Load shedding RF Hyperparameters.

	Values
hidden_layer_sizes	(200, 100, 100, 50)
activation	relu
solver	adam
alpha	0.0001
learning_rate	constant

Table 5.8: Load shedding MLP Hyperparameters.

After analyzing the dataset, more than 79% of the simulations are yielding a load shedding

value of zero while the rest is giving very different positives values. In this case, the choice of the metric selection is important.

Considering the definition of the r^2 score :

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y})^2} \quad (5.3)$$

A high R^2 score might be misleading if it mostly reflects how well the model performs on the common zero values, rather than showing how well it handles the rare but important high values. In other words, a high R^2 score can give a false impression of good model performance if the model isn't actually doing a good job with the most significant high values.

The choice of the RMSE metric looks more sensible. Large errors are penalized due to the square imply in its formula. And it will be seen in the next section that larger errors happen for non-zero values, values that have more interests.

Cross-validation results can be observed between the two algorithms for the RMSE and MAE metric :

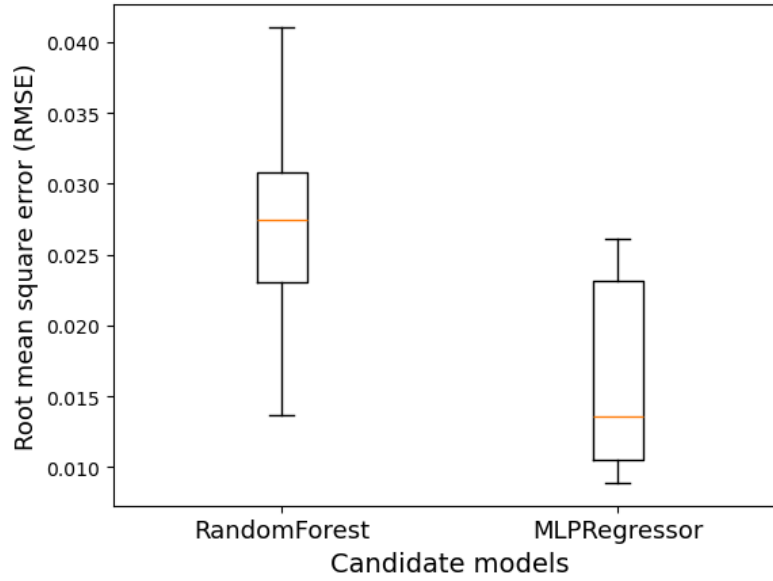


Figure 5.15: Comparison between RF and MLP for RMSE metric - Load shedding target.

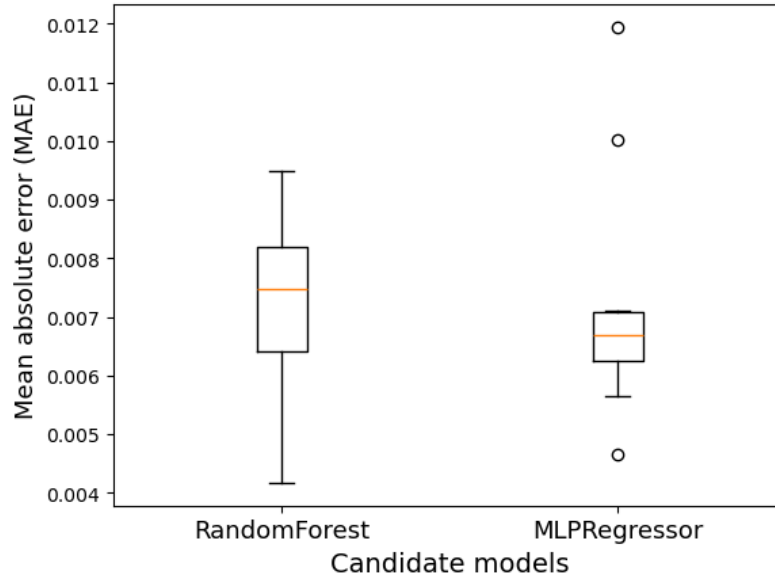


Figure 5.16: Comparison between RF and MLP for MAE metric - Load shedding target.

Figures 5.15 and 5.16 are both showing that the MLP regressor have better performance in terms of cross-validation results.

5.9 Load Shedding - Results

On unseen test fold, error metrics gives :

	RF Regressor	MLP Regressor
Root Mean Squared Error (RMSE)	0.034	0.016
Mean Absolute Error (MAE)	0.009	0.007

Table 5.9: Performance metrics for RF Regressor and MLP Regressor - Test fold on the load shedding target.

Results on unseen data, presented in Table 5.12, are showing that the MLP regressor performs better than the RF regressor. RMSE on the test fold for the MLP regressor is close, slightly higher, to the training set, indicating a good generalization.

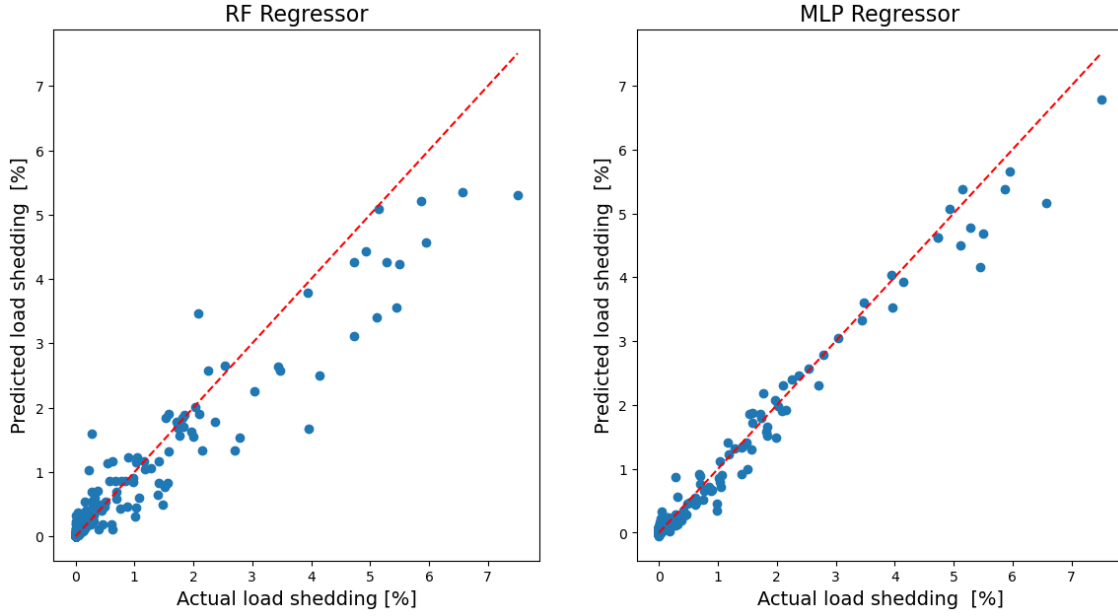


Figure 5.17: Scatter plots comparison between RF regressor and MLP regressor for the load shedding target.

Figure 5.17 shows how the MLP regressor performs better than the RF model.

However, in both cases, higher values are under-estimated. The system clearly faces an imbalance class. The definition given in [16] indicate that : "class imbalance occurs when observations in one class are higher than the observations in the other classes. The model will predict the class with fewer observations worse because it has fewer instances to train on".

Different tactics can be used to improve the model. Two of them have been used in this work : Under-sampling and Over-sampling

- **Under-sampling** : removal of majority class data.
- **Over-sampling** : duplication of minority class data.

Theses two methods have been implemented but only one of them was conclusive, the over-sampling, and will be developed in Section 5.10.

5.9.1 Execution Times

In Table 5.10, prediction and training time for the load shedding target are displayed :

	RF Regressor	MLP Regressor
Training Time [s]	6.24	6.27
Prediction Time [s]	0.014	0.063

Table 5.10: Training and Prediction Time Comparison between RF Regressor and MLP Regressor for load shedding target.

As observed, training and prediction times look similar this time. This is due to the fact that the number of trees is considerably reduced compared to the analysis made with the curtailment target. From 500 down to 50 trees only.

5.9.2 Feature importances

The RF Regressor model is still used for the feature importances even though it looks to perform not so well as the MLP regressor :

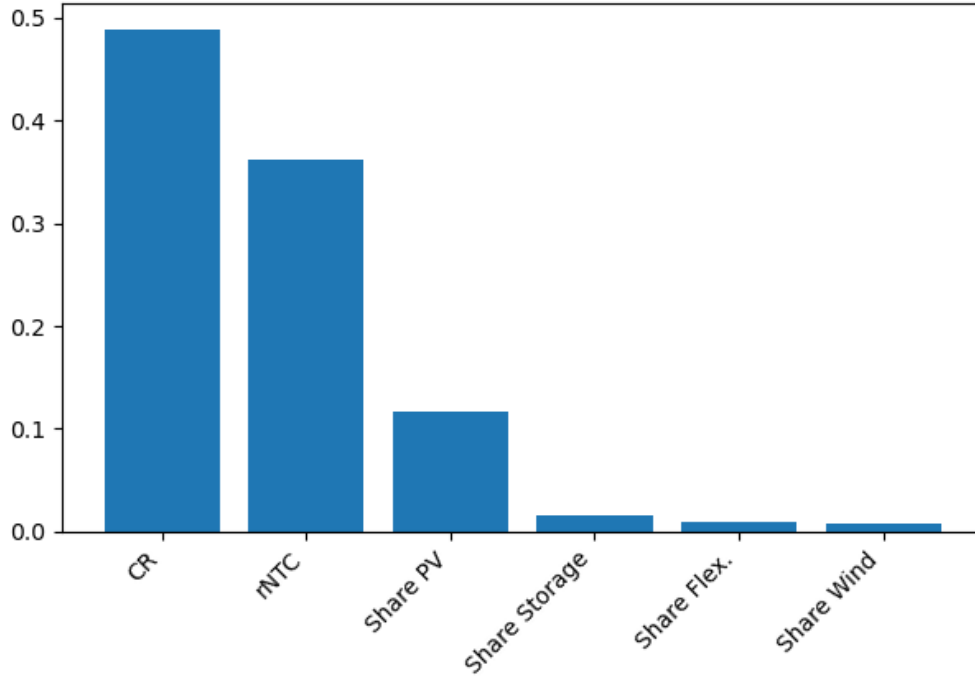


Figure 5.18: Features importances - RF Regressor Model - Load shedding target.

Concerning the capacity ratio and the ratio of NTC, nothing is surprising. The analysis about how they impact the load shedding are described in Section 5.2.2. However, the share of PV looks to be more implied than the share of wind which is surprising.

5.10 Load Shedding - MLP Regressor with over-sampling

The hyperparameters obtained are presented in Table 5.11.

Then, the model is validated via the unseen test fold, giving the results shown in Table 5.12.

The model with oversampling, compared to the one without, has a lower error on the same unseen test fold leading to the conclusion that it is interesting to deal with class-imbalance using over-sampled data.

	Values
hidden_layer_sizes	(200,100,100,50)
activation	relu
solver	adam
alpha	0.0001
learning_rate	constant
RMSE result	0.011

Table 5.11: Load shedding MLP (with over-sampling) Hyperparameters.

	MLP Regressor with Over-sampling	MLP Regressor
RMSE	0.014	0.016

Table 5.12: Performance metrics for the two MLP Regressor models - Test fold on the load shedding target.

The scatter plots depicted in Figure 5.19 shows that the oversampling slightly increases the score on the test fold and allows for a better prediction of the larger values.

5.10.1 Execution Times

In Table 5.13, prediction and training time for the load shedding target are displayed :

	MLP Regressor with over-sampling	MLP Regressor
Training Time [s]	14.4	6.27
Prediction Time [s]	0.022	0.063

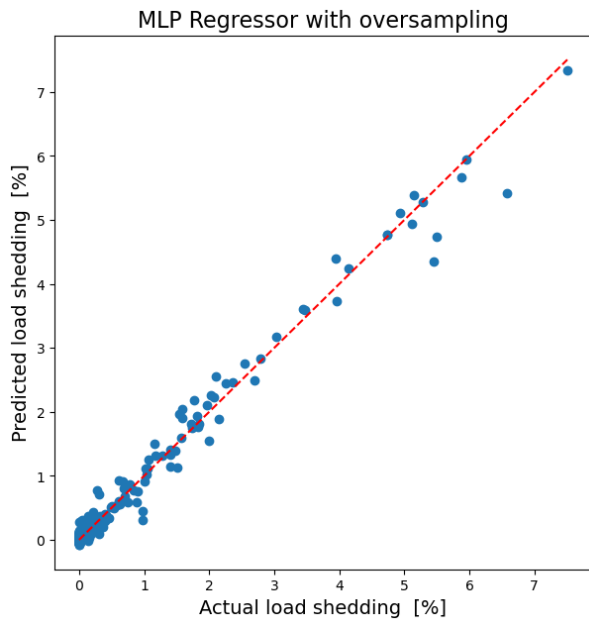
Table 5.13: Training and Prediction Time Comparison between MLP Regressor and MLP Regressor with over-sampling for load shedding target.

When a model is oversampled, training takes longer because the dataset is larger, requiring more computation during each iteration to adjust the model's weights and optimize learning, especially for complex algorithms.

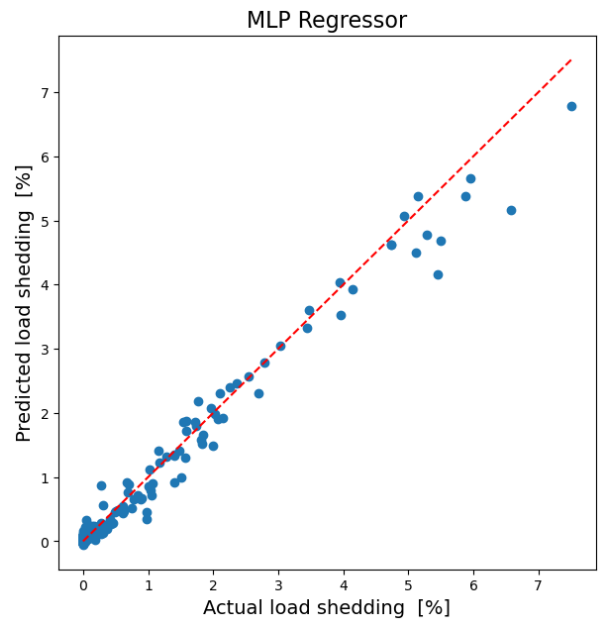
Considering the prediction time, it is so low that it can be considered that the prediction is similar. Trivially, because it used the same test fold.

5.10.2 Learning Curve

The learning curve is represented on Figure 5.20 :



(a) Over-sampled - MLP regressor.



(b) MLP regressor.

Figure 5.19: Scatter plots comparison for the load shedding target.

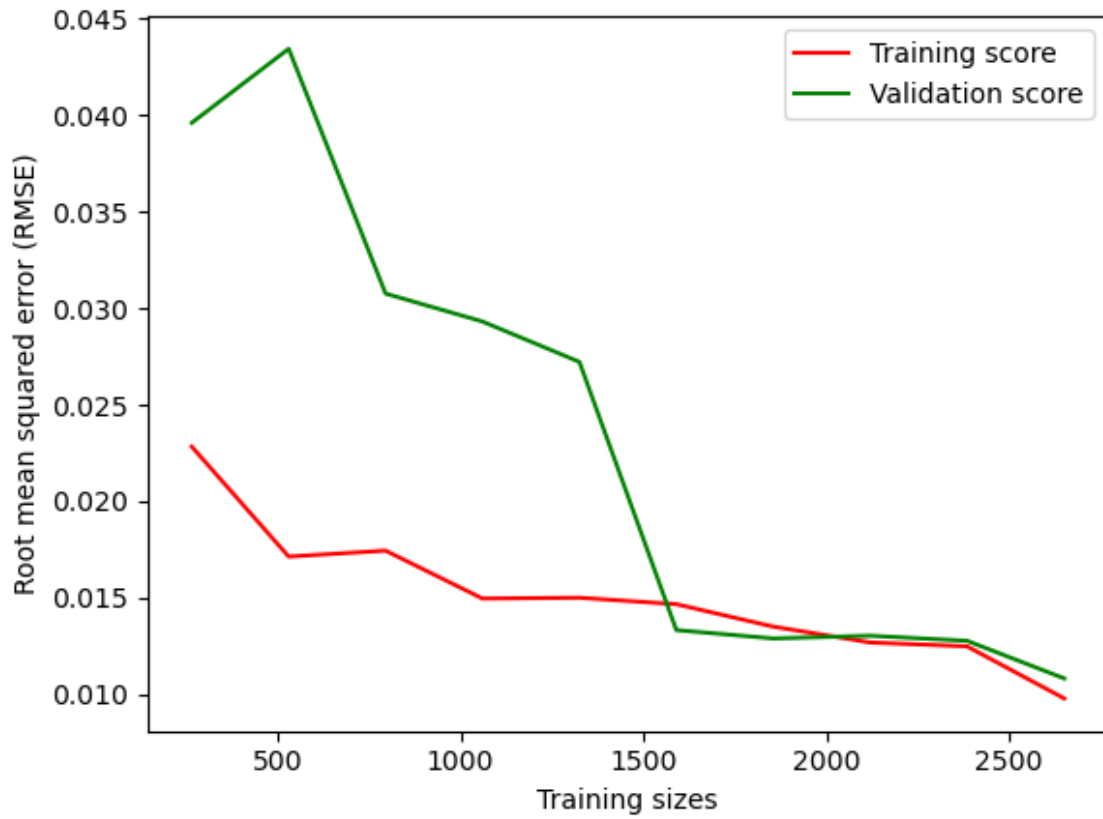


Figure 5.20: Learning curves on validation and training set - MLP regressor with oversampling - Load shedding target.

As more data is added the validation error decreases, indicating that the model is generalizing better. The training and validation errors are converging, which is a good sign that the model is neither over- nor under-fitting. The training curve does not increase upon adding training examples which is also good [15]. However, the curves seem to decrease still slightly using the maximum training size. Leading to the conclusion that perhaps more data (i.e. more simulations) could be interesting.

5.10.3 Loss curves

The loss curve for the load-shedding target's model is displayed in Figure 5.21 :

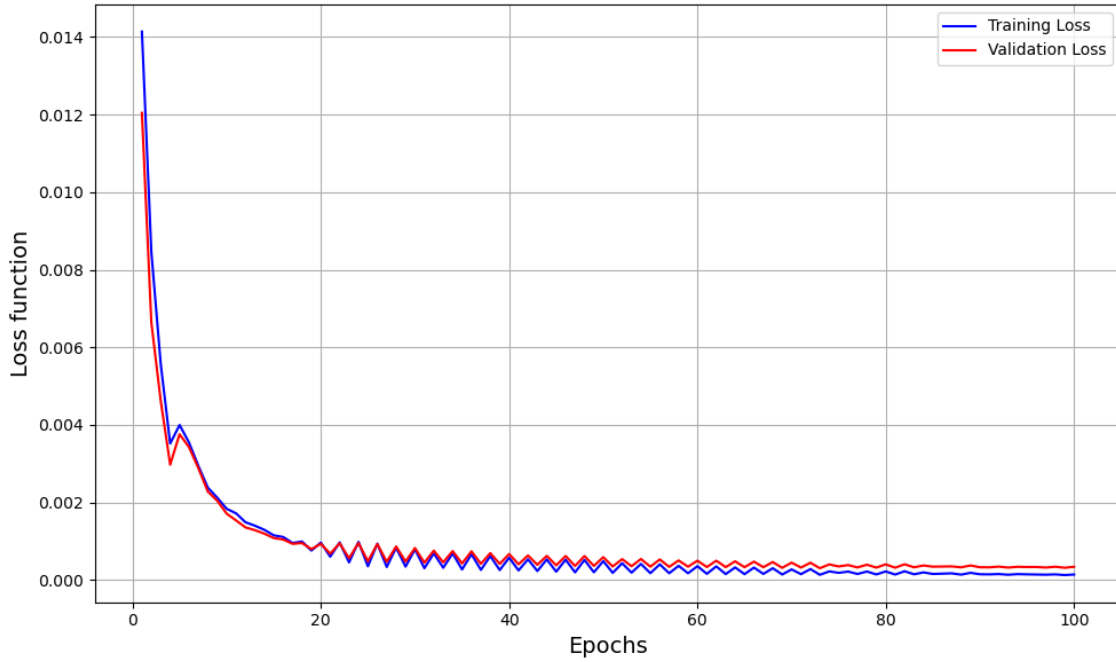
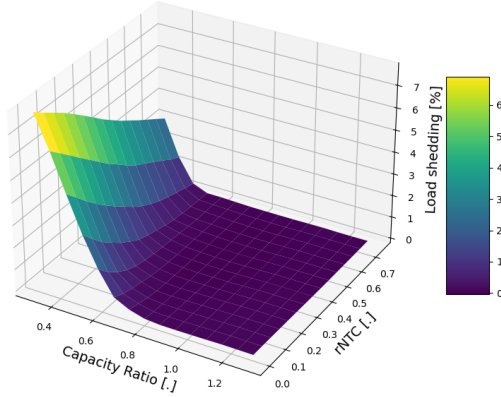


Figure 5.21: Loss curves on validation and training set - MLP Regressor model - Load shedding target.

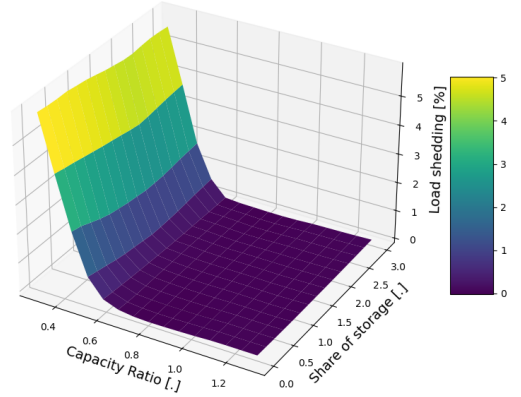
As can be observed, the curves look to converge correctly also, indicating that no overfitting occurs. Moreover, it shows that the maximum number of iterations can be limited to 100 by example.

5.10.4 Surface plots - Load shedding

Using the same methodology used for the curtailment target, below are the results of the surrogate model for the load shedding :

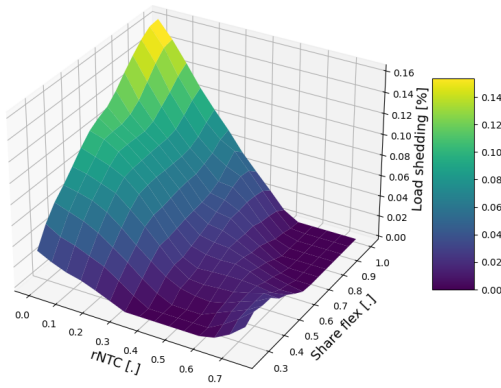


(a) Capacity ratio with rNTC.

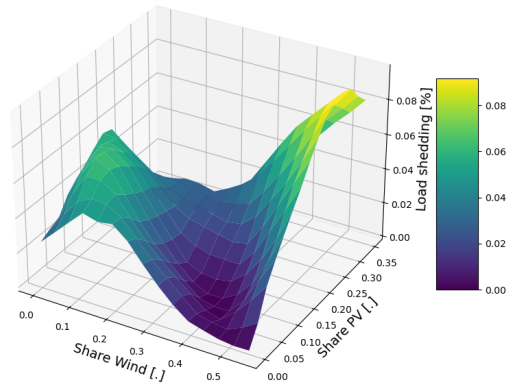


(b) Capacity ratio with share of storage.

Figure 5.22: Results of the surrogate model for the load shedding target - Surface plots with four constant and two varying features.



(a) rNTC with share of flexibility.



(b) Share of wind with share of PV.

Figure 5.23: Results of the surrogate model for the load shedding target - Surface plots with four constant and two varying features.

In Figure 5.22, it can be observed that the features exercising the most importance on the load shedding are the capacity ratio and the ratio of net transfer capacities. A too low capacity ratio combined to any transfer between zones leads to a demand which cannot be fulfilled until 7% of load shedding.

In Figure 5.23a, one can see that the share of flexible influences slightly the load shedding. One explanation could be that flexible units generally have lower capacity factors and may not always be reliable during peak demand or when renewable energy output is low.

In Figure 5.23b, the variation using less important features are not relevant. First, the load shedding vary between 0 and 0.08 % only. So, a hundred times less than by varying the capacity ratio. Secondly, non-linear models are designed to capture non-linear relationships between features

and target. The less important features may cause the model to overemphasize certain patterns that do not generalize well to new data. Less important features often add noise to the model rather than providing meaningful information. The model may try to learn patterns from this noise and that is why complex surface plots is observed. A path to try would be to train the model on predicting the load shedding without using some features of less importance as input parameters.

5.11 Further results beyond the regression range

Exploring beyond the training range of an MLP regression model is essential for assessing its generalization and robustness, as it reveals how well the model extrapolates to unseen data and behaves at the boundaries. This exploration helps identify limitations and ensures reliable predictions by highlighting potential biases or overfitting issues that may not be evident within the training range.

Table 5.14 show the extended bounds explored :

	Bounds	Extended Bounds
Capacity Ratio	[0.4 - 1.3]	[0.1 - 2]
Share Flexibility	[0.25 - 0.9]	[0.1 - 0.99]
Share Storage	[0.00 - 3.00]	[0.00 - 6.00]
Share Wind	[0.0 - 0.55]	[0.0 - 1.10]
Share PV	[0.0 - 0.35]	[0.0 - 0.7]
rNTC	[0.0 - 0.75]	[0.0 - 1.5]

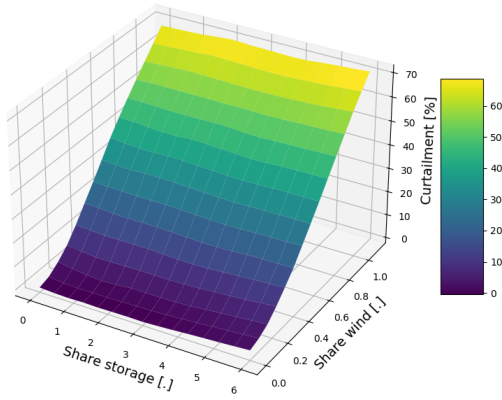
Table 5.14: Reference bounds and explored bounds of the different features of the two surrogate models

Figure 5.24 presents surfaces plots which explore this extended domain.

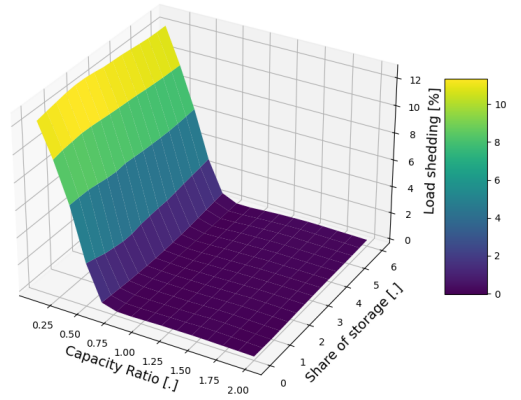
Half of the figures presented below are showing a good generalization of the extended range features. However, some observations can be made on the others.

The MLP regressor can capture complex non-linear interactions between variables, and in highly interconnected systems like electrical grids, indirect effects may emerge. For instance, in Figure 5.24c, high flexibility combined with a high rNTC might lead to unexpected behaviors, especially if the flexibility is concentrated in specific areas. However, due to the low importance of this feature in the prediction of the curtailment (0.5%), it is probably an artifact of the prediction.

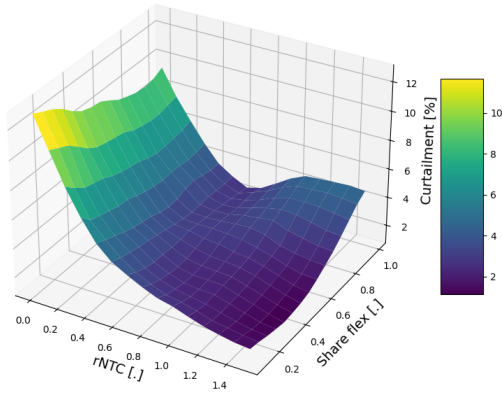
In Figure 5.24d and 5.24f, the apparent complexity in these relationships may be also influenced by artifacts emerging from the low importance of some features, like the share of wind, the share of flexibility or the share of PV. All of which contribute minimally to the analysis. These minor contributors can introduce noise in multi-dimensional plots, complicating the interpretation



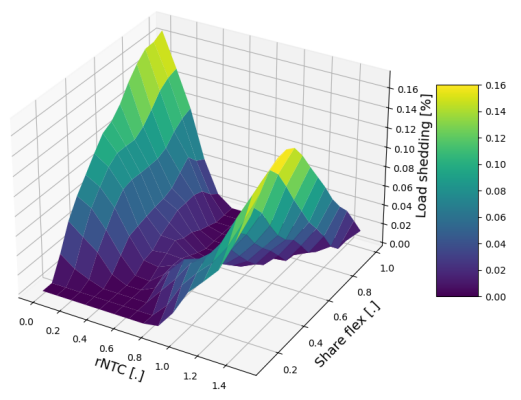
(a) Curtailment against share of storage and share of wind.



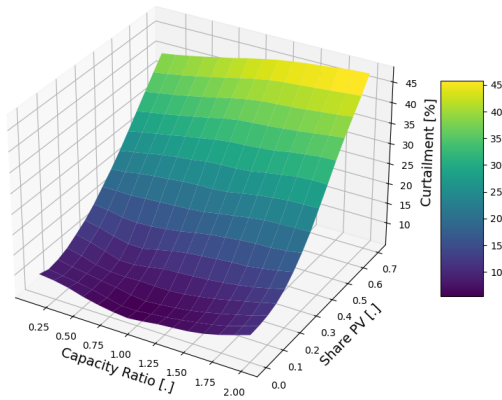
(b) Load shedding against capacity ratio and share of storage.



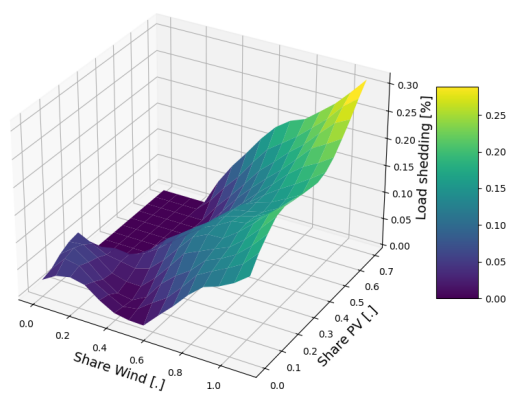
(c) Curtailment against rNTC and share of flexibility.



(d) Load shedding against rNTC and share of flexibility.



(e) Curtailment against capacity ratio and share of PV.



(f) Load shedding against share of wind and share of PV.

Figure 5.24: Results of the two surrogate models by exploring beyond the regression range.

of the results. The very low values in the load shedding, less than 1 %, observed on these two figures are also pointing in the same direction as the analysis above.

These results show how crucial it is to consider feature importance to avoid obscuring more significant trends.

6 Conclusion

In this work, a detailed sensitivity analysis of the DISPA-SET model was conducted by varying six key features within a six-dimensional feature space to understand how these factors influence dispatch operations. To facilitate this process, adjustments were made to allow the variation of parameters from a reference simulation, enabling the completion of all necessary DISPA-SET simulations and the creation of a comprehensive dataset.

Sensitivity analysis reveals several key insights:

1. Increasing Variable Renewable Energy Sources (VRES), like PV and wind, reduces carbon emissions but leads to higher curtailment during peak periods due to excess energy. This can be mitigated through storage technologies and inter-zonal energy transport to manage surplus and match supply with demand.
2. Predicting future changes in the capacity ratio, which balances slow and flexible units, is complex. Slow units, such as nuclear, provide stable energy but are inflexible, while flexible units, like gas plants, adapt to VRES fluctuations but produce more CO₂. Balancing these units is crucial for a reliable and green electricity network.
3. Improved net transfer capacity (rNTC) increases electricity exchanges, leading to increased production in high-capacity countries, lower import costs for others, and reduced gas use in gas-dependent countries.
4. The main purpose of stationary batteries is to store excess renewable energy and prevent curtailment, the analysis reveals that they also reduce dependence on cross-border imports for some countries by optimizing the use of locally generated energy.

Due to the distinctly different distributions of the target metrics—curtailment and load shedding—two separate surrogate models were developed to predict each outcome accurately. These models provide efficient tools for estimating these critical metrics, enhancing the analysis and optimization of power system operations.

The surrogate models developed for each of the targets - curtailment and load shedding has demonstrated a significant reduction in computational time, making it a powerful tool for simulating the European power system. By replacing traditional simulations, which can take up to 1.5 hours, the surrogate models complete the same tasks in a matter of seconds, for the desired output they were trained on and for the same values of the fixed parameters. This drastic reduction in time comes with a minor trade-off in precision. However, the slight loss in accuracy is generally acceptable given the substantial gains in speed for more extensive scenario analysis. Thus enabling better prediction and estimation of risks related to curtailment or load shedding. Overall, the surrogate model represents a valuable advancement in the optimization of energy system simulations.

Results indicate that curtailment is strongly correlated with the transfer capacities between zones, the installed capacity of wind turbines, and slightly with the installed capacity of photovoltaic. In contrast, load shedding is more closely associated with the capacity ratio (reflecting all production units other than PV and wind) and the transfer capacities between zones.

6.1 Future prospects for continuing this work

First, considering the impact of each feature of the surrogate model, adding/deleting one or another feature that can be more/less correlated with each of the targets. For example, power plant outages increase the risk of load shedding, and the amount of flexible units available in the affected areas is a key factor that can influence this need. High availability of flexible units helps reduce the need for load shedding by providing rapid and adjustable production capacity to compensate for the loss of generation due to outages. Also, for the curtailment, the location of the storage utilities. In a region with a large wind farm, the intermittent nature of wind power often results in times when wind generation exceeds the immediate demand for electricity, leading to potential curtailment (where excess wind energy is wasted). To address this, a network of storage units close to the wind turbines.

The modified work needs to be integrated into MEDEAS, a framework for Integrated Assessment Models (IAMs) that examines interactions between the economy, society, and environment. MEDEAS is used to explore sustainability and energy transitions by solving dynamic differential equations and assessing scenarios. Once integrated, the new work can be compared with prior results to evaluate its impact and accuracy.

Exploring solutions to deal with the use of the MILP formulation on the cluster which requires high computational costs but should obtain more accurate and constrained results.

Adding noise to demand parameters introduces variability that can simulate real-world uncertainties and unexpected fluctuations in electricity demand. This helps in creating more robust dispatch strategies that can handle unexpected changes. Or, noise in cost parameters (such as fuel prices or operational costs) reflects the real-world variability in generation costs. This can affect decisions on which generators to dispatch and when to use expensive or cheap resources.

Finally, it could be valuable to introduce noise or adjust the prices of curtailment and load shedding. Modifying the curtailment cost, especially by increasing it, could reveal the influence of stationary batteries more clearly, as the model would then be incentivized to use batteries to reduce curtailment. Additionally, lowering the cost of load shedding could produce a dataset with more positive, non-zero values, enhancing the ability of machine learning models to identify and predict these occurrences more accurately. This approach would lead to a more robust analysis by providing a broader range of data for training and improving model performance in identifying critical energy system behaviors.

References

- [1] François Straet. *Improving the simulation of variable renewable energy in the MEDEAS integrated assessment model*. 2023. URL: <http://hdl.handle.net/2268.2/18379>.
- [2] Ghent University. *Surrogate Modeling*. Accessed in August 2024. URL: <https://www.ugent.be/ea/idlab/en/research/machine-learning-and-data-mining/surrogate-modeling.htm#:~:text=Surrogate%20modeling%20is%20an%20interdisciplinary,complex%20time%2Dconsuming%20computer%20simulations..>
- [3] Stefan Pfenninger, Adam Hawkes, and James Keirstead. “Energy systems modeling for twenty-first century energy challenges”. In: *Renewable and Sustainable Energy Reviews* 33 (2014), pp. 74–86. ISSN: 1364-0321. DOI: <https://doi.org/10.1016/j.rser.2014.02.003>. URL: <https://www.sciencedirect.com/science/article/pii/S1364032114000872>.
- [4] Sylvain Quoilin, Ignacio Hidalgo Gonzalez, and Andreas Zucker. *Modelling Future EU Power Systems Under High Shares of Renewables: The Dispa-SET 2.1 open-source model*. English. Publications Office of the European Union, 2017. ISBN: 978-92-79-65265-3. DOI: [10.2760/25400](https://doi.org/10.2760/25400). URL: <https://ec.europa.eu/jrc/en/publication/eur-scientific-and-technical-research-reports/modelling-future-eu-power-systems-under-high-shares-renewables-dispa-set-21-open-source>.
- [5] GAMS Development Corporation. *General Algebraic Modeling System (GAMS) Release 24.5.6*. 2015. URL: <https://www.gams.com/>.
- [6] Selen Cremaschi Bianca Williams. *Surrogate Model*. Accessed in August 2024. URL: <https://www.sciencedirect.com/topics/engineering/surrogate-model>.
- [7] Amelia R. Shaw Heather Sawyer. *Hydropower Optimization Using Artificial Neural Network Surrogate Models of a High-Fidelity Hydrodynamics and Water Quality Model*. Accessed in August 2024. URL: https://www.researchgate.net/publication/320603904_Hydropower_Optimization_Using_Artificial_Neural_Network_Surrogate_Models_of_a_High-Fidelity_Hydrodynamics_and_Water_Quality_Model.
- [8] Michael Kummert Florent Herbinger Colin Vandenhof. *Building energy model calibration using a surrogate neural network*. Accessed in August 2024. 2023. URL: <https://www.sciencedirect.com/science/article/pii/S0378778823002876?>
- [9] *EU Reference Scenario 2020 - Energy, transport and GHG emissions - Trends to 2050*. Brussels: European Commission. URL: https://energy.ec.europa.eu/data-and-analysis/energy-modelling/eu-reference-scenario-2020_en?prefLang=sv.
- [10] *Power System Flexibility for the Energy Transition, Part 1: Overview for policy makers*. Abu Dhabi: International Renewable Energy Agency, 2018. URL: <https://www.irena.org/publications/2018/Nov/Power-system-flexibility-for-the-energy-transition>.
- [11] Nucleair Forum. *Small Modular Reactor*. Accessed in May 2024. URL: <https://www.forumnucleaire.be/topics/smr-small-modular-reactors>.

- [12] Jonas Hoersch et al. “PyPSA-Eur: An open optimisation model of the European transmission system”. In: *Energy Strategy Reviews* 22 (2018), pp. 207–215. DOI: [10.1016/j.esr.2018.08.012](https://doi.org/10.1016/j.esr.2018.08.012). eprint: [1806.01613](https://arxiv.org/abs/1806.01613).
- [13] Thierry Labro. *Batteries stationnaires: une révolution attendue*. Accessed in May 2024. URL: <https://paperjam.lu/article/batteries-stationnaires-revolu>.
- [14] *scikit-learn Machine Learning in Python*. Accessed in June 2024. URL: <https://scikit-learn.org/stable/>.
- [15] KSV Muralidhar. *Learning Curve to identify Overfitting and Underfitting in Machine Learning*. Accessed in August 2024. 2021. URL: <https://towardsdatascience.com/learning-curve-to-identify-overfitting-underfitting-problems-133177f38df5>.
- [16] PENALBA ARAGUES M. *Data science applied to electrical energy systems - S6 : Classification*. Diapositives, Universitat Politècnica de Catalunya. Accessed in August 2024. 2023.
- [17] Carla Vidal Montesinos. *Integrating short-term dispatch constraints in a system dynamics energy planning model*. 2022. URL: <http://hdl.handle.net/2268.2/16566>.
- [18] IBM. *What is a neural network?* Accessed in June 2024. URL: <https://www.ibm.com/topics/neural-networks>.
- [19] Gourav Singh. *Introduction to Artificial Neural Networks*. Accessed in June 2024. URL: [https://www.analyticsvidhya.com/blog/2021/09/introduction-to-artificial-neural-networks/#:~:text=Artificial%20Neural%20Networks%20\(ANN\)%20are,human%20brain%20Biological%20Neural%20Networks..](https://www.analyticsvidhya.com/blog/2021/09/introduction-to-artificial-neural-networks/#:~:text=Artificial%20Neural%20Networks%20(ANN)%20are,human%20brain%20Biological%20Neural%20Networks..)
- [20] Kiprono Elijah Koech. *The Basics of Neural Networks (Neural Network Series) — Part 1*. Accessed in June 2024. URL: <https://towardsdatascience.com/the-basics-of-neural-networks-neural-network-series-part-1-4419e343b2b>.
- [21] Nima Beheshti. *Random Forest Regression*. Accessed in June 2024. URL: <https://towardsdatascience.com/random-forest-regression-5f605132d19d>.
- [22] Jason Brownlee. *A Gentle Introduction to k-fold Cross-Validation*. Accessed in June 2024. URL: <https://machinelearningmastery.com/k-fold-cross-validation/>.
- [23] IAMC. *Integrated assessment modeling consortium*. Accessed in August 2024. URL: <https://www.iamconsortium.org/what-are-iams/>.
- [24] Iñigo Capellán-Pérez et al. “MEDEAS: a new modeling framework integrating global biophysical and socioeconomic constraints”. In: *Energy Environ. Sci.* 13 (3 2020), pp. 986–1017. DOI: [10.1039/C9EE02627D](https://doi.org/10.1039/C9EE02627D). URL: <http://dx.doi.org/10.1039/C9EE02627D>.
- [25] Jason Brownlee. *How to use Learning Curves to Diagnose Machine Learning Model Performance*. Accessed in August 2024. 2019. URL: <https://machinelearningmastery.com/learning-curves-for-diagnosing-machine-learning-model-performance/>.
- [26] Arnold Johan Rix Kelly Kemper Ulrich Minnaar. *Analysis of utility scale wind and solar plant performance in South Africa relative to daily electricity demand*. Accessed in August 2024.

2017. URL: https://www.researchgate.net/figure/High-season-solar-PV-hourly-average-capacity-factors-and-the-high-season-national-demand_fig2_321192910.