Exploring Soft-linking of Market Modules and an Optimization Model

A case study on the coupling of the energy models
EM-Lab and COMPETES

Exploring Soft-linking of Market Modules and an Optimization Model

A case study on the coupling of the energy models EM-Lab and COMPETES

Dissertation

for the purpose of obtaining the degree of Master of Science
Electrical Power Engineering
at Delft University of Technology,
to be defended publicly Monday August 30th, 2021 at 10:00

by

Jim Hommes

Supervised by:

Dr. Milos Cvetkovic	Technische Universiteit Delft
M.Sc. Ingrid Sanchez Jimenez	Technische Universiteit Delft

Composition thesis commitee:

Prof. dr. P. Palensky	chair
Dr. M. Cvetkovic	Technische Universiteit Delft
Dr. ir. L.J. de Vries	Technische Universiteit Delft

Keywords: Soft-linking, Model coupling, Agent-based modeling, Optimization modeling, CO2 Market, Capacity Mechanism

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Contents

Summary

The European Green Deal states that the European Union will have to cut 55% of carbon-dioxide (CO2) levels and implement a share of at least 40% of renewable energy sources in the electricity sector by 2030, compared to levels of 1990. While renewable energy technologies are effective in cutting CO2 levels, they also bring challenges like variability and unpredictability which affect power system stability. The TradeRES project was founded in order to create and test new innovative electricity market designs that incorporate near 100% renewable energy generation. Multiple research groups across the EU participate in TradeRES, including TU Delft and TNO. In order to be able to research future system stability, this project seeks to improve investment decision-making in the energy models EM-Lab and COMPETES, provided by TU Delft and TNO respectively. To fully utilize the sophistication of both models and to prevent development of a new model, this thesis focuses on the coupling of the models through soft-linking and its validation in a Dutch energy transition context.

This thesis presents a conceptual approach and implementation towards the aforementioned soft-linking. First, the definition and requirements of soft-linking are defined to mitigate ambiguity created by previous work. These requirements are then implemented in a conceptual model describing data exchange, data mapping and timing of the models in the context of soft-linking. The conceptual model describes how the strengths of both models are utilized in the coupling. EM-Lab is an agent-based model used for studying investment behavior and has strong policy and market modules. COMPETES is an optimization model that has a very sophisticated and detailed dispatch module. The nature of these models provide coupling potential and constraints. A series of design choices describe how these models interact in the soft-linking, and how the decision was made to couple specific modules from EM-Lab with COMPETES to improve overall investment decision-making.

The implementation entails the recreation of EM-Lab market modules with Python and the soft-linking of these modules with COMPETES using the software kit SpineToolbox. From EM-Lab, the CO2 market and the capacity market were recreated in Python. These modules could be used to complement COMPETES and improve overall investment decision-making. Pseudo-code of the implemented algorithms is provided and a thorough description of the programmatic interactions of the models is given. Also, the SpineToolbox implementation is shown and elaborated on.

To ensure correctness, the soft-linking is validated and verified. Standard verification methods like face validity and static functional testing are used. For validation, the soft-linking is used to generate results in case studies which have been used to validate COMPETES. These case studies resemble a Dutch energy transition scenario. The results from the soft-linking are then compared to the validated COMPETES results. As the differences in results could all be explained, there is no reason to assume the soft-linking is invalid.

Results of the soft-linking are provided and contain graphs regarding the CO2 market, capacity market, dispatch, and investment and decommissioning decisions. The CO2 market presents volatile and extreme results, begging the question whether the current implementation is sufficient for future power system analysis. However, the results do suggest that the soft-linking has effect on investment and decommissioning decision-making. In conclusion, additional work, like representation of mechanisms that help stabilize the CO2 prices, is required in order to use this soft-linking for future power system analysis. The potentials of soft-linking are explored and suggestions are made for future work.

Samenvatting

De Europese Green Deal geeft aan dat de Europese Unie de niveaus koolstofdioxide (CO2) met 55% zal moeten verminderen en dat minstens 40% van de energiebronnen duurzaam zullen moeten zijn in de energiesector voor 2030, vergeleken met de niveaus van 1990. Duurzame energiebronnen zijn effectief in het drukken van de uitstoot van CO2, maar ze brengen ook uitdagingen met zich mee zoals variabiliteit en onvoorspelbaarheid die effect hebben op de stabiliteit van het elektriciteitsnetwerk. Het TradeRES project is opgericht om een nieuw en innovatief eletriciteitsmarkt ontwerp met een nabije 100% duurzame energievoorziening te creëren en te testen. Meerdere onderzoeksgroepen in de EU nemen deel aan het TradeRES project, waaronder de TU Delft en TNO. Om de stabiliteit van het toekomstige elektriciteitsnetwerk te kunnen onderzoeken is het doel van dit project het verbeteren van de besluitvorming van investeringen in de energiemodellen EM-Lab en COMPETES, respectievelijk de modellen van de TU Delft en TNO. Om volledig van de complexiteit van beide modellen te kunnen profiteren en om te voorkomen dat een volledig nieuw model ontwikkeld moet worden focust deze thesis op het koppelen van de modellen door middel van soft-linking en de validatie daarvan in context van de Nederlandse energietransitie.

Deze thesis presenteert een conceptuele benadering en implementatie van de genoemde soft-linking. Eerst zijn de definitie en de eisen voor soft-linking gedefinieerd om de ambiguïteit die ontstaan is in de literatuur te verhelpen. De eisen zijn dan geïmplementeerd in een conceptueel model dat de data uitwisseling, data transformatie en het tijdschema van de modellen beschrijft. Het conceptuele model beschrijft hoe de krachten van beide modellen zijn gebruikt in het koppelen. EM-Lab is een agent-based model dat gebruikt wordt voor het bestuderen van het gedrag van investeerders en heeft sterke beleids- en marktmodules. COMPETES is een optimalisatie model dat een erg complexe en gedetailleerde generatie toezeggingsalgoritme heeft. De typen van beide modellen zorgen voor potentie en uitdagingen in het koppelen. Een serie aan ontwerpkeuzes beschrijven hoe de modellen elkaar gebruiken in de soft-linking, en hoe de keuze is gemaakt om specifieke modules van EM-Lab te koppelen met COMPETES om de besluitvorming met betrekking tot investeringen te verbeteren.

De implementatie omvat het recreëren van EM-Lab marktmodules in Python en de soft-linking van de modules met COMPETES door middel van het softwarepakket SpineToolbox. Vanuit EM-Lab zijn de CO2- en de capaciteitsmodules gerecreëerd in Python. De modules kunnen worden gebruikt om COMPETES te complementeren en investering besluitvorming te verbeteren. Pseudo-code is gegeven van de geïmplementeerde algoritmen en een gedetailleerde beschrijving is gegeven van de

programmatische interacties van de modellen. Tot slot is de SpineToolbox implementatie gepresenteerd en uitgelegd.

Om de correctheid van de soft-linking te kunnen garanderen is de soft-linking gevalideerd en geverifieerd. Standaard verificatiemethoden, zoals het gebruik van de intuïtie van experts en statisch functioneel testen, zijn gebruikt. Voor de validatie, de soft-linking is gebruikt om resultaten te genereren in een casus die gebruikt is om COMPETES mee te valideren. Deze casus lijkt op een Nederlands energietransitie scenario. De resultaten van de soft-linking zijn vervolgens vergeleken met de gevalideerde resultaten van COMPETES. Omdat de verschillen in de resultaten verklaard konden worden is er geen reden om te denken dat de soft-linking niet valide is.

De resultaten van de soft-linking zijn gegeven waaronder grafieken van de CO2 markt, capaciteitsmarkt, generator toezegging en investeringen en buitengebruikstellingen. De CO2 markt toont gevoelige en extreme resultaten, waardoor afgevraagd kan worden of de huidige methode voldoende is voor het analyseren van het toekomstige energiesysteem. Echter, de resultaten geven aan dat de soft-linking een effect heeft op de besluitvorming met betrekking tot investeringen en buitenwerkingstellingen. Concluderend, extra werk, waaronder de representatie van mechanismen die de CO2 markt stabiliseren, is vereist om deze soft-linking te kunnen gebruiken voor de analyse van de stabiliteit van het toekomstige energienetwerk. De potenties van de soft-linking zijn verkend en er zijn suggesties gedaan voor toekomstig werk.

Preface

My academic career started in Computer Science. While I am passionate about this subject, I felt my ambitions lay in applying Computer Science in a different field rather than following a career in Computer Science. This, and my fascination of renewable energy technologies, led to me making the switch to Electrical Power Engineering. It is this mix of interests and abilities that have led me to Dr. Milos Cvetkovic and this project.

The EU has set very high and noble ambitions in regard to the decarbonization of its energy sources. These ambitions will partly be realized by the implementation of a high share of renewable energy. This brings challenges which not all current market designs and energy models are accurately able to handle. For this reason, the TradeRES project is founded in order to create and design market designs and energy models. Under this project the TU Delft and TNO are coupling their energy models, EM-Lab and COMPETES, through means of soft-linking. This project aims to implement and validate this soft-linking, for which the mix of programming and knowledge of the energy context is required.

This paper provides the implementation, validation and results of the soft-linking. The soft-linking is validated by analyzing the effects of the coupling on the results compared to the validated model without soft-linking. This validation is done in a scenario resembling the Dutch energy transition. The results show that the soft-linking was effective, but begs the question whether the current implementation is the right way forward as the results show volatile prices among other issues.

The project is developed from perspectives of multiple fields. Expert in the field of energy markets and often available for counsel regarding the soft-linking is Dr. ir. Laurens de Vries, who will be participating in the committee evaluating this thesis. Chair will be Prof. dr. Peter Palensky, who is active in intelligent electrical power grids and who has a very broad history of research in future and complex power systems. Lastly, my personal supervisor for this thesis, Dr. Milos Cvetkovic, will be taking part in the thesis committee. Milos provided guidance during the entire thesis and persevered in advising during challenges this project, for which I give special thanks.

This report is the conclusion of more than 9 months of work. With that, it indicates the end of an 8-year-long chapter in my life. I have enjoyed every moment of work on this thesis, so please enjoy reading this report!

Jim Hommes

Delft, August 2021

Acknowledgements

First and foremost, I would like to thank my supervisors for their unrelenting support and wisdom. The weekly meetings with Dr. Milos Cvetkovic and M.Sc. Ingrid Sanchez Jimenez quickly became routine and was looked forward to every week. Milos provided good support and often came with quick and creative solutions to the issues encountered in this thesis. Special thanks go to Ingrid with whom I have worked closely together the last months. Ingrid could provide knowledge and her abilities in research helped me further with my work.

Multiple times we met with Dr. ir. Laurens de Vries from the TU Delft TBM faculty. Laurens provided a unique and useful insight into the coupling, and I was able to learn from his expertise in energy market design.

This project would not have succeeded without the help of TNO's modelers German Morales España and Ricardo Hernandez Serna. German's expertise was essential to the development of the conceptual model and provided great insight in the workings of TNO's energy model COMPETES. Special and great thanks go to Ricardo, who persisted and worked through his summer holiday to assist my work. Ricardo understood the unfortunate timing of finalizing a thesis during the summer and came through, proving how kind and thoughtful he is.

Thanks go to VTT's specialists Manuel, Pekka, Antti, Juha and more for their willingness to cooperate and provide technical support throughout the thesis. Spine-Toolbox was under heavy development, but if problems occurred, they were always able to prioritize my operational status so that work could continue as fast as possible.

Without the unconditional and emotional support of my girlfriend and family this thesis would not have been possible. They have always provided me with comfort during stressful times and were always available for a quick dinner or simply for some company. These are small acts but help mentally and provide new energy to keep working.

List of Abbreviations

ACER	Agency for the Cooperation of Energy Regulators
BASE	COMPETES only
CM	Capacity Market
CO2	Carbon-dioxide
EMCAS	Electricity Market Complex Adaptive Systems
EM-Lab	Energy Modeling Laboratory
ETS	Emissions Trading System
EU	European Union
H2	Hydrogen
ICAP	Installed Capacity Market
IRM	Installed Reserve Margin
MSR	Market Stability Reserve
MVS	Multi Vector System
NOM	Nominal or non-scarcity
NYISO	New York Independent System Operator
O&M	Operations and Maintenance
PV	Photovoltaics
RES	Renewable Energy Sources
SCAR	Scarcity
SDC	Sloping Demand Curve
SEPIA	Simulator for Electric Power Industry Agents
SOFT	Full Soft-Linking
STEMS-RT	Short-Term Electricity Market Simulator-Real Time
TNO	The Netherlands Organization for Applied Scientific Research
VOLL	Value of Lost Load
VTT	Technical Research Centre of Finland
VRE	Variable Renewable Energy
WTP	Willingness To Pay

1
Introduction

The European Union (EU) is moving towards 100% sustainable energy generation. The integration of renewable energy brings new challenges, like variability and unpredictability, which call for a change in the existing market designs and energy models. For this reason, the TradeRES project was created with the purpose of developing and testing an innovative electricity market design that allows a near 100% sustainable energy generation. This exploratory research aims to integrate the potential flexibility of short-term markets with the imperfect investment behavior of energy companies through means of soft-linking.

1.1. Background

The EU is making changes to aid in the decarbonization of its energy sources [1][2]. This includes the integration of renewable energy sources (RES) like photovoltaics (PV) and wind [3]. In December 2019, the European Green Deal was proposed setting ambitious goals for the future of the EU, like cutting 55% of carbon dioxide (CO2) levels of the energy sector by 2030 compared to the levels in 1990 [4]. The deal also entails that by 2030 the EU's electricity generation must have a share of 40% renewable energy sources and a 36-39% efficiency improvement compared to 1990.

Sustainable sources are an effective tool towards the energy transition, however they are variable and hard to accurately predict as opposed to the currently implemented energy sources [5][6]. These challenges have a direct result on energy system stability and availability, but also on the current energy market designs [7]. This calls for a new and critical look towards the current models and market designs.

In the EU the TradeRES project was created with multiple research groups across the continent, including Delft University of Technology (TU Delft) and the Netherlands Organization for Applied Scientific Research (TNO), with the intention of creating and testing new market designs and energy models that can meet society's needs with a near 100% sustainable energy generation [8]. The goal is to create an economically efficient long-term sustainable market design which incorporates investment opportunities, a high participation of renewable energy sources, flexible electricity demand from households (e.g., prosumers), and more features that are critical towards the integration of sustainable energy. In addition, their goal is to improve the current energy models available using these developments.

Crucial to researching future system stability is to be able to predict the future energy mix. This prediction is made by determining the investment and decommissioning decisions up to that point in time. This decision-making is subject for research as, for example, policy makers and other governing entities need to understand the effects of their work [9][10]. Hence, modelers seek to improve this decision-making in their energy models.

Dispatch is an important part of an energy model. Through dispatch profitability of generators is determined, affecting the investment decision-making. Access is given to TNO's energy model COMPETES [11]. COMPETES is an optimization model incorporating an investment model and a detailed dispatch model. The model's main purpose is economic dispatch and it's capable of least-cost unit commitment or least-cost target determination for capacity and operation expansion. Its strengths lie in its detail on flexible generation technologies and interconnection capacities.

TU Delft has given access to their EM-Lab energy model [12]. This is also an investment and dispatch model. The main difference is that it is an agent-based simulation model where the roles of investors and other actors are modeled as individual agents. This leads to the strength of EM-Lab: the agent has a limited understanding of the environment it operates in, creating a more realistic representation of a real-life scenario. Investors strive to create the highest net present value but base their decisions on their own forecast of an amount of years ahead for their investments. This will from hereon be referred to as imperfect investment or imperfect simulation. The agent-based structure makes EM-Lab attractive to study energy policies and market mechanisms as agent decision-making under these policies and mechanisms can be effectively studied.

EM-Lab and COMPETES are both sophisticated models and were developed over multiple years. Both models have been verified using multiple case studies. This begs the question of how to combine the models with the purpose of creating more flexibility regarding renewable energies. Creating a new model is costly and inefficient. Comparing results from two models not developed with the same methodology and with a different granularity level on the temporal, spatial and sectoral representation will most likely lead to inconclusive results, if comparable at all. An approach is needed where both models communicate and iteratively alter each other's results, known as co-simulation [13].

In the co-simulation field soft-linking is referred to as the coupling of separate models while allowing the exchange of information surrounding the model runs. A more precise definition considering previous work is given in Section 2.4.1. It has been shown that soft-linking is an effective method for creating stronger models [14][15]. The purpose of this project is to explore the potential of soft-linking EM-Lab and COMPETES. The soft-linking of these two models could result in a model where COMPETES's detailed dispatch is complemented by external modules from EM-Lab, creating imperfect simulation. In the scope of this project the modules in this soft-linking are the capacity market and CO2 market implementations in EM-Lab.

1.2. Motivation and Objective

It is a well-established fact that the world will have to move away from fossil fuels to renewable energy. The EU is making progress and will have even more ambitious goals for 2030 with the European Green Deal. This stresses the need for more complex and flexible market designs and energy models. The objective of this research is to investigate the modeling of imperfect myopic investment from EM-Lab, an agent-based model, through a model that allows detailed dispatch flexibility from COMPETES, an optimization model. This project, while a small step, will help transition the EU to a situation with a high share of renewable energy generation.

Applications for the model this project will produce are broad, but most importantly it will be a steppingstone for combining the sophistication from models in different fields. In terms of the European Green Deal, the model will allow the simulation of future scenarios with a high share of renewable energy and a high degree of flexibility. These simulations will be used to research under which conditions security of supply can be guaranteed.

Furthermore, soft-linking is a relatively new subject. The coupling of specific modules of an agent-based model with an optimization model is novel and this exploratory research will improve understanding within the scientific community.

1.3. Research Question

The entire project can be characterized by answering the following research questions:

1. **How can investment decision-making be improved by the soft-linking of an agent-based model and an optimization model?**
2. **How can the result of soft-linking be validated in a near-term energy transition scenario that resembles the Netherlands?**

The first question is broad and has been split up into the following sub-questions:

1

a) **What are the requirements for the soft-linking of an agent-based model and an optimization model?**

b) **Which data has to be exchanged and what is the data mapping in this soft-linking?**

c) **What time schedule can be used in order to soft-link models with different granularity level regarding temporal representation?**

1.4. Soft-Linking Validation

In order to gain confidence in the correctness of the model, the model will have to be validated. The soft-linking in this project will be validated in a scenario resembling the Netherlands, e.g., modeled after existing Dutch power plants and using Dutch renewable energy targets. This data set has also been used to validate the COMPETES optimization model. The demand in this scenario is altered to create two variations: one with, and one without scarcity. To validate the model, the results of the soft-linking are compared to the validated results of COMPETES. Chapter 5 goes in-depth regarding verification and validation of the model and the used data.

1.5. Thesis Outline

Chapter 2 will provide an overview of current literature on the topics of the energy modeling, model coupling, the CO2 and capacity market. It also does so by going in-depth in the models EM-Lab and COMPETES and the used software kit SpineToolbox. Chapter 3 identifies the coupling points and describes the path taken from the conceptual model to the soft-linking implementation. Chapter 4 goes in-depth on the modeling strategies and structures and discusses the coding implementation. Chapter 5 describes how the resulting model of the soft-linking was verified and validated. Chapter 6 presents the results of the non-scarcity and scarcity runs described in the previous chapters. Lastly Chapter 7 discusses conclusions drawn and future work.

2

Literature and Context

To be able to understand the scientific significance of the model coupling produced by this project it is important to understand the state-of-the-art and previous work. This chapter goes in-depth on the topics of energy system modeling, model coupling, and the CO2 and capacity market. In addition, an overview of literature is provided. Lastly, the capabilities, strengths and weaknesses of the used models and software are presented.

2.1. Capacity Market

The practical short-comings of the energy-only market designs gave rise to the implementation of different capacity mechanisms across the EU [16]. The increasing share of RES creates more scarcity situations driving the need for a capacity mechanism to uphold system adequacy [17].

The main purpose of a capacity mechanism is to guarantee a certain amount of capacity, resulting in grid stability and security of supply. It does so by creating an incentive to invest by a remuneration system: plants that would not run but are necessary for security of supply will be remunerated for their available capacity. This reasoning also works the other way around: plants are incentivized to not immediately disband when they are no longer profitable in the energy-only market.

In the EU multiple types of capacity mechanisms have been implemented. Table 2.1 shows which European countries have implemented which kind of capacity mechanism according to the European Union Agency for the Cooperation of Energy Regulators (ACER) [18].

There is still heavy discussion on the effects of the different capacity mechanism types as opponents see the subsidy as unnecessary or believe it to be suppressing other market prices and effectively increasing the price for the con-

Table 2.1: The currently implemented capacity mechanisms in European countries according to ACER. [18]

Capacity Mechanism	Countries
Capacity Market	France, Great Britain, Ireland, Poland, Portugal
Capacity Payment	Spain, Italy, Croatia
Strategic Reserve	Belgium, Finland, Germany, Poland, Sweden

sumer [19][20][21]. Others claim capacity mechanisms to be effective remedies for scarcity problems in the energy-only market and investment challenges [22][23][24].

A capacity market is a type of capacity mechanism in which generators and other load serving entities participate by bidding. Through this bidding process the market clearing price and which units receive revenue are decided. [16] Markets can differ in their implementation (e.g., timing and years ahead) or in their definition (e.g., what prices and what type of capacity is bid).

Multiple capacity markets have been proposed for multiple regions, suggesting the creation of investment incentives. In 2005, a capacity market was proposed that claimed to be learning from earlier capacity mechanism failures [25]. It is argued that capacity markets are required because of the lack of demand response in current energy-only markets. In this capacity market the target capacity is met using a short-run profit function. In this function the chance of serious error is minimized by use of a Locational Installed Capacity Market (ICAP) demand curve. This curve is found in Figure 2.1.

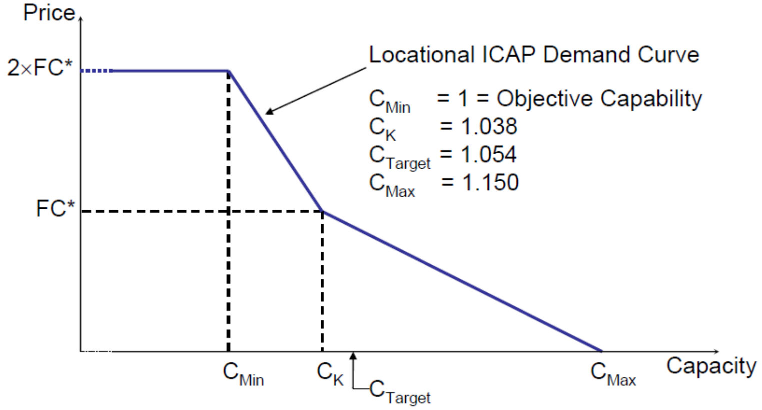


Figure 2.1: The locational ICAP demand curve [25]. The C parameters are user defined capacity amounts. FC is the expected carrying cost or the average profit required to make the investment into capacity profitable.

Bids are sorted in ascending order and cleared at the intersection. This way the supply bids determine who will supply but not the clearing price.

In 2017 a new capacity market was proposed for Poland and compared to the British model [26]. At the time of the proposal three capacity mechanisms were operational in Poland:

1. **Strategic Reserve**
   In a strategic reserve older or non-profitable plants that are scheduled to be decommissioned are subsidized to stay operational and used whenever there is a scarcity.
2. **Operating Reserve**
   In an operating reserve eligible generators receive compensation for unused capacity.
3. **Demand Side Response**
   In this scheme medium to high-end consumers are compensated for lowering their demand during peak hours to prevent scarcity.

The Polish ministry noted that these measures are short-term and are a direct response to the current challenges of the energy transition. Hence there is a call for a long-term solution. The proposed capacity market is a long-term solution for the mentioned market failures.

In this design existing, new, refurbished and certain aggregated generators are allowed to participate. To incentivize investment, the lengths of capacity agreements differ. For existing generators, the agreements yield 1 year. For refurbished generators this is 5 years and for new generators this is up to 15 years. The full capacity is bid for the price needed for the generator to break even that year.

2.2. CO2 Market

In 2005, the EU Emissions Trading System (ETS) was implemented in an effort to decarbonize the EU [27]. The EU ETS is a ‘cap and trade’ system, meaning that the amount of CO2 emission allowed is capped and producers must buy credits for their emissions or face penalties. Critical aspects of the ETS are the allocation program and the pricing mechanism. The EU ETS is a well-researched system and has been proven to be an effective tool [28].

Multiple challenges have arisen from the implementation of the ETS. In 2012, at the end of Phase II of the EU ETS a gap of 2 billion unused CO2 credits causing low CO2 prices sparked criticism of the EU ETS [29]. This initiated the debate on “back-loading”, a system where credits would be auctioned later than planned, and the Market Stability Reserve, a mechanism that would store credits in case of a surplus and release credits in case of a shortage.

Regarding the pricing mechanism, at the end of 2007 the CO2 price dipped and was set to go to 0. Research was done into this price volatility and found a discrepancy between their expected values (0.6 - 0.9 €/ton) and the real values (20 - 25 €/ton) [30]. It was found that the unlucky coincidence of high natural gas prices was of influence and that there was scarcity in hydro and nuclear energy.

Much research and assessments are continuously being done towards the operation of the ETS [31][32][33]. The current COVID-19 pandemic seems to influence the MSR [34]. Consensus on the MSR seems to be that it is effective and stabilizing towards the allocation of allowances. However, careful analysis is required to detect unintended side effects.

2.3. Energy System Modeling

An energy system can be defined as a system with the main purpose of providing energy services [35]. In this context, energy services are defined as human needs and desires that require the use of energy. An energy system model is used to model a problem within the energy system context and can use several disciplines from fields such as engineering, economics, operations research, and management science [36]. The main types of energy models can be defined as planning, supply-demand, forecasting, optimization, and emission reduction models [37]. In the scope of this project only planning and optimization models will be realized as they are the types of the models EM-Lab and COMPETES respectively.

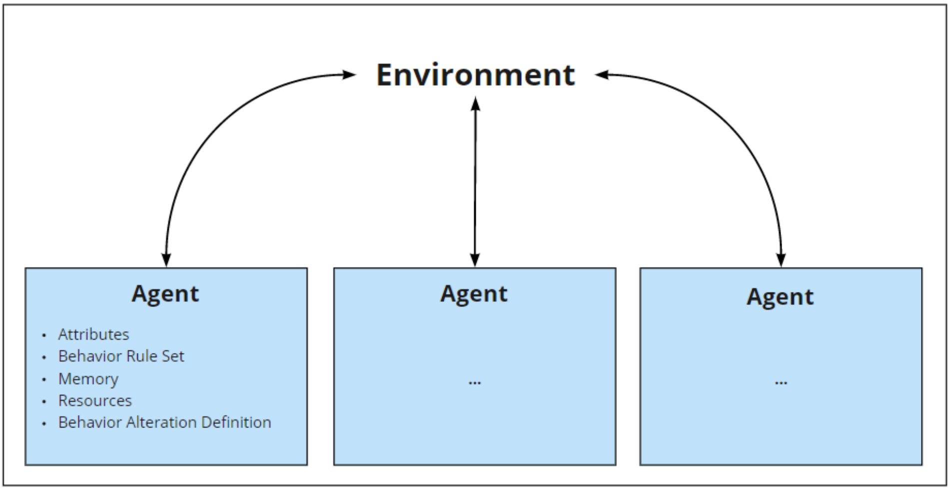
The energy transition and the integration of renewable energy sources have greatly increased the complexity of the energy system models due to the modeling of the challenges like high variability and unpredictability [7]. This high complexity calls for a new approach away from traditional models.

2.3.1. Agent-based Modeling

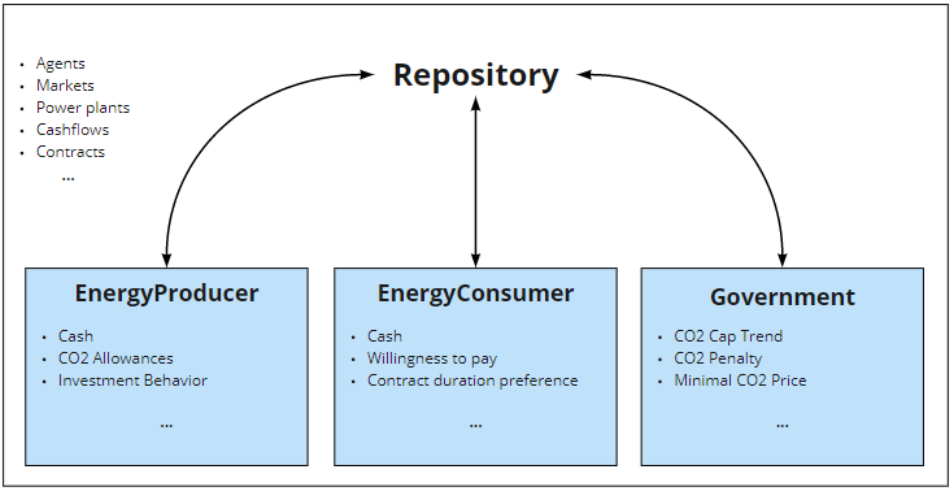
Agent-based modeling is a modeling ideology where separate, autonomous and decision-making entities called agents are central. In this type of modeling agents act according to a specified behaviour often described by a set of rules. These agents are situated in a common environment through which agents can interact with each other. Agents have an objective (not necessarily minimizing or maximizing) and are able to adapt their behavior in order to try reaching their objective. This methodological approach allows research of decision-making in complex systems with multiple interacting actors and a large number of interdependencies [38]. An overview is given in Figure 2.2a.

Agent-based modeling is broadly being applied. Macal gives four reasons for this rise in popularity for agent-based modeling [38]:

1. **Advancing Computational Power**
   The most important reason is the rapidly increasing amount and availability of



(a) Conceptual overview of agent-based model summarizing the properties of the agent.



(b) In this figure the conceptual overview of Figure 2.2a is applied to EM-Lab. Three example agents and their properties are shown. A few examples of properties of the environment, in EM-Lab called the repository, are shown as well.

Figure 2.2: A conceptual overview of agent-based modeling and its EM-Lab application.

computational power. Agent-based models require computational power that was not available some time ago. However, results have shown that traditional modules are not adequate and that the utilization of this computational power is necessary.

2. **Rise in Model Complexity**

The increasing number of interdependencies in modeling call for a modeling approach in which effects of interdependencies can be efficiently explored.

3. **Imperfect Simulation**
   Macal states that some systems have always been too complex to model. Agent-based modeling can provide a view on such a system that more closely represents a real-life scenario while preventing the deterministic modeling of all involved complexities.

4. **Micro-data**
   The improved organization of data and storage of micro-data creates the opportunity for more data utilization in modeling.

Traditional modeling methodologies have been shown not to be adequate anymore. An example is that the well-known predator versus prey model has been shown to be unrealistic because individual motivations are not taken into consideration [39]. Agent-based modeling would be suitable for modeling the dynamic social interactions in such a system.

Energy Applications

Agent-based modeling has been broadly implemented in the energy field. In a literature review the sectors electricity market, consumption dynamics / consumer behavior, policy and planning, new technologies / innovation, energy system and transitions were defined [40].

Zhou reviews the three most popular ABM implementations for electricity markets [41]: Simulator for electric power industry agents (SEPIA) [42], Electricity market complex adaptive systems (EMCAS) [43] and Short-term electricity market simulator-real time (STEMS-RT) [44].

EM-Lab: Energy Modeling Laboratory

EM-Lab is short for Energy Modeling Laboratory and is developed by the TU Delft in order to research the long-term effects of climate and energy policies [12]. Challenges caused by policies in modeling named are cross-policy effects, cross-border effects, imperfect foresight, lumpiness of investment, differences in actor behavior and path dependence [45]. EM-Lab is developed as an effort to explore these complexities.

The application of the conceptual overview of agent-based modeling can be found in Figure 2.2b. An overview of agents can be found in Figure 2.3 [46]. Most importantly the investor is modeled as an agent, named EnergyProducer. Agents take up a Role, which is a set of rules that define the agent's behavior. This behavior entails, for example, investment, decommissioning and willingness to pay.

The most important assumptions made in EM-Lab are as follows:

Agent Names	Complexity	Class
Energy Producer	High	`domain.agent.EnergyProducer`
TargetInvestor	Simple Rules	`domain.agent.TargetInvestor`
PowerPlantManufacturer	Accounting	`domain.agent.PowerPlantManufacturer`
PowerPlantMaintainer	Accounting	`domain.agent.PowerPlantMaintainer`
BigBank	Accounting	`domain.agent.BigBank`
CommoditySupplier	Accounting	`domain.agent.CommoditySupplier`
EnergyConsumer	Accounting	`domain.agent.EnergyConsumer`
Government	Simple Rules	`domain.agent.Government`
NationalGovernment	Simple Rules	`domain.agent.NationalGovernment`
ElectricitySpotMarket	High	`domain.market.electricity.ElectricitySpotMarket`
CommodityMarkets	Simple Rules	`domain.market.electricity.CommodityMarket`

Figure 2.3: Entities modeled as agents in EM-Lab, their complexity and Java class name. [46]

- **Load duration blocks**
  In order to reduce computational complexity, the amount of hours in the load duration curve is grouped and averaged.
- **Innovation is limited to learning**
  New technologies are not introduced during the simulation. Existing technology characteristics improve gradually over time.
- **Biomass 100% CO2-neutral**
  Biomass is realized as other CO2-neutral energy sources: it produces no CO2, but is more expensive.
- **Limited Generator Characteristics**
  In order to maintain typical or plausible estimated numbers within the model generator characteristics, like capacity or efficiency, are limited per technology.

EM-Lab has been developed with a core that is extendable by separate modules. The core is meant to be the starting point for all projects and contains the required short-term operations in the electricity sector like the electricity market and dispatch. Modules have been developed to research the EU ETS interactions with the electricity sector and to research the capacity mechanisms in the EU.

Table 2.2 provides an overview of properties of EM-Lab and comparison to COMPETES.

Capacity Market
In 2016 EM-Lab was extended by Bhagwat in his doctoral thesis with a module to study the effect of different capacity mechanisms, like the annual capacity market, the forward capacity market, and the strategic reserve [24]. The implementation of capacity mechanisms has long been a subject of debate. This research explores the question of whether in a transition to low CO2 economy capacity mechanisms

2

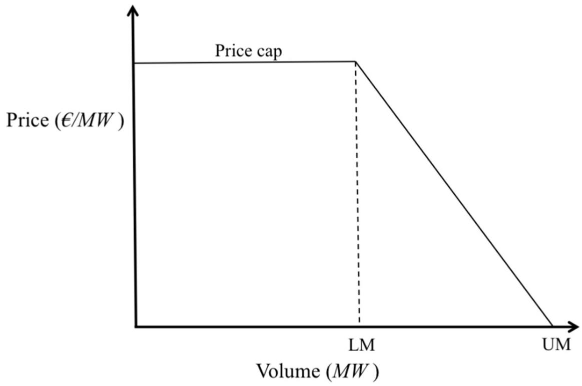


Figure 2.4: EM-Labs sloping demand curve provided by Bhagwat [47]. This is the returned price by the SDC object as described in Algorithm 3. LM and UM refer to Lower Margin and Upper Margin respectively.

are an effective tool towards retaining long-term generation adequacy.

Relevant from this doctoral thesis is the annual capacity market implementation. This was modeled after the New York Independent System Operator (NYISO) [24]. In this model, the generators bid their unforced capacity for the price required to break even that year. These bids are sorted in ascending order and matched against the sloping demand curve. This sloping demand curve can be found in Figure 2.4. In order to generate the sloping demand curve, the user sets the Installed Reserve Margin (IRM), the capacity market price cap, the upper margin and lower margin. Similar to the ICAP curve described in Figure 2.1, the generators do not determine the price this way.

The use of a sloping demand curve forces the change in capacity market price to be a slight one. This means that implementing the curve decreases price volatility.

The results indicate that the presence of a yearly capacity market can increase adequacy in a system with a high percentage of renewable energy sources. User defined parameters are highly influential and should be set right. It is mentioned that the cost to the consumer is sensitive to the growth rate of demand, but the market would remain effective.

CO2 Market

In 2015, EM-Lab was extended by Richstein in his doctoral thesis to investigate the interactions of the EU ETS and the electricity sector [48]. This doctoral thesis was multifold: it entailed the effect of the ETS on general and national price caps, the effect of a Market Stability Reserve (MSR) and hedging of CO2 credits and finally the possibility of subsidized renewable energy generation alongside an ETS.

The most impactful addition to EM-Lab was the development of the CO2 market.

This market was developed to be interlinked with the core's electricity spot market. A conceptual overview for the CO2 market clearing algorithm is given in Figure 2.5. The markets are interlinked as every iteration in clearing the CO2 market calls the electricity spot market. The clearing algorithm starts by taking an initial CO2 price. It will clear the electricity spot market with this CO2 price and the electricity spot market in three years' time with an extrapolated CO2 price. It is assumed that the CO2 price will grow with 5% every year. This future run is taken into consideration to prevent price volatility. Once completed, the sum of the emissions of both clearings are matched with the cap of both years, adjusted by the banking target. If they are approximately equal, the CO2 price is set. If not, the price is raised, and the next iteration clears the markets again.

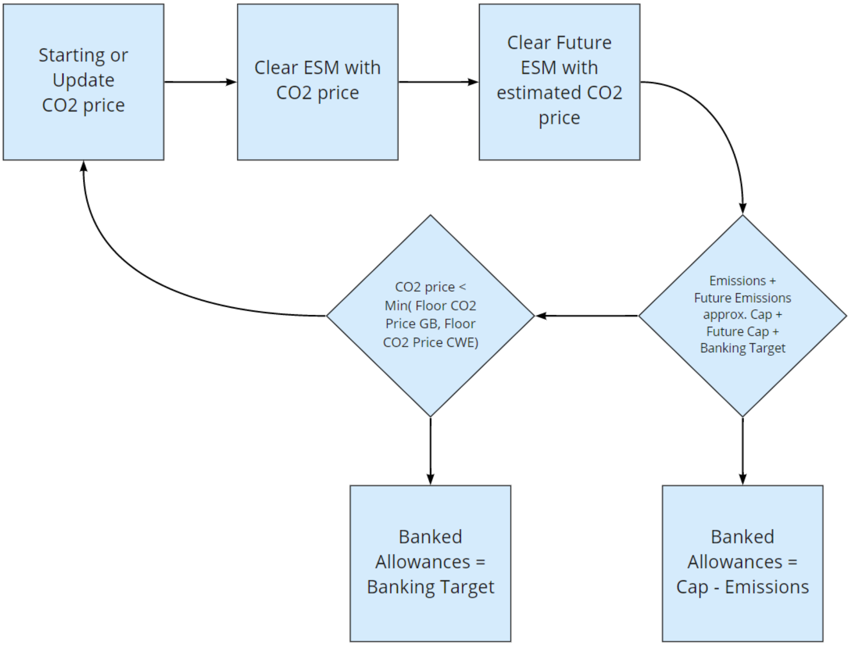


Figure 2.5: A conceptual overview of EM-Labs interlinked CO2 market and the electricity spot market as described by Richstein [48].

The thesis concludes that modifications of the ETS will have no or a bad impact on reaching the emission targets or economic efficiency. Agent-based modeling is seen to be an effective tool for this research. It was concluded that a national ETS price floor had little effect, depending on the size of the country, while a general price floor seemed to reduce price volatility. Backloading and the revision of the MSR performed well, reducing overall price volatility.

2

2.3.2. Optimization Modeling

In optimization modeling the minimization or maximization of an objective function under certain constraints is central [37]. Optimization modeling has proven to be a useful methodology to study necessities to achieve a specified goal under the constraints of a scenario. The three key elements in optimization modeling are the objective function, the decision variables and constraints. Under the constraints, the objective function results in a value which is to be minimized or maximized. After the optimization, the model outputs the optimal result of the objective function and the values of the decision variables through which this optimum is achieved. The constraints enable the creation of an environment in which this function must be minimized.

Examples of objective functions are the minimization of system cost [49], the minimization of the annual electricity generation costs and total emissions of three greenhouse gases [50], and the maximization of the net present value [51].

COMPETES

COMPETES is an optimization model developed by TNO that seeks to minimize total power system costs [11]. It does so under multiple constraints like power balance, generation capacity and cross-border transmission constraints. COMPETES contains detailed information on flexible power generating technologies in the EU and on trading capacities and policies across the EU. Table 2.2 provides an overview of properties of COMPETES and EM-Lab.

Table 2.2: Overview of properties of EM-Lab and COMPETES.

	EM-Lab	**COMPETES**
Developer	TU Delft	TNO
Language	Java	AIMMS
Framework	AgentSpring	AIMMS
Time resolution	hourly, yearly	hourly
Policy Implementations	CO2 Market, CO2 tax, subsidies	None
Time horizon	Any amount of years	1 year

COMPETES possesses a very sophisticated dispatch algorithm including heat, hydrogen (H2), storage and renewables. The investment module can invest in these technologies, transmission lines and can recycle gas pipelines for hydrogen usage.

2.4. Model Coupling

Model coupling or co-simulation is a relatively new subject in the scientific world, and it has been applied in multiple engineering domains. It has been shown to be an effective method for creating more accurate models [52][53].

2.4.1. Soft-Linking

Within the co-simulation context, soft-linking is a methodology in which models are combined as separate models with an exchange of information. In previous work, the term "soft-linking" has shown to be ambiguous. Some papers only refer to information exchange between two separate models as soft-linking [54][55][15]. Other papers refer to the iterative process in which two models are engaged and the convergence of both models in central parameters as soft-linking [56][53]. Some papers make the distinction between uni-directional and bi-directional soft-linking. This is referred to as the direction of information between the models [14] or not as a direction but as an indication of directness of linkage between the models [53].

To mitigate the ambiguity of the term soft-linking, the following definition is proposed and is used in the scope of this report. Soft-linking refers to the coupling of multiple separate autonomous models by providing the information exchange of parts that enables the individual model to run with parameters provided by the other model. The distinction of uni-directional and bi-directional in this definition refers to the direction of information flow: in uni-directional soft-linking one model depends on parameters provided by another model and in bi-directional soft-linking this dependency goes both ways. In practice, bi-directional soft-linking requires some form of convergence based on shared parameters or some predefined condition to indicate the end of the run. Soft-linking allows the models to run in their respective time spans. In contrast, hard-linking refers to a continuous information exchange during the runs of each model.

In a novel cooperation, the challenge arises that somehow individually developed complex models must reach a common goal. Soft-linking can be a remedy for such a challenge, as creating a new model is costly and means discarding the individual models which have been developed by multiple people over multiple years. Additionally, the models can be coupled while keeping their core intact, regardless of the type of model. For example, an optimization model can be coupled with an agent-based model.

To answer the first research sub-question defined in Section 1.3, the following requirements for soft-linking have been extracted from the definition:

1. **Information Exchange**
   The definition of soft-linking states that an exchange of information between the coupled models must be present. Otherwise, the coupled models would run in their original (uncoupled) state. This exchange can be uni- or bi-directional.
2. **Information Mapping**
   In addition to the exchange of information, the model must have an interpretation of the exchanged data.
3. **Timing Definition**
   Soft-linking requires a definition of how the models run in the coupled sched-

ule respecting their time granularities. This schedule indicates when information is exchanged. For soft-linking at least one model must be able to run completely. If there is a continuous information exchange before the models run completely, this would be known as hard-linking.

2

2.4.2. SpineToolbox

Part of this project is the exploration of the SpineToolbox software kit [57]. SpineToolbox is developed by the Technical Research Centre of Finland (VTT) [58] specifically for managing and combining multiple energy system models. It aids in transparency because of the visualization of the implementation of the conceptual model. SpineToolbox provides a layout for SQL Databases which is central in the approach for this soft-linking.

SpineToolbox can execute Python, Julia, GAMMS and other executables in an enclosed environment with intercommunication. A flow through these processes must be specified. SpineToolbox can pass inputs and outputs between the processes. Results can be imported and exported from and into the Spine database (SpineDB), providing the basis for a common ontology.

3 Soft-Linking Methodology

This chapter describes the methodology towards the soft-linking of COMPETES and EM-Lab. This is done through the satisfaction of the requirements of soft-linking defined in Section 2.4.1: data exchange, data mapping and time scheduling. The data exchange is defined in Section 3.1. The data mapping and time scheduling is described in Section 3.2. The individual model responsibilities have been summarized in Table 3.1. Lastly, the data organization in the soft-linking and data transformations are discussed. These elements form the answer to the first research question and its sub-questions as defined in Section 1.3.

3.1. Coupling Potential and Constraints

As discussed in Section 2.4.1, soft-linking offers a methodology that can combine the strengths of multiple models. Section 2.3.1 and Section 2.3.2 go in depth on the workings of both models. To satisfy the soft-linking design requirements defined in Section 2.4.1, the coupling points are identified. This way the data exchange, data mapping and timing schedule can be defined.

In order to achieve a stronger model, design choices must be made regarding which model functionalities will be used. This section elaborates on the opportunities available in soft-linking. However, design choices in both models naturally lead to challenges which make coupling an intricate task.

The model responsibilities have been summarized in Table 3.1.

3.1.1. Time Resolution

An important difference originating from the core of agent-based and optimization models is time resolution. In an optimization model like COMPETES the model must complete its run for it to produce accurate results. If interrupted, the model has not reached an optimum and the data is unusable. EM-Lab produces results iteratively as agents make and log decisions. In terms of coupling, being able to continuously import and export data is desired. This makes EM-Lab more suitable than COMPETES and led to the early decision to have EM-Lab run in parts before and after the full COMPETES run.

3.1.2. Investment Decision-Making

COMPETES bases its investment decisions on the optimization of total power system costs. However, this is not an accurate representation of real-life investment decision-making as investors do not always have perfect sight of the power system costs and future demand. In addition, real-life investors are not all-knowing entities capable of minimizing total power system costs. In EM-Lab, investors are modeled as agents trying to create the highest net present value for themselves. They do so with their own estimations and understanding of the environment which is imperfect. This imperfect investment is an important property of EM-Lab and could be beneficial regarding modeling future power system investments, as more resemblance with a real-life scenario is desired.

Both EM-Lab and COMPETES are capable of making investment decisions, thus a design choice had to be made which model would make these decisions. EM-Lab's investment forecasting depends on future runs of the dispatch in order to estimate whether an investment would be profitable. This requires multiple dispatch runs per agent per investment iteration. As stated, COMPETES has a highly complex dispatch algorithm and is very computationally intensive. So, if EM-Lab was to make the investment decisions this would require COMPETES to run multiple times every iteration. This would mean an exponential growth in run time and would make the soft-linking impossibly complex. If COMPETES were to make the investment decisions, the model would still have high computational complexity, however this complexity now originates from the detailed model of the system. Therefore, COMPETES is chosen to make the investment decisions. In order to retain imperfect investment EM-Lab provides imperfect information to COMPETES in the coupling.

In EM-Lab's investment module the investment takes into consideration the amount of time between the investment decision and the moment the generator is operational. Naturally this differs per technology: a nuclear power plant has more safety regulations and thus more needed permits as opposed to a gas turbine. COMPETES does not take this time into account and assumes that investments are operational in the same year. Regarding this 'build time', two design choices were made. First, in order to have accurate results, an iteration must run COMPETES twice: once for dispatch where investments are disabled, and once for determining investment decisions. Second, the investment module runs for the current year with an added time horizon which has the length of the longest technology build time. Investment decisions from that year will be implemented in the current year with the added technology build time.

Table 3.1: Responsibilities of EM-Lab and COMPETES summarized.

EM-Lab	**COMPETES**
Capacity Market	Investment
CO2 Market	Decommissioning
	Power Optimization
	Economic Dispatch

3.1.3. Dispatch

In order to reduce the complexity of the model and improve run times, EM-Lab down-samples and averages the hourly duration curves into 20 load duration blocks. While effective in improving run times, this is costly regarding information complexity. EM-Labs dependency on this structure is inflexible and not suitable for expansion of renewable technologies or energy storage. COMPETES has a very elaborate dispatch algorithm incorporating e.g., high variability. Many renewable technologies besides the traditional ones, like heat, storage, and H2 technologies, are also implemented. In addition, COMPETES can be run for the entire EU. Therefore, the COMPETES dispatch is used.

3.1.4. Market Modules

While having very sophisticated operational modules, COMPETES has few policy modules and no market modules (apart from the energy-only market). In order to extend the applications for this soft-linking, policy and market modules should be coupled. EM-Lab is designed specifically for studying policies and markets and thus could extend COMPETES in that regard. EM-Lab has been extended with a CO2 market and capacity market in the past and these make excellent candidates for coupling with COMPETES, as it would allow COMPETES to endogenously calculate the capacity support and the CO2 price. COMPETES has implemented a CO2 price which is a model input and is added directly to the operational costs of the generators, but it does not model the entire complexity of a CO2 market.

3.2. Conceptual Model

After the coupling potentials and challenges had been identified, a conceptual model was made which can be found in Figure 3.1. A conceptual model is an abstraction and representation of the desired soft-linking in order to incorporate the strengths as described in Chapter 2. In this conceptual model, the main roles identified are the CO2 and capacity market, hosted by EM-Lab, and the dispatch and investment module, hosted by COMPETES. This figure shows the execution of 1 year, which has been further visualized in the timing diagram found in Figure 3.2. The timing diagram also shows the direction and interpretation of data.

The timing diagram shows the implemented conceptual model. Therefore, for example, the COMPETES elements are shown executing in different time steps. In theory, this could be in parallel. Unfortunately, practical limitations of AIMMS, the COMPETES programming language, do not allow this.

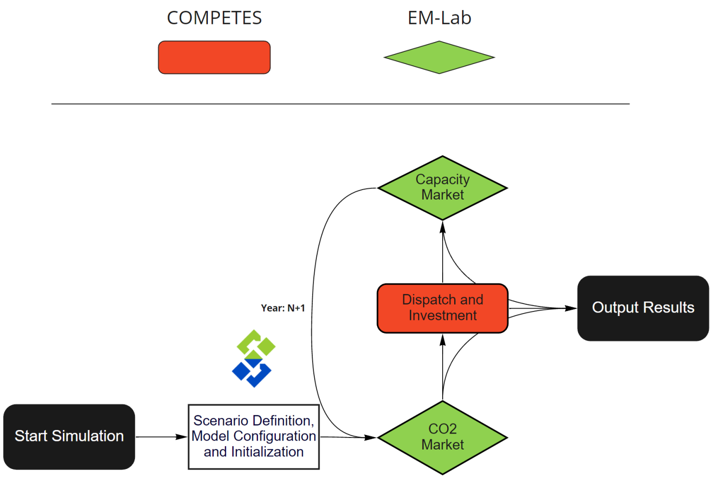


Figure 3.1: The conceptual model describing the module sequence and information direction. The flow indicates EM-Lab's CO2 Market is activated first. This module provides a CO2 price to COMPETES which will run the dispatch and investment modules. Finally, the Capacity Market can run in addition to the COMPETES operational modules.

3.2.1. CO2 Market

EM-Lab's CO2 market module outputs a single CO2 price for the year the model is executing. In order to determine this price, the generator's willingness to pay (WTP) must be determined. From the previous year's dispatch run the generators yearly operational profits and emissions are determined. Profits are calculated from spot market revenues and marginal costs. Section 4.1.1 elaborates on these calculations.

As EM-Lab runs for the Dutch power plants, the emissions are matched with an estimated CO2 cap for the Netherlands. This is an estimation as the ETS CO2 cap is set EU-wide. This CO2 price, which is calculated for the Dutch CO2 market, is passed on COMPETES which applies it to the EU. The emissions are taken from the COMPETES output. COMPETES takes more details, like plant temperature and dynamic efficiency, into account regarding the calculation of emissions which are features not present in EM-Lab, and therefore not able to be mapped. If the plant's emissions are not present in the COMPETES output, EM-Lab attempts to calculate this amount.

COMPETES adds the CO2 price to the fuel costs and the unit commitment module is run. For the investment module the CO2 price is extrapolated with a growth rate, assuming the price will grow over time. These processes output investment decisions, decommissioning decisions, hourly balance, hourly nodal prices and generation allocation. This output is used in next year's determination of the CO2 price.

3.2.2. Capacity Mechanism

Unlike the CO2 market, COMPETES has no direct interpretation of capacity mechanisms available as it has no interpretation of cash flow. The main purpose of implementing a capacity mechanism is to affect investment and decommissioning decisions. Revenues from these mechanisms are therefore implemented as fixed operations and maintenance (O&M) cost reduction and, if these costs reach 0, as Capital Expenditure (CAPEX) reduction. CAPEX costs in COMPETES are yearly and are implemented for the year the investment module runs. The fixed O&M costs are per technology and thus implemented over all years. These reductions are implemented per technology that receive revenue from the capacity mechanism. Through this method COMPETES retains the investment incentive gained through the capacity mechanisms and could prevent decommissioning.

In order to calculate the bids in the capacity market the power plant dispatch plans and profits are necessary. From the spot market revenues, marginal costs and fixed O&M costs it can be determined if the plant breaks even that year. The bid placed is based on the compensation necessary to break even. This calculation is specified and elaborated on in Section 4.1.2.

3.2.3. Dispatch

The dispatch modules of EM-Lab and COMPETES differ in organization. Section 3.1 establishes that COMPETES dispatch will be run and that the results must be translated to EM-Lab. The COMPETES output is simple and is comprised of two tables: hourly nodal prices and hourly unit generation. The hourly nodal prices table shows the price per MWh per country per hour. The hourly unit generation provides the amount of MWh per unit per hour.

EM-Lab uses what is called Power Plant Dispatch Plans. Such a dispatch plan is comprised of an amount of generation, a price, a status, a time and the market for which it is created. Initially, the dispatch plan is used as a bid in the market awaiting confirmation from the market clearing algorithm. After the clearing of the market the dispatch plan describes when, how much and for what price it is going to generate.

In the original EM-Lab a dispatch plan was created per load duration block. Since in the conceptual model the models are run per year, the coupling produces one dispatch plan for the entire year. The amount of generation is summed, and the price is averaged per unit. This way, if EM-Lab determines the yearly amount of revenue, this single dispatch plan provides all information.

The dispatch depends on the marginal costs of the plants which is largely determined by the fuel costs. In EM-Lab the fuel prices are generated through a triangular trend. In this trend the numbers vary stochastically based on the current year the model runs. In order to have a complete coupling this trend is used to output prices to the COMPETES input, which takes the fuel prices per month per year.

3.2.4. Investment and Decommissioning

In Section 3.1 it is established why COMPETES will generate the investment and decommissioning decisions. The section also describes the conflict in COMPETES and EM-Lab their interpretations of plant statuses. This conflict also creates a challenge in the coupling of investment and decommissioning decisions.

3

The COMPETES investment module produces VRE and regular investments, described by the unit and its properties, and decommissioning decisions, described by a unit name which is to be decommissioned. In the COMPETES data structure, plants are added twice: once a regular entry and a copy with '(D)' in the title, the status set to 'DECOM' and the year for which the plant is to be decommissioned.

Adding a new plant means that this decommissioned copy must be added as well. For the decommissioning year a large difference is chosen (current year + 40) as due to the complexity and high run time of the model plants will never reach this year. As for the operational version, as described in Section 3.1, the year of operation is the current year + the build time of the technology of the plant. In case of decommissioning, the year of the '(D)' version is set to the year of when the investment module is run.

This still begs the question of how EM-Lab structurally handles this difference in data organization. The answer is with an extra pre-processing step in the coupling before EM-Lab runs. This step checks the years of the '(D)' plants and will set the status of the actual plant to 'DECOM' if decommissioned. EM-Lab only handles plants which are operational. This 'translation step' is further elaborated on in Section 3.3.

To prevent investment decisions being immediately available in the capacity market (which is run directly after the COMPETES investment module) plants are added to EM-Lab with the status 'DECOM'. The pre-processing step, which is executed in the next year, sets the status to 'OPR' when the generator is online.

3.3. Data Organization

EM-Lab and COMPETES differ in data organization and requirements. This is natural as they differ in capabilities. The difference in data organization results in the need for 'translation' steps in the soft-linking, referring to the transformation of data from one model to the other. The difference in data requirements result in a 'split' in the data for soft-linking. This soft-linking's data organization showing this split is illustrated in Table 3.2, where the data objects required by the individual models and both models are displayed.

Table 3.2: This table shows the data organization of the soft-linking. The table shows objects which are further defined with specific data. E.g., 'Fuel' contains properties like the fuel name, energy density, CO2 density and more. The column 'Shared' indicates the data used by both EM-Lab as well as COMPETES. The complete data structure is discussed in further detail in Appendix B.

EM-Lab	**Shared**	**COMPETES**
Market	Bus	Biomass Potential
Energy Producer	Country	Demand Response (regular, EV, H2, Heat, Shifting)
Government	Fuel	H2 System, Technologies and Storage
National Government	Fuel price	Historic Nuclear Availability
	Hourly Demand	Hourly Profiles (DR, EV, H2, Hydro)
	(VRE) Technologies	HVDC Investments
	VRE Capacities	New Technologies
	NL Installed Capacity (regular, RES and decentralized)	Overnight Cost (OC) (CAPEX)
		Trading Capacities
		Unit Commitment
		VRE FLH and Loadfactors
		HVDC Overlay
		Installed Capacity Abroad (regular, RES)
		Storage

The data organization described is further reflected in the SpineToolbox implementation, as there is a SQL database behind both models. The data is mapped to these databases from their individual initialization data set and from the shared data set. The SpineToolbox implementation is further specified in Section 4.2.

In the soft-linking, there are multiple places where a translation step is needed. Naturally, there is a 'EM-Lab to COMPETES' and a 'COMPETES to EM-Lab' step. In addition, there is a need for an 'EM-Lab Pre-Processing' step due to structural differences between EM-Lab's and COMPETES' data organization. For result purposes, in some cases COMPETES works through aggregates: as an example, a power plant is decommissioned by adding another instance of that power plant with status 'DECOM' and by setting its capacity to negative. This way, when COMPETES aggregates for a certain year, the amount of capacity available will have decreased. This organization forms a challenge as EM-Lab integrates single power plants with a 'current' status. If this status is set to operational, the plant is operational and the other way around. So, a pre-processing step is required in order to set the correct statuses.

In the EM-Lab to COMPETES step the CO2 price and capacity market revenues are passed to COMPETES. While EM-Lab outputs a single yearly CO2 price, COMPETES requires the CO2 price for every month of the year. In order to run the future investment, the CO2 price is extrapolated. Regarding the capacity market revenues, in addition to the implementation described in Section 3.2.2, the difference in units (Euro / MW vs. Euro / kW) should be realized. As the CAPEX are implemented per year in COMPETES, if there is a CAPEX reduction this should be implemented for the future year as well.

COMPETES generates hourly unit commitment, hourly spot market prices, investment and decommissioning decisions and CO2 emissions. The unit commitment and spot market prices are averaged and summed per unit and exported to EM-Lab. The CO2 emissions are translated to tons CO2 and also exported. Finally, the investment and decommissioning decisions are simple implemented as power plants in EM-Lab, considering the pre-processing step earlier described.

These issues are solved programmatically and elaborated on in Chapter 4. In addition, the complete data structure is discussed in further detail in Appendix B.

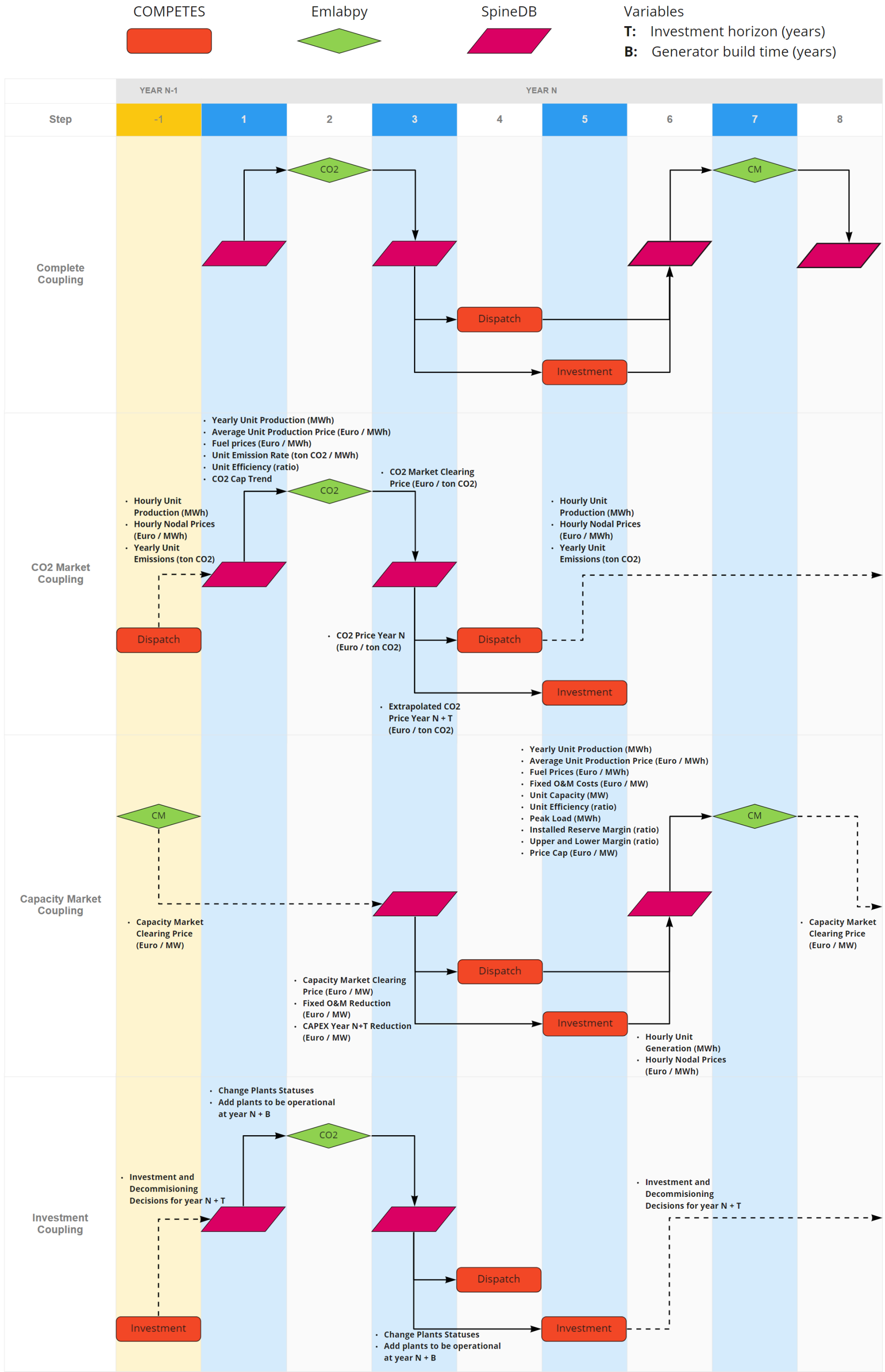


Figure 3.2: This figure describes the sequential execution of the parts in the soft-linking for year N. All data passed is in regard to year N, unless stated otherwise. First the CO2 Market runs, then the operational parts from COMPETES and finally the Capacity Market. Afterwards the model progresses to the next year. Note that MW always refers to installed capacity.

4

Implementation

The main challenge of this project emerges from the implementation of the design choices made in Chapter 3. This chapter will dive into the coding specifics and will provide pseudo-code for the most important developed algorithms. The usage of SpineToolbox is also shown and elaborated on. This chapter constructively answers the second research question defined in Section 1.3.

4.1. Python Implementation EM-Lab

Early in the developing process it became clear that coupling the original Java version of EM-Lab would be too big a task for the scope of this project because of multiple reasons. Most importantly, there was a heavy dependency on an external Java module which acted as a wrapper for the entire execution. This not only made it problematic to run separately, but also to break open the model and extract the single individual market modules.

Since the coupling, as defined in Chapter 3, only involves smaller individual parts of EM-Lab the most evident approach was to simply recreate these parts. For this task Python was chosen. Python is a well-known open-source language with a large and active developer community. Through this dedicated community, multiple modules are easily and readily available. This made it easy to verify that Python was suitable. More importantly, the database API from SpineToolbox, which is central in the soft-linking, is developed in Python.

The Python implementation, from here on referred to as Emlabpy, functions as a wrapper for the SpineToolbox database (SpineDB) and executes modules according to information passed from SpineToolbox. Emlabpy is developed in such a way that modules can be developed and run separately, as expansion in the future is likely. A conceptual overview is given in Figure 4.1. The conscious decision was made to split the modules from the database access. In this implementation, it is relatively easy to create a new interpreter. This would be necessary if the application moved away from SpineToolbox.

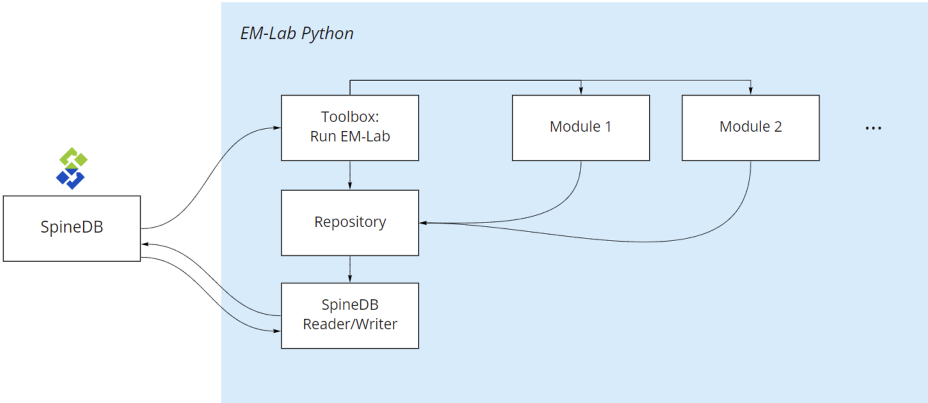


Figure 4.1: The conceptual Emlabpy model where the largest parts are roughly described. On the left, SpineDB can be seen as the outside database. The initialization is done by SpineToolbox creating the repository and initializing the modules. The SpineDB Reader/Writer afterwards handles all communication between the SpineDB and Emlabpy.

For this project two modules were developed: the CO2 market and the capacity market. In the conceptual model defined in Section 3.2 the flow of information and dependencies are determined. This section discusses the algorithms and their implementation. The implementations are discussed with the use of mathematical symbols. An overview of these symbols is found in Table 4.1.

The full documentation can be found in Appendix A.

4.1.1. CO2 Market

The original EM-Lab CO2 market implementation was heavily interlinked with the electricity spot market, as discussed in Section 2.3.1. The code would execute multiple future electricity spot market clearings for different CO2 prices and try to converge to an equilibrium where the amount of emissions match the CO2 cap. As discussed in Section 3.1, COMPETES does not fit in this implementation due to its high complexity from having to run the dispatch multiple times. An approach to determining the CO2 price without iterations was therefore necessary.

Inspired by Richstein's approach [48] and following basic economics, the following approach was created. In this approach, the CO2 price p_{CO2} in Euros per ton CO2 is based on the willingness to pay p_{WTP} per plant in Euros per ton CO2. A plant's p_{WTP} is based on the amount of operational profits in Euros pr_{opr}, determined from the hourly nodal prices and generation, and its emissions in ton CO2

Table 4.1: An overview of mathematical symbols used in Chapter 4.

Symbol	Value and Unit
E	Yearly Emissions (ton CO2)
Er	Generator emission rate (ton CO2 / MWh)
G	Yearly Generation (MWh)
IRM	Installed Reserve Margin (ratio)
LM	Lower Margin (ratio)
LMV	Lower Margin Volume (MW)
R	Yearly Revenue (Euro)
R_{ESM}	Yearly Electricity Spot Market revenue (Euro)
UM	Upper Margin (ratio)
UMV	Upper Margin Volume (MW)
c	Yearly Generator Cost (Euro)
$c_{O\&M}$	Yearly Generator Fixed Operation and Maintenance costs (Euro)
c_{mc}	Yearly Generator Marginal costs (Euro)
d	Yearly Demand (MWh)
d_{peak}	Demand peak (MWh)
g	Generator
g_{cap}	Generator installed capacity (MW)
m	Market Object
p	Price (Euro)
p_{CM}	Capacity Market clearing price (Euro / MW)
p_{CO2}	CO2 Market clearing price (Euro / ton CO2)
p_{WTP}	Willingness to Pay (Euro / ton CO2)
p_{cap}	Capacity market price cap (Euro / MW)
pr_{opr}	Yearly Generator Operational profits (Euro)
v	Capacity volume (MW)
y	Year

from last year E. If pr_{opr} is divided by E then this number indicates the amount of money the plant can pay per ton CO2 before it stops being profitable.

The algorithm has been described in Algorithm 1. Because the algorithm depends on previous data, the first time the model is run the algorithm takes a predefined p_{CO2}. In this case the number 21.165 Euro / ton CO2 is taken which is in line with previous real European CO2 prices.

If this iteration is not the first, the algorithm will find the CO2 cap E_{cap} through the function CO2_Cap_Trend. This function linearly decreases the CO2 cap by a user-defined value and is set at the EU ETS regulation decrease. Afterwards, the list of plant operational profits is calculated by determining the spot market revenues R_{ESM} and subtracting the plants marginal costs c_{mc}. R_{ESM} is determined from the dispatch plans and average unit pricing as discussed in Section 3.2.3. c_{mc} is based on the fuel costs, which is a triangular trend defined by the user, and the plant efficiency.

In this implementation, the Dutch plants in EM-Lab define the CO2 price. Normally, the CO2 cap is EU-wide, and an estimation must be made to be able to set a Dutch CO2 cap. Looking at the emission results from the COMPETES only run, the CO2 cap was set at 10 Mton CO2.

After the profits have been calculated, the list of plant emissions E_{plants} is extracted from COMPETES. If this value is not present for a plant, it will determine the amount of emissions through the plants generation G_{plants} and their respective emission rate Er_{plants}. The total amount of generation is in MWh and the emission rates are user-defined per technology in ton CO2 / MWh. Finally, p_{WTP} is calculated by dividing the profits with the emissions.

The list of p_{WTP} is now sorted in descending order and through the merit order p_{CO2} is determined. The CO2 price is determined through the intersection of the merit order and the CO2 cap as illustrated in Figure 4.2.

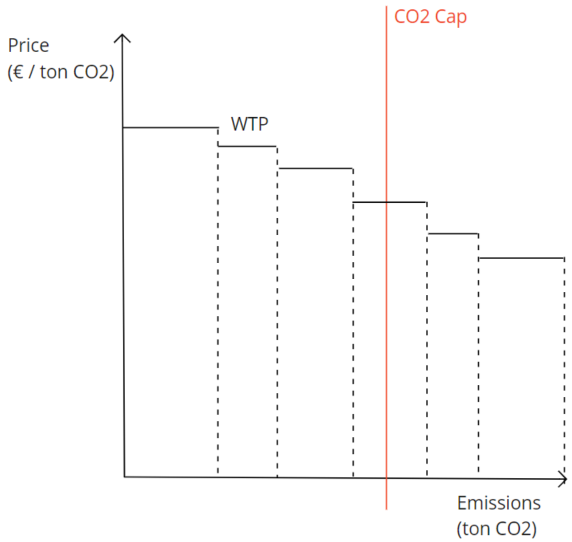


Figure 4.2: Emlabpys CO2 market module determines the CO2 price through the intersection of the merit order and the CO2 cap. The graph shows the willingness to pay (WTP) of the plants. These prices are sorted in a descending manner and the amount of emissions are summed.

4.1.2. Capacity Market

The general capacity market working has been described in Section 2.1. This project's implementation has been heavily inspired by the EM-Lab implementation developed by Bhagwat [24]. The algorithms are split into the bidding (Algorithm 2) and clearing (Algorithm 4) of the market. The clearing algorithm uses what is

Algorithm 1 Pseudo code for Emlabpy's algorithm that determines the CO2 price.

```
for all CO2 Market m_CO2 do
    if Current Clock Tick y = 0 then
        p_CO2 = 21.165, based on previous real data
    else
        E_cap = m_CO2 → Country → Government → CO2_Cap_Trend(y)
        pr_opr = R_ESM − c_mc
        E_plants = E_COMPETES if E_COMPETES ≠ 0 else G_plants * Er_plants
        p_WTP = pr_opr / E_plants
        Sorted_WTP = sort p_WTP descending

        p_CO2 =max(p_WTP)
        Var E_total = 0
        for all Operational power plant g do
            if E_plants(g) > 0 then
                if E_cap ≥ E_total + E_plants(g) then
                    E_total+ = E_plants(g)
                    p_CO2 = p_WTP(g)
                else
                    break
                end if
            end if
        end for
    end if
end for
```

4

called the sloping demand curve (SDC), which has a separate definition found in Algorithm 3.

The bidding behavior is defined in Algorithm 2. After COMPETES has run in the conceptual model, the capacity market will run using the output of the dispatch (hourly prices and hourly unit generation). First, the plant profits pr are calculated by subtracting the marginal costs c_{mc}, determined by fuel prices and unit efficiency, and the fixed O&M costs $c_{O\&M}$ from the electricity spot market revenues R_{ESM}, determined by the dispatch. If the plant is making profit, the plant acts as price-taker and the bid price p_{bid} is 0. If not, the plant sets p_{bid} to what is necessary for the plant to break even. This is calculated by dividing the profits pr by the generator capacity g_{cap}.

In the algorithm also a peak segment dependent availability is taken into consideration. This is a factor of around 8 percent for renewables and has the effect that renewables must place higher bids and likely will not participate in the capacity market. This value is an estimation done to consider variability and intermittency [24]. It is argued whether subsidized generators should participate in the capacity market [59].

Algorithm 2 Pseudo code for Emlabpy's algorithm that determines the power plant bids in the capacity market.

for all Operational power plant g **do**
  $pr = R_{ESM} - c_{mc} - c_{O\&M}$
  if $pr \leq 0$ **then**
    $p_{bid} = -1 * pr/(g_{cap}*$peak segment dependent availability$)$
    Bid g_{cap} for p_{bid} Euro / MW
  else
    Bid g_{cap} for 0 Euro / MW
  end if
end for

The clearing of the capacity market is done according to the sloping demand curve (SDC) for which the pseudo code can be found in Algorithm 3. The SDC requires the Installed Reserve Margin IRM, Lower Margin LM, Upper Margin UM, the peak load d_{peak} and the price cap p_{cap}. These values, except for d_{peak}, are defined by the user. d_{peak} is programmatically taken from the dispatch.

In the initialization of the SDC object the margins are translated from ratios to volumes, taking into consideration the IRM. The SDC object has one function: get_price_at_volume. This function uses if statements to determine and return the price as shown in Figure 2.4.

Algorithm 3 Pseudo code for Emlabpy's sloping demand curve used in the capacity market.

Require: $IRM, LM, UM, p_{cap}, d_{peak}$
  $LMV = d_{peak} * (1 + IRM - LM)$
  $UMV = d_{peak} * (1 + IRM + UM)$

function get_price_at_volume(volume v)
  $sl = p_{cap}/(UMV - LMV)$
  if $v < LMV$ **then return** p_{cap}
  else if $LMV \leq v \leq UMV$ **then return** $p_{cap} - sl * (v - LMV)$
  else if $UMV < v$ **then return** 0
  end if
end function

For the clearing of the capacity market, the peak load d_{peak} is extracted and the SDC object is created. The merit order is created by sorting the bids according to their price p_{bid} in an ascending manner. Per sorted bid p_{bid} is checked whether it exceeds the price provided by the SDC. The intersection of the SDC and the merit order is the capacity market clearing price p_{CM}. Finally, the clearing sets the bid status s to whether the bid has been accepted, partly accepted or has failed.

The clearing algorithms matching with the SDC has been illustrated in Figure 4.3.

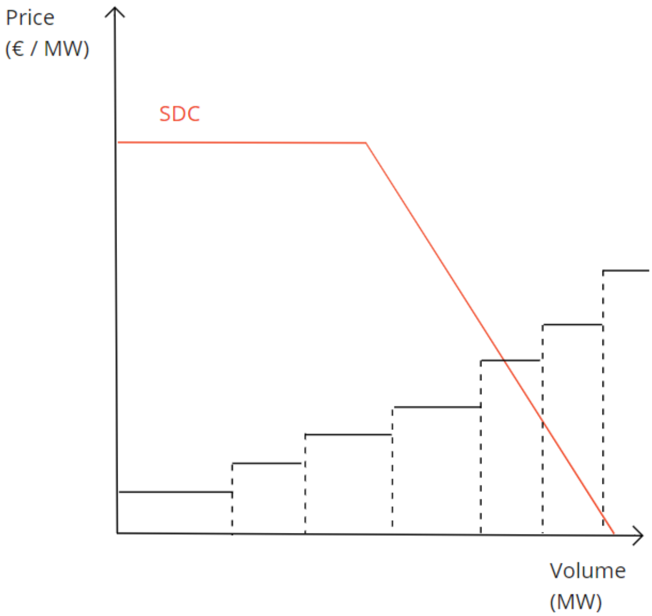


Figure 4.3: Emlabpy's capacity market is cleared by sorting the bids in an ascending manner. Afterwards, the bid capacities are summed and the capacity market clearing price is set at the intersection of the sloping demand curve (SDC) and the bid price.

4.2. SpineToolbox

In Section 2.4.1 it was discussed how in the essence of soft-linking the models run as much as possible in their own environment and without changes to their cores. It can be concluded in such a case that the overlapping software has to work as a mediator, exchanging information between models. SpineToolbox is being developed for this purpose in addition to bringing oversight in information flow and execution sequence.

Section 3.3 briefly mentions the SpineToolbox implementation. Behind the working of both models is a SQL database (SpineDB). These SpineDBs create the backbone of the soft-linking as they import and export all data being exchanged. This enables the use of the data organization as described in Section 3.3.

The SpineToolbox implementation can be found in Figure 4.5. The blue blocks with the file icon, or *Data Connection*-blocks, are direct references to data found on

Algorithm 4 Pseudo code for Emlabpy's algorithm that clears the capacity market.

for all Capacity Market m_{CM} **do**
  $d_{peak} = max(m_{CM} \rightarrow$ Node $\rightarrow$ Hourly demand)
  SDC = new SlopingDemandCurve(d_{peak})
  sorted_bids = sort m_{CM} bids on p_{bid} ascending

  $p_{CM} = 0$
  Var $T_{cap} = 0$
  for all Bid b in sorted_bids **do**
    if $b \rightarrow p_{bid} \leq$ SDC.get_price_for_volume($T_{cap} + b \rightarrow cap$) **then**
      $T_{cap} + = b \rightarrow cap$
      $p_{CM} = b \rightarrow p_{bid}$
      $b \rightarrow s =$ "Accepted"
    else if $b \rightarrow p_{bid} <$ SDC.get_price_for_volume(T_{cap}) **then**
      $p_{CM} = b \rightarrow p_{bid}$
      $b \rightarrow s =$ "Partially Accepted"
    else
      $b \rightarrow s =$ "Failed"
    end if
  end for
end for

the machine. Directly connected to the Data Connection-blocks are the dark purple *Importer*-blocks which contain the mapping of the input data to the SpineDB. Consequently, the Importer-blocks are directly connected to the *Data Store*-blocks which contain the reference to the SpineDB on the machine. Finally, the red blocks with the hammer are *Tool*-blocks which contain the execution of an external script, like Python or a Windows executable.

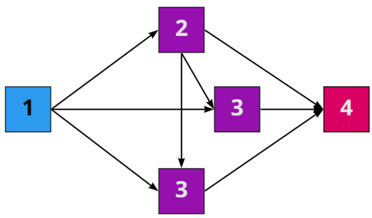


Figure 4.4: An abstraction of a portion of the SpineToolbox implementation to illustrate sequence indication. This illustrates the multiple uses of the arrows: in other cases, arrows indicate the transfer of data. The numbers in the boxes indicate the sequence of execution.

The layout is the result of an attempt to visualize the flow of information from left to right. Something unintuitive in SpineToolbox is the multiple definitions of arrows. Arrows indicate two things: flow of information and sequence. As an example, arrows are drawn between the Importer-blocks. However, no information passes through these but because of dependencies it is vital that one importer runs before the other. This is illustrated in Figure 4.4.

The central database approach from SpineToolbox enables flexibility regarding soft-linking. For example, currently, the workflow implemented is capable of allowing a common ontology in the future. The advantage is centralized ordering of data. The disadvantage, however, is that there is a large amount of overhead data which in this project is unnecessary.

The workflow in SpineToolbox is tied together by execution scripts and translation scripts. These translation scripts extract and transfer the critical data as intended in the conceptual model. This allows for the soft-linking methodology to be implemented as purely as possible. These translation scripts melt the data in the form desired by the other model.

4.3. Translation Scripts

Section 3.3 mentions the necessity of translation steps in the conceptual model. This section elaborates on the programmatical implementation of these steps. There are three scripts which have been developed for translation purposes: EM-Lab preprocessing, the EM-Lab to COMPETES script and finally the COMPETES to EM-Lab script.

4.3.1. EM-Lab Pre-processing

In Section 3.3, the pre-processing and the problem it solves have been discussed. The pre-processing handles data organization conflicts regarding power plant statuses. Table 4.2 shows an example of the two pre-processing cases. Plant 1 is ready to go: according to the years it is operational, and the status is 'OPR'. Plant 2 is, according to the '(D)' version, supposed to decommission in 2021. If the preprocessing is run in year 2021, the status changes to 'DECOM'. As an investment example, new plant 1 is an example of an investment decision. According to the year it's supposed to go operational in 2021 and decommission in 2061. When pre-processing is run for 2021, the status is set to 'OPR'.

4.3.2. EM-Lab to COMPETES

There are two main parts to the translation script of EM-Lab to COMPETES. One is the CO2 market clearing price, the other the capacity market revenues. This section describes the implementation of the coupling discussed in Section 3.2.

COMPETES accepts a CO2 price as input per month of every year. This means that the single yearly CO2 price, which has been output by EM-Lab, must be translated to multiple prices. This is done by repeating the price for every month of the year. The same is done for the future year (current year with the investment

Table 4.2: This table shows the effect of the pre-processing script if the current year is 2021. If the year has been reached of the '(D)' plant the actual plant has been set to decommissioned. Also, if the plant has been set to decommissioned but the '(D)' indicates this is not the case, the plant will be set to operational.

Name	Capacity	Year	Status before pre-processing		Status after pre-processing
Plant 1	25 MW	2012	OPR		OPR
Plant 1 (D)	-25 MW	2040	DECOM		DECOM
Plant 2	50 MW	1980	OPR	→	DECOM
Plant 2 (D)	-50 MW	2021	DECOM		DECOM
New Plant 1	100 MW	2021	DECOM	→	OPR
New Plant 1 (D)	-100 MW	2061	DECOM		DECOM

horizon), but for a CO2 price which has been extrapolated. A growth of 2.5% per year is used for this future price.

The capacity market revenues are implemented as fixed O&M and CAPEX reduction. There is a structural difference to these two properties in COMPETES: Fixed O&M costs are implemented per technology and CAPEX costs are mapped per technology per year. Because the CAPEX only affects the investment decisions in this coupling, the reduction is only applied to the investment year. The initial values are saved to check whether and how much, if the fixed O&M costs are 0, has been subtracted from the CAPEX costs.

4.3.3. COMPETES to COMPETES and EM-Lab

As opposed to the EM-Lab to COMPETES script, the COMPETES to EM-Lab script needs a COMPETES to COMPETES functionality. As Emlabpy has been developed for this coupling, it can commit its own decisions to its Spine database. Effectively, this means that through this script the investment and decommissioning decisions are also exported to the COMPETES Spine database.

Section 3.2 discusses the investment horizon and the build time for new investment decisions. This script extracts the technology build times and uses these for the investment decisions. For EM-Lab, the statuses are set to 'DECOM' as the pre-processing will set the correct statuses as described in Section 4.3.1. This prevents the investment decisions from being operational in this year's capacity market clearing. For COMPETES, the investment decisions are taken as output.

VRE and regular generators have different locations of data in COMPETES. While the generators have comparable values, different functions are required for this translation. Regarding EM-Lab, these generator differences have been mapped at initialization and for new investment decisions the same mapping is used. Finally, there is a structural difference between VRE and specifically PV, wind on-shore and wind off-shore. For the Netherlands, these investments have been fixed to the real-life set goals. This results in the script adding any investment to the already existing values in the database.

For decommissioning in EM-Lab and COMPETES, the year of the plant with '(D)' in its name will be set to the current year of investment. In EM-Lab, the iteration after the pre-processing will set the correct status of the decommissioned plant. This is described in Section 4.3.1. Using this pre-processing prevents the plant from being decommissioned before it participated in this iteration's capacity market clearing.

The dispatch results in an export of EM-Lab's dispatch plans. Per unit, one dispatch plan is created for this year with the sum of its generation and the average price for when it produces. Consequently, EM-Lab will find one dispatch plan and calculate the entire year's revenues.

Finally, the emissions are exported to EM-Lab. A single object containing the yearly emissions per plant is loaded in EM-Lab. If the plant is found in this mapping, this emission amount is realized. If not, EM-Lab continues with its traditional calculation.

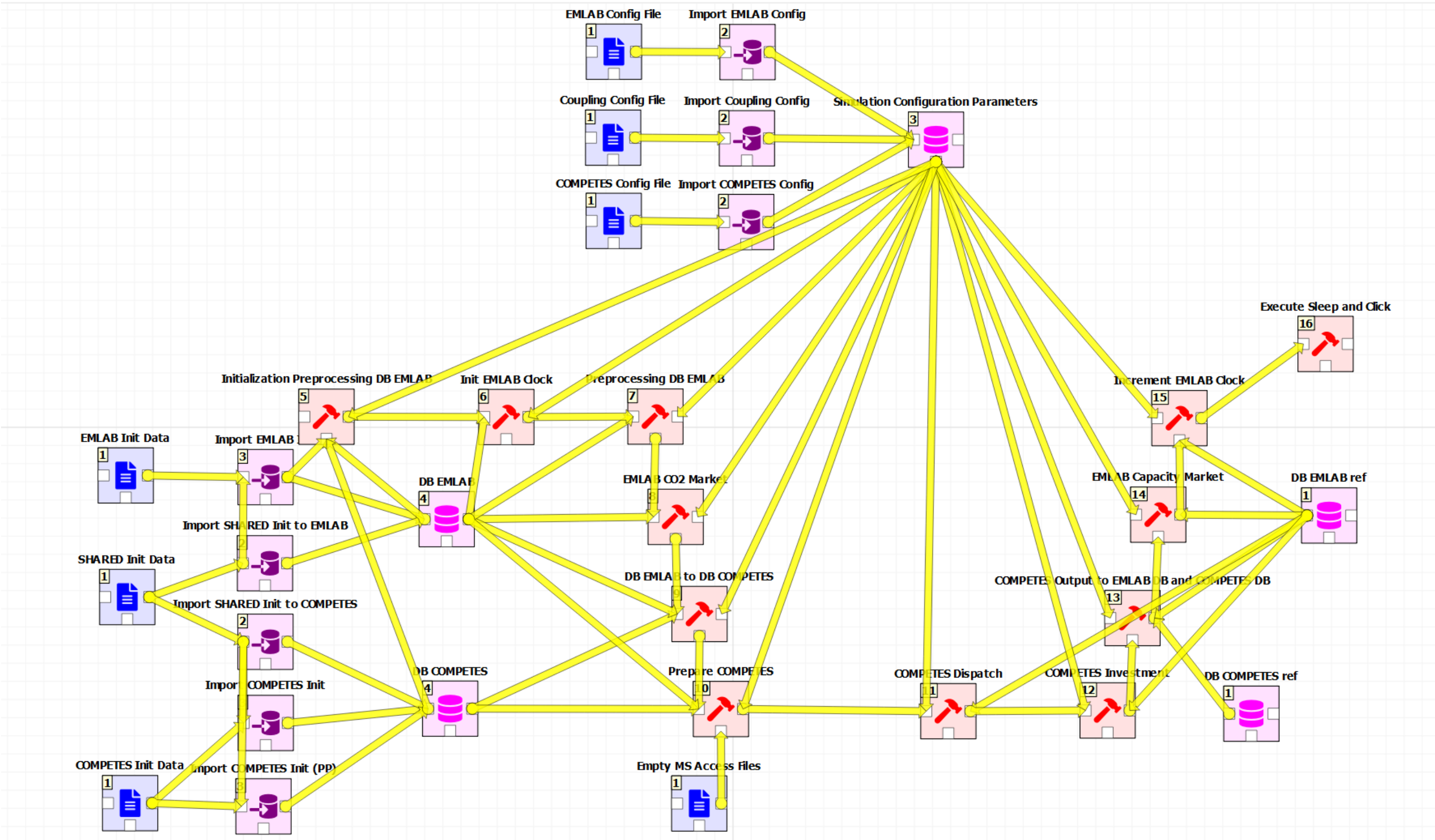


Figure 4.5: The SpineToolbox implementation. On the far left the references to the initialization data can be seen. These blocks are directly connected to "Importer" blocks where the mapping to the Spine database (SpineDB) is defined. The blocks DB EMLAB and DB COMPETES are the SpineDB blocks forming the backbone for each model respectively. What finally follows is the chain of model executions and translations between the executions. On top, the central simulation configuration parameters can be found. As all blocks require some interpretation of this data, there is a connection to almost every block.

5

Verification, Validation and Case Scenarios

After the development, the coupling has been verified and validated. This chapter provides a definition of the terms validation and verification. TNO has provided a data set for which COMPETES has been validated. The results of this soft-linking have been generated for the same data set and the results have been compared. The execution of the validation can be found in Section 6.1.

The definition of verification and validation follows the article by El Mir [60]. El Mir provides Figure 5.1, showing the evolution of the definition of verification and validation over the years. While this figure is developed in the context of Multi Vector Systems (MVS), the methodology can be applied to other subjects, such as soft-linking.

For the scope of this project, validation is the process of ensuring that the resulting model is in line with the research purpose. Verification is the process of ensuring the correctness of the resulting model. Since the soft-linking performed in this project can be seen as the creation of a new model, it is essential to continuously validate and verify. It is important to reiterate that the goal of this research is to explore the possibility and effects of soft-linking of the models in question. This means the validation and verification are focused on the effects of the coupling, not to be confused with the validation of results of a real-life scenario.

Section 1.4 briefly introduces the topic of validation and the taken approach. El Mir calls this approach of comparing the results to the results of a validated model the validation technique 'comparison to other models'. Means of doing so mentioned could be through graphs, confidence intervals, or hypothesis tests. This chapter further discusses the implementation of face validity and static testing. Fur-

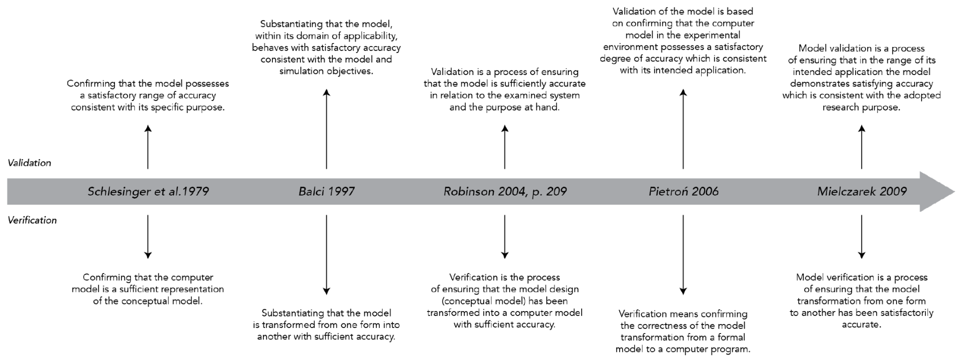


Figure 5.1: Figure from El Mir [60] showing the validation and verification definitions over the years [61][62][63][64][65].

thermore, the non-scarcity scenario provided and validated by TNO is elaborated on and the scarcity scenario is discussed.

5.1. Face Validity

The most basic validation method used in this project is face validity. Face validity is testing by eyeballing the results and reasoning through expertise and experience whether the results are sensible or reasonable.

During the development of this soft-linking, the implementations and intermediate results have been continuously discussed and checked with experts from multiple fields. Their opinions and intuitions of first interpretations have been weighed and implemented. In addition, this coupling was developed in close relation with TNO modelers who have specific expertise in this field.

5.2. Static Testing

In order to ensure verification of the Emlabpy model, static unit testing was implemented. In unit testing, an isolated piece of code is executed and the result is compared to an expected result [66]. This comparison results in a *True* or *False*, indicating the result of the test. This feature enables the possibility of creating many tests and executing packages of tests while the success or failure of tests remain insightful. Another advantage is that unit testing can be defined and executed separately from the code.

Unit testing results in what is known as *Line coverage* which is an indication what percentage of lines of code have been executed by unit tests. In this project, it was always desired to keep the line coverage as high as possible.

Static testing increases the confidence that the core operations of Emlabpy (e.g., reading the database, summing capacities of operational plants) work as intended. Additional testing is required for verifying correct model output or interpretation.

As stated in Section 4.1, Python has many open-source modules available. This is also the case for testing. For this project pytest is used. Pytest is capable of all necessary testing functions and can generate a coverage report. The focus in this project is to test the core functionalities of Emlabpy and to test the coupling scripts as these are most error-prone and crucial to other elements. Appendix D elaborates on the testing approach and results.

5.3. Comparison To Other Model

The coupling is validated by producing results with the coupling in two modes: a COMPETES-only mode, where the coupling features of Emlabpy (CO2 and capacity market) are disabled but where the soft-linking still introduces the COMPETES results back to itself, and a full coupling mode. The results are then compared, and the effects of the coupled models are studied. If the results can be explained as a direct effect of the coupling, the model is validated. This validation is built up from the developed modules, as they are elementary, to the coupled mechanisms.

For the validation the data set used is created by interpolating the data set provided by TNO for which COMPETES has been validated. The structure and organization of this data set have been introduced in Section 3.3 and a complete overview can be found in Appendix B. All data required for EM-Lab, as can be seen in Table 3.2, has been added.

This data set resembles the Dutch energy transition in multiple aspects:

- The installed capacity is modeled after existing Dutch generators.
- The Dutch renewable energy investment targets are implemented as fixed investments.
- The future technologies available for investment (H2, heat, storage) are modeled after probabilities in the Netherlands.

The data set provides Dutch and EU wide generators, technology and profiles for the years 2020 to 2032. The model will run from 2020 to 2025 and the investment horizon has been set at the longest technology build time, which in the data set is 7 years for nuclear plants (an approximation, in reality this number can be much higher). Because in the soft-linking COMPETES' investment module runs in the current year plus the investment horizon, data must be added until 2032.

The Dutch generators are expressed individually and in the other EU countries, the generators are aggregated by technology. This has no effect on validation, as units will only be compared aggregated by, e.g., technology.

The COMPETES-only run uses linearly interpolated data for the CO2 price. This data assumes the price of CO2 will be €103.59 per ton CO2 by 2050. The full coupling produces its own CO2 prices. Regarding the capacity market, the full coupling simulation has been run with an IRM of 10%, LM and UM of 3.5%, and a price cap of 75 kEuro per MW [47].

In order to study the soft-linking behavior under multiple circumstances, two scenarios are created: **Scarcity** and **Non-Scarcity**. The TNO data set contained moments where COMPETES hourly spot market prices would hit the value of lost load (VOLL). This is now known as the scarcity scenario. The peaks in demand across the EU causing this scarcity were gradually decreased until COMPETES was no longer producing such spot market prices. Both scenarios will be used for validation.

The load profiles in the data set are marginally the same every year. A more notable growth can be seen in the Dutch load profile. In terms of validation, this small difference over the years allows for more accurate studying of effects of investment and decommissioning as the energy mix will change but the load will stay roughly the same. The VOLL has been set to $10,000$ Euro / MWh.

The validation will be performed by looking at the market clearing prices for the CO2 and capacity market and their effect on the total emissions, spot market prices, investment and decommissioning decisions and energy balances. These elements will be compared to the COMPETES-only run, where there is no effect of the capacity market and a pre-set CO2 price.

The model validation conclusions are given in Section 6.1.

6
Results

In this chapter the results generated by the coupled model and the stand-alone COMPETES are discussed. Following Section 5.3, there are four types of results: full coupling non-scarcity, full coupling scarcity, COMPETES-only non-scarcity, and COMPETES-only scarcity. The results are discussed per main element in the coupling: dispatch, CO2 market, capacity market, and the investment and decommissioning.

For clarity, the COMPETES-only and full coupling results will be referred to as BASE and SOFT respectively. The names also represent the scenarios as described in Section 5.3, using NOM for nominal or non-scarcity and SCAR for scarcity. This results in the names BASE-NOM, BASE-SCAR, SOFT-NOM and SOFT-SCAR.

6.1. Validation of Results

The results produced directly by the model are discussed in Section 6.2, Section 6.3, Section 6.4 and Section 6.5. This section shows the validation of the soft-linking by focusing on the link between the BASE and the SOFT results. As discussed previously in Section 5.3, the validation is performed by evaluating the results in contrast to the already validated base scenario. This section zooms in on the results discussed in the other sections in Chapter 6 to determine the validity.

CO2 Market

The main function of the CO2 market is to produce a CO2 price based on the WTP of the operational plants. The coupling then transfers this price between the models. Section 6.3 shows the results regarding the CO2 market.

First, a clear correlation can be seen between the CO2 price and the level of emissions. For higher CO2 prices, there are lower emissions. The SCAR results reiterate this, as the correlation holds for more extreme numbers. Second, there seems to be interdependency between the CO2 price and spot market prices. The

CO2 market price is directly affected by spot market revenues. In addition, the spot market prices are affected by the CO2 price. This interdependency is further described in Section 6.3.

The prices are highly volatile. However, the elements determining the CO2 price, like the WTPs, the CO2 cap, and the emission amounts, seem to be in line with the resulting CO2 prices. The volatility invokes no reason to suspect the model is invalid. However, it does question model usability and whether the approach is appropriate and complex enough.

Capacity Market

The capacity market results have been presented and discussed in Section 6.4. As opposed to the CO2 market, the capacity market has a higher complexity regarding integration, as its clearing prices affect investment and decommissioning which are complex modules.

The price resulting from the capacity market is based on the planned losses of the plant and its capacity. Observing the output values, the values can be explained by them being the result of the method of coupling. The results show values which are 0 or extremely high which are out of the ordinary.

The values which are 0 can be explained by the nature of the capacity market. These values occur in the SCAR scenario where the spot market prices spike up to the VOLL. This causes all plants to have no losses and bid 0 in the capacity market. Therefore, these values are valid result of the coupling.

The extremely high values can be explained as an effect of the low CO2 prices in these years. This causes an increase in participating capacity driving up the clearing price. These points are further discussed in Section 6.4.

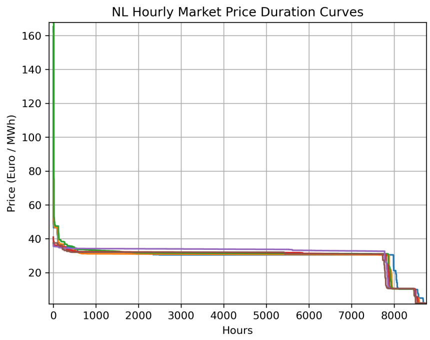
Differences in investment and decommissioning decisions can be observed in Figure 6.9 and Figure 6.8. Because of the many and thoroughly changed values, it is difficult to say whether these changes are a direct effect of the coupling. However, the results do seem to indicate an overall explicable change in decision-making as participating technologies seem to have more investment and less decommissioning. In addition, the results do not indicate an extremely negative or impossible change.

To conclude, regarding the capacity market, these observations indicate a valid soft-linking.

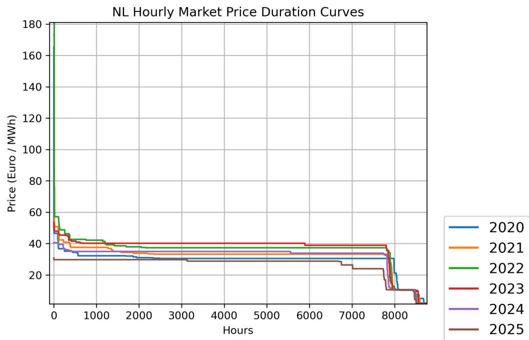
6.2. Dispatch

Figure 6.1 show the hourly spot market prices in the Netherlands for the simulated years. The BASE results, found in Figure 6.1a and Figure 6.1c, show a predictable progression through the years with the main difference being that the scarcity scenario has a higher average price and contains prices equal to the VOLL. The SOFT runs, found in Figure 6.1b and Figure 6.1d, show variable and increasing prices. Especially the SOFT-SCAR scenario shows an explosive increase in pricing.

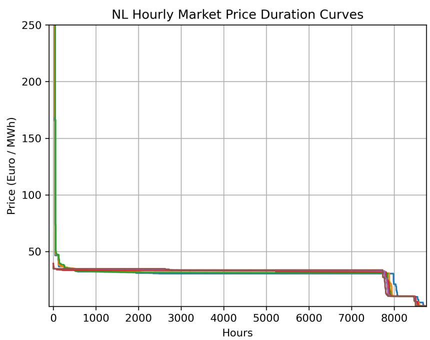
Important to note is the correlation of the rising spot market prices with the rise in CO2 prices discussed in Figure 6.4. The rising spot market prices can be explained by the effect the CO2 price has on the marginal costs of the plants.



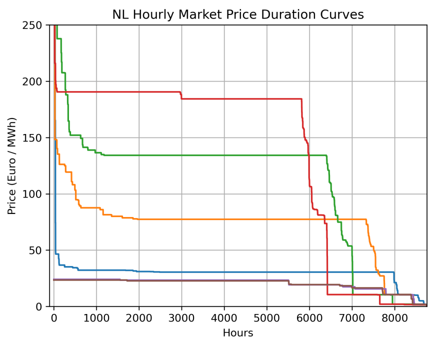
(a) BASE-NOM



(b) SOFT-NOM



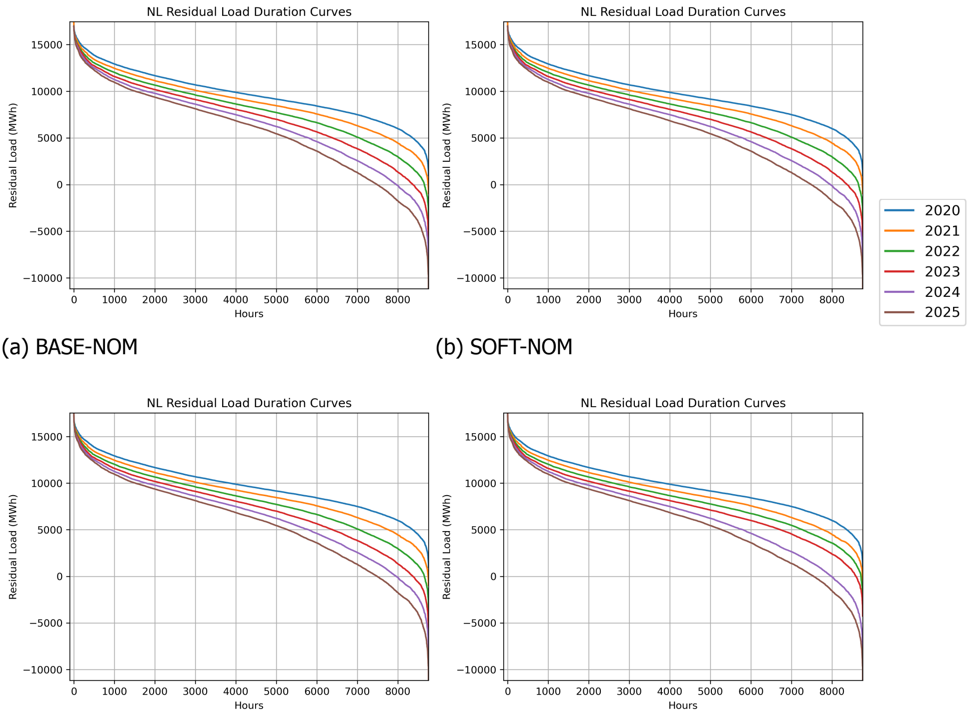
(c) BASE-SCAR



(d) SOFT-SCAR

Figure 6.1: The electricity spot market duration curves in the Netherlands. Note that the limit of the plots has been set to 250 Euro / MWh. In the scarcity scenarios there are prices which equal the VOLL, which is 10,000 Euro / MWh.

The graphs in Figure 6.2 show the residual load curves. The residual load is calculated by subtracting the supply from RES from the load duration curve. The load duration curves only differ marginally between the runs and are therefore only shown in Appendix C. The residual load graphs show the steady increase in renewable energy participation in the Netherlands. The graphs are similar as expected due to two reasons: the investment in renewable energy in the Netherlands is fixed (as discussed in Section 4.3.3) and the demand profiles are the same for the SCAR and NOM.



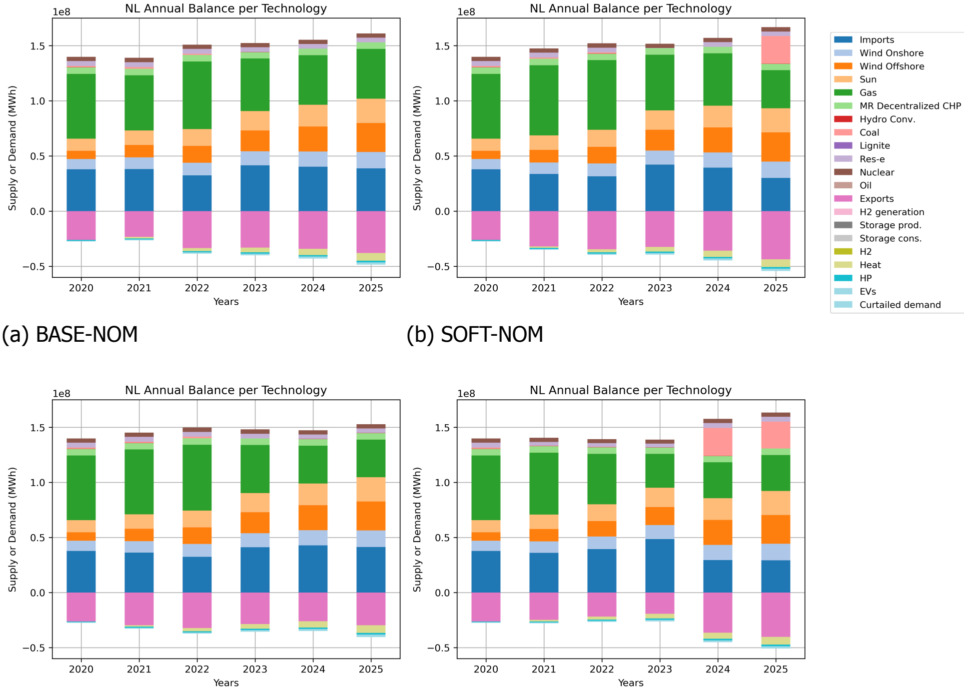
(a) BASE-NOM
(b) SOFT-NOM
(c) BASE-SCAR
(d) SOFT-SCAR

Figure 6.2: The residual load curves. These curves are calculated by subtracting the renewable energy generation from the load duration curves.

The annual balances found in Figure 6.3 show the unit commitment per technology in the Netherlands over the years. A clear correlation with the volatile CO2 prices in the coupling can be seen in Figure 6.3d and Figure 6.3b. The low CO2 price gives an opportunity for coal plants to start generating. In all figures, the steadily increasing share of renewable energies can be seen. The differences in total commitment can be explained by the fact that the low CO2 price allows generators to generate more energy to export and will import less energy.

6.3. CO2 Market

The produced CO2 market prices can be found in Figure 6.4. Important to note is that for BASE runs the CO2 price is interpolated based on an expectation in 2050. This graph, Figure 6.4a, also shows the price growing per month where the



(a) BASE-NOM (b) SOFT-NOM

(c) BASE-SCAR (d) SOFT-SCAR

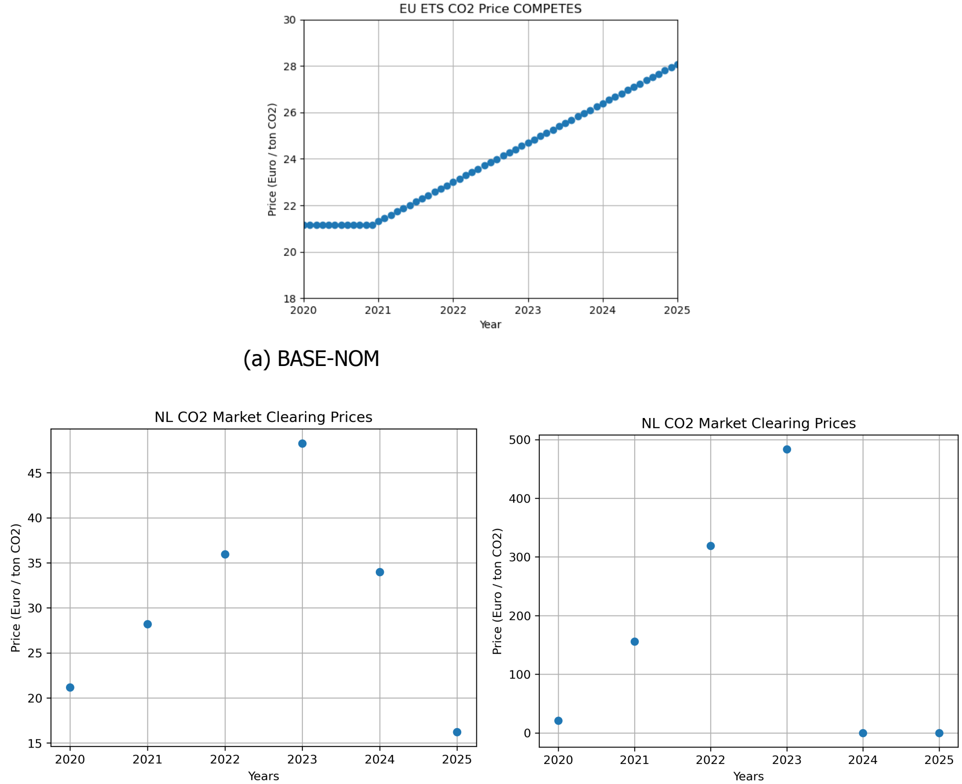
Figure 6.3: These graphs show the Dutch annual total generation and demand per technology.

prices in the SOFT scenarios (Figure 6.4b and Figure 6.4c) show the prices per year.

The prices in the SOFT scenarios, Figure 6.4b and Figure 6.4c, show high volatility. The observed trend can be explained by the following points:

- **Interdependency Spot Market Price**
  As discussed in Section 4.1.1, the CO2 market price is cleared based on the WTP of the operational plants. The WTP rises if the average spot market price rises, which is the case as observed in Figure 6.1. This means that, after the first clearing in 2021 where the price is higher than average, the CO2 market price will keep rising.
- **Scarcity**
  Specifically in Figure 6.4c, the model deals with scarcity. The spot market clearing prices that reach VOLL directly cause an explosive increase in the spot market as per the previous point.
- **Investments**
  The decrease of the CO2 price after 2023 in both graphs can be explained by

the fact that the first investments in this modeling are operational in 2023, which can be observed in Figure 6.9. In Figure 6.3, it can be observed that there is a rise in 'Imports' in 2023, as the investments elsewhere in the EU make it cheaper to import. This has a direct effect lowering the spot market prices, observed in Figure 6.1, which as per the previous points affects the WTP of the plants and with that the CO2 price.

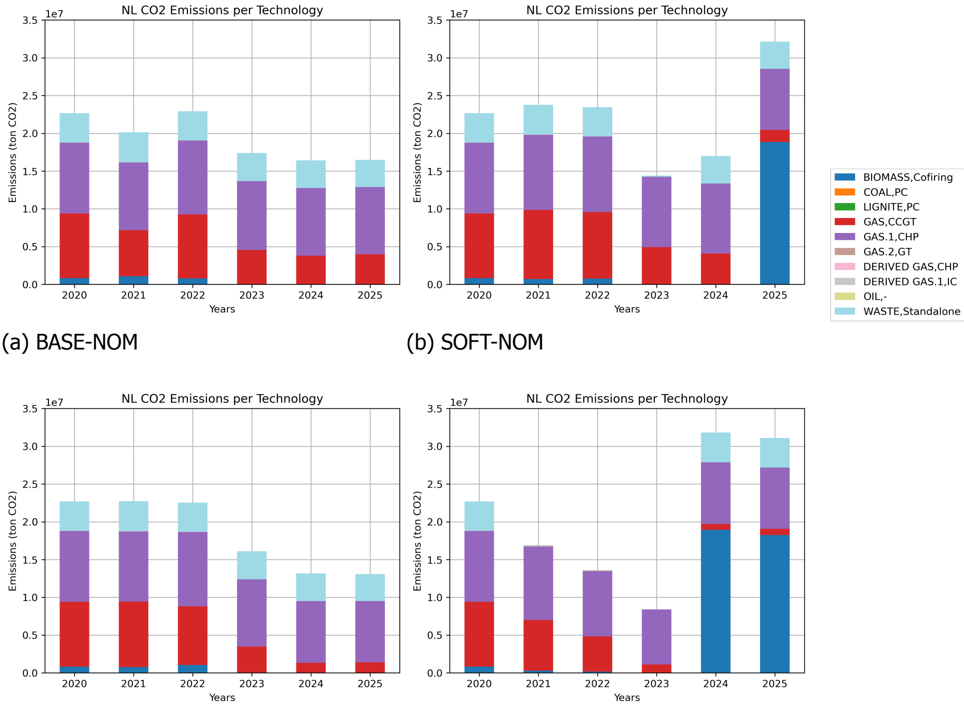


(a) BASE-NOM

(b) SOFT-NOM (c) SOFT-SCAR

Figure 6.4: These graphs show Emlabpy's CO2 market clearing prices. For COMPETES, in the base and scarcity scenarios, the same linearly interpolated data is used.

The CO2 price can be seen to have a direct effect on the amount of emissions in the Netherlands, as shown in Figure 6.5. It is expected that an increase in the CO2 price will make it less attractive for plants to emit CO2.



(a) BASE-NOM

(b) SOFT-NOM

(c) BASE-SCAR

(d) SOFT-SCAR

Figure 6.5: The Dutch annual CO2 emissions per technology. Important to note that 'Biomass, Cofiring' can be described as 'Coal' in other graphs.

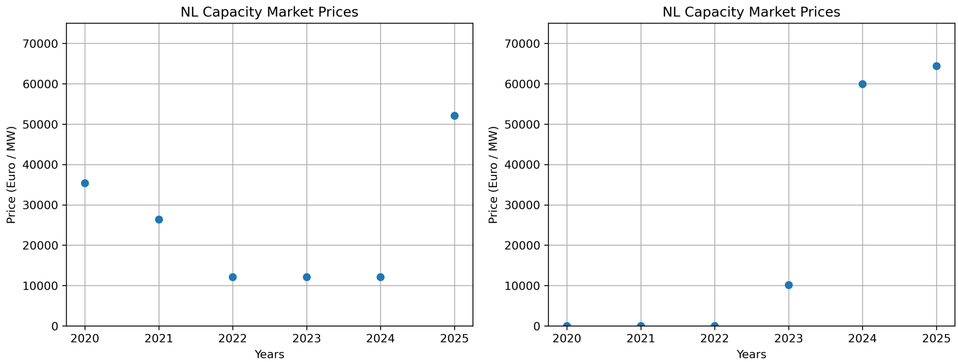
6.4. Capacity Market

At the end of every iteration in the coupling, the capacity market outputs results. These results can be found in Figure 6.6. Noteworthy is that in the SCAR scenarios, the capacity market returns 0 Euro / MW. This is an expected result. The capacity market is a remuneration system to compensate losses by means of capacity. In the years 2020 until 2023, the plants are making profits through the spot market that delivers VOLL. As a result, the plants do not need to be compensated anymore.

The increase in capacity market price in 2024 and 2025 can be explained by the increase in total generation, caused by the low CO2 price. The increase in participating capacity drives up the capacity in the merit order as explained in Section 4.1.2. A different explanation for a rise in capacity market price could be a drop in spot market prices or increase in generator costs, increasing predicted generator losses. This seems less likely as the other graphs suggest lower costs and higher returns through the increase of generation.

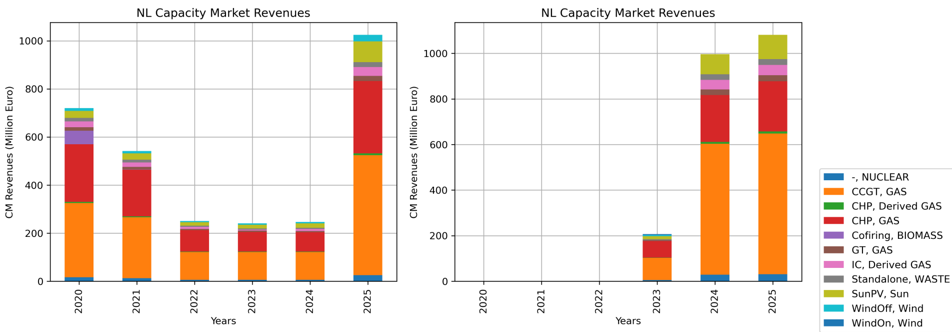
Figure 6.7 shows the annual capacity market revenues by technology. This specifies which technologies are participating in the capacity market and will thus receive the reductions as discussed in Chapter 3.

In Section 6.5, the possible effect of the capacity market on the investment and decommissioning is discussed.



(a) SOFT-NOM (b) SOFT-SCAR

Figure 6.6: Emlabpy's capacity market clearing prices for the base and scarcity scenarios.



(a) SOFT-NOM (b) SOFT-SCAR

Figure 6.7: The yearly capacity market revenues per technology. From the graph it is clear which technologies are participating in the capacity market.

6.5. Investment and Decommissioning

This section discusses the investment and decommissioning decisions in the Netherlands, found in Figure 6.8, and in the EU, found in Figure 6.9. The graphs show the year the new capacity is operational. In contrast to the previous results, the results regarding investment are not as straightforward as the investment module is complex and depends on many factors.

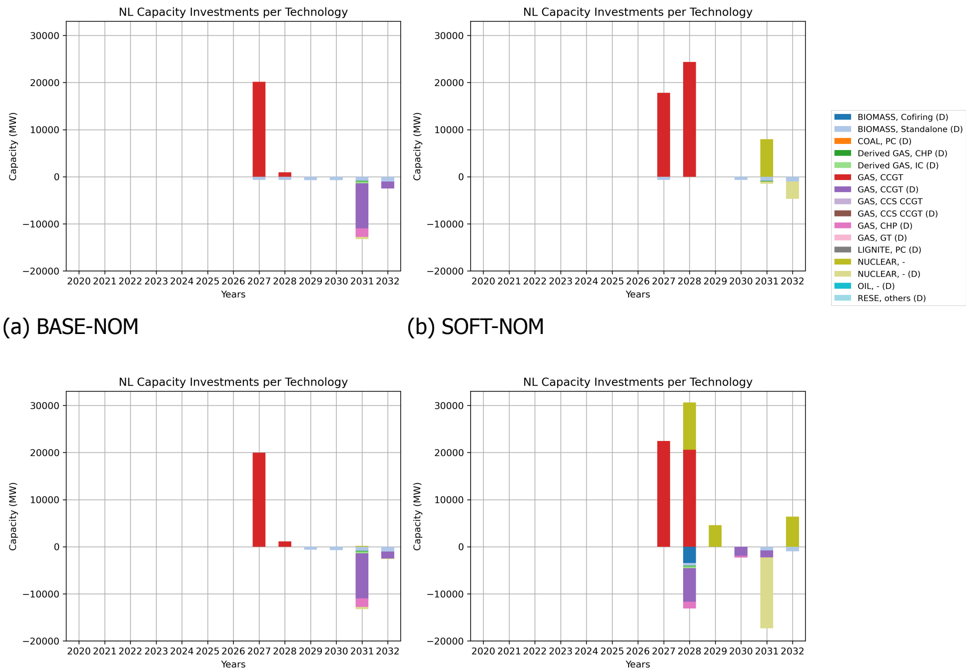
As discussed in Section 3.2.4, investment and decommissioning decisions are conceptualized through investment delays which are defined through build time. A direct effect in the results is that there are no investments in 2020, 2021 and 2022, as there are no technologies with such small build times. This investment delay can however bring inconsistency to the simulations, as investments could be operational before their optimal time. For example, when the investments are run for year 2027, there is CCGT implemented in 2023, 4 years before the optimal point. The difference with the timing of decommissioning, which would happen in the year the investment module is run, could also bring inconsistency to the simulations.

An interesting observation is that the BASE scenarios, Figure 6.8a, Figure 6.8c, Figure 6.9a and Figure 6.9c, are similar in total investments. EU-wide, some of the CCGT investment in the SCAR scenario in 2027 has shifted to 2028 and there is an increase in nuclear investment which has also shifted from 2032 to 2031. It is possible that the years have changed regarding optimal investment timing. In the Netherlands, the decommissioning of biomass is delayed. This most likely indicates that in 2020, when the model is experiencing scarcity, it decides to delay the decommissioning of biomass in the year 2020 with an investment horizon of 7 years.

When comparing the BASE results to the SOFT results, there are differences in investment as well as decommissioning decisions. Focusing on the Netherlands, Figure 6.8b shows a large investment in CCGT in 2028 as opposed to the BASE runs in Figure 6.8a. CCGT was participating in the capacity market and had thus received the reduction in fixed O&M costs and CAPEX costs. A possible explanation is that this reduction incentivized more investment in CCGT. This would also explain why there is less CCGT decommissioning in 2031 in the SOFT scenarios.

In 2031, the BASE run shows decommissioning (largely CCGT) and the SOFT run shows investment (largely nuclear). Nuclear also participates in the capacity market and receives reductions, likely contributing to this investment decision. However, in the SOFT results, in 2024 there is a high CO2 price. This is also the year the investment decision was made to invest in nuclear in 2031. The high CO2 price seems a possible explanation in the investment in nuclear, which provides a high amount of ready capacity for a low emission output. However, something that is unclear is the decommissioning of nuclear in 2031 and 2032 in the SOFT scenarios. An explanation could be that some nuclear generators are set to decommission due to e.g. old age, as the decommissioning is also shown in the BASE scenarios. It also could be that, due to the sudden interest in nuclear, it is optimal to start upgrading (decommissioning and investing) nuclear generators. Lastly, these graphs represent all countries in Europe and it could be the case that investment in one country and decommissioning in another country is optimal. It is reasonable to assume that the energy mix is approaching an optimum faster through these reductions.

Comparing the EU-wide scenarios, it can be observed that in total fewer decommissioning takes place. This is especially notable when comparing the NOM scenario graphs, Figure 6.9a and Figure 6.9b, where there is more nuclear decommissioning in SOFT-NOM but almost no other technologies. This difference could be attributed to the implementation of the capacity market, as generators are still profitable using capacity market reductions. However, it is also likely that the higher electricity spot market prices increased profitability across all generating plants.



(a) BASE-NOM

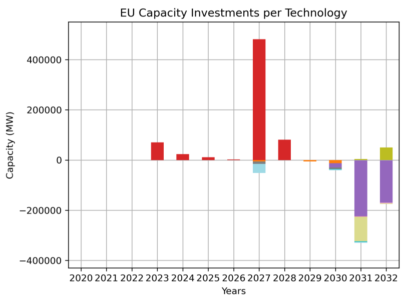
(b) SOFT-NOM

(c) BASE-SCAR

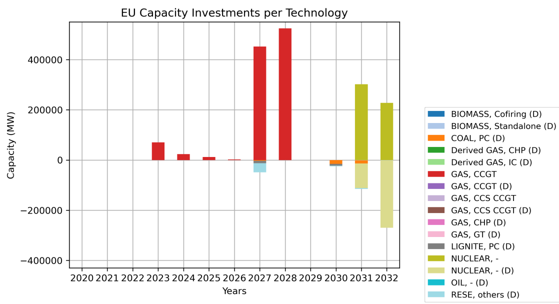
(d) SOFT-SCAR

Figure 6.8: Investments and decommissioning in capacity in the Netherlands per technology. Decommissioning is indicated by '(D)' in the legend and it is portrayed as negative capacity.

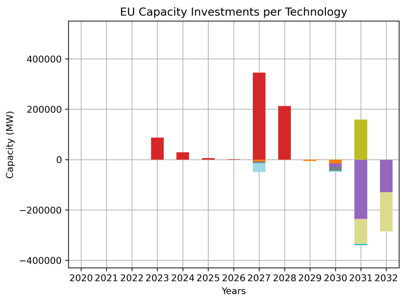
6



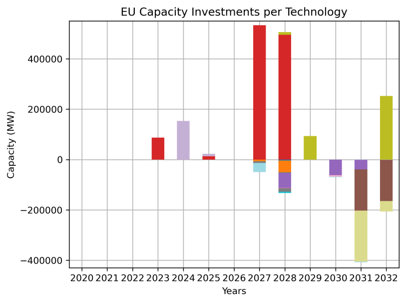
(a) BASE-NOM



(b) SOFT-NOM



(c) BASE-SCAR



(d) SOFT-SCAR

Figure 6.9: Investments and decommissioning in capacity in the EU per technology. Decommissioning is indicated by '(D)' in the legend and it is portrayed as negative capacity.

7 Conclusion

This dissertation described the soft-linking of modules from an agent-based model to an optimization model. The dissertation is based around answering two research questions defined in Section 1.3. The first question regarding the improvement of investment decision-making through soft-linking has been split into multiple sub-questions, which have been answered throughout the report.

First, to answer the first research sub-question, the requirements of this soft-linking are defined in Section 2.4.1. The necessary elements for soft-linking are information exchange and information mapping between two elements. In addition, there should be a timing definition of when and how the models interact.

These requirements are reflected in the design choices made in Chapter 3, structurally answering the first research question and its other sub-questions. The soft-linking data exchange and mapping have been elaborated on in Section 3.1 and Section 3.2. Figure 3.2 shows a detailed description of information flow, information transformation and timing of the individual model runs in the soft-linking. It is discussed how the individual model time granularities are respected and utilized.

Finally, Chapter 4 discusses the implementation of the conceptual definitions created in Chapter 3. The soft-linking focused on the connection of four elements: the CO2 market, the capacity market, dispatch and the investment and decommissioning decisions. These elements are reflected in the chapters Chapter 4 and Chapter 6.

Before the results were discussed, the model was validated realizing the second research question. Section 1.4 provides some background regarding this validation. Chapter 5 then goes in-depth on the topic of verification and validation. Section 5.3 provides context regarding the Dutch scenario. The static testing results can be found in Appendix D.

The results show the effect of the coupling. These effects beg the question whether this approach is suitable for studying future power systems. For example, it can be concluded that the CO_2 market requires future work as the prices are highly volatile and reach extreme numbers. The capacity market has shown an overall adequate influence on investment and decommissioning decisions.

While the results show high volatility and unrealistic numbers, there is no reason to assume the soft-linking has failed. In the respective chapters, the results have been explained being a product of the market implementation. Section 7.1 describes how the implementation can be altered or further developed to achieve more realistic results.

During this research, the software kit SpineToolbox was explored. SpineToolbox was still under heavy development during this project. Nonetheless, it proved to be a useful tool for implementing the conceptual soft-linking strategy. Especially in such a soft-linking with separate modules, SpineToolbox provides transparency and oversight in an otherwise 'closed' project. In addition, its database centered approach provides potential for future coupling of other energy models.

7.1. Future work

Because of the limited scope and time frame of this project there is definite potential for further research. The current research was exploratory and with the purpose of finding out whether this soft-linking approach is feasible.

The current results of the soft-linking are all results where both the CO_2 market and capacity market were linked simultaneously. In hindsight, it would have been interesting to see the effect of the individual modules by creating more soft-linking scenarios. E.g., the SOFT scenarios would be split to SOFTFULL, SOFTCM and SOFTCO2 to indicate the full soft-linking, soft-linking with the capacity market and soft-linking with the CO_2 market respectively. The effects of each mechanism could then be more effectively studied, as the effects that overlap would be more clear.

It can be concluded that the CO_2 price is too volatile to be accurately used in future energy system analysis. These prices have been explained through their correlation with the spot market prices and generator profits. This volatility is enabled by a lack of market foresight, which would be considered in the real world. A subject for future work would be the implementation of a future CO_2 market that influences the current CO_2 price. COMPETES realizes future generator emissions as part of its investment module which could be beneficial towards this goal. In addition, the introduction of a CO_2 price cap and floor would also work towards stabilizing the CO_2 price.

Furthermore, future work preventing CO2 price volatility would be the complete implementation of an MSR, described in Section 2.2. An MSR would store and release CO2 credits according to shortage or surplus of allowances in circulation. This would increase price stability if the banking of CO2 credits is implemented. EM-Lab provides such an implementation of CO2 credit banking and an MSR, however COMPETES does not. As this was not taken into consideration when developing the conceptual model, the MSR was not included in the scope of this project.

In addition, because EM-Lab is a Dutch oriented model, the current CO2 price is cleared using Dutch plants and an estimated CO2 cap. It would be more accurate to clear the CO2 market for the entire EU, using the actual EU CO2 cap for the electricity sector. This has shown to prevent price volatility and increase accuracy.

Similarly, although the capacity market only considers Dutch plants, the integration of the capacity market revenues through the fixed O&M costs and CAPEX costs affect all generators in Europe. In reality, this would not be the case and should be changed in future work.

In the current conceptual model, many of the features and strengths of EM-Lab, like the fact it is agent-based, were not profited from. Future research could focus on the enabling of the agent-based implementation of EM-Lab and draw some of the investment modules to EM-Lab. This would require more intense soft-linking and more transparency from COMPETES. To implement the agent-based approach, it would be required to have a detailed dispatch model that can be run quickly to be used by EM-Lab's investment forecasting each iteration.

The current implementation makes use of combined initialization data of EM-Lab and COMPETES. For clarity and flexibility towards coupling of other models, the implementation of a single ontology would be highly advantageous. A single ontology would simplify the translation scripts and improve transparency in the coupling.

As the SpineToolbox implementation is built for future expansion, all data required by EM-Lab and COMPETES individually is taken into consideration in this coupling. This creates a large overhead of unnecessary data, because this data is only required by the individual models and not in the soft-linking. Future work could focus on a critical look at which data is loaded in the soft-linking. A single ontology would help define what data is necessary in soft-linking of other models.

In the definition of the conceptual model, it was mentioned that parallel execution of the COMPETES runs is not possible. AIMMS, the programming language and execution environment of COMPETES, does not support this. A significant speed-up could be acquired if the COMPETES parts are executed in parallel. It is possible that in the future AIMMS will support this kind of parallel execution.

Bibliography

[1] e. a. M. Banja N. Scarlat, *Renewable Energy Progress In EU 27 2005-2020*. Luxembourg: Publications Office of the European Union, 2013. doi: 10.2790/13181.

[2] F. Gökgöz and M. T. Güvercin, "Energy security and renewable energy efficiency in eu", *Renewable and Sustainable Energy Reviews*, vol. 96, pp. 226–239, 2018, issn: 1364-0321. doi: https://doi.org/10.1016/j.rser.2018.07.046. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1364032118305549.

[3] R. Lacal Arantegui and A. Jäger-Waldau, "Photovoltaics and wind status in the european union after the paris agreement", *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 2460–2471, 2018, issn: 1364-0321. doi: https://doi.org/10.1016/j.rser.2017.06.052. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S136403211731002X.

[4] S.-G. European Commission, *Communication from the commission to the european parliament, the european council, the council, the european economic and social committee and the committee of the regions the european green deal*, 2019. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=COM:2019:640:FIN.

[5] e. a. Chen Mcelroy, "Transition towards higher penetration of renewables: An overview of interlinked technical, environmental and socio-economic challenges", *J. Mod. Power Syst. Clean Energy*, vol. 7, pp. 1–8, 2019. doi: https://doi-org.tudelft.idm.oclc.org/10.1007/s40565-018-0438-9.

[6] J. Ma, V. Silva, R. Belhomme, D. S. Kirschen, and L. F. Ochoa, "Evaluating and planning flexibility in sustainable power systems", *IEEE Transactions on Sustainable Energy*, vol. 4, no. 1, pp. 200–209, 2013. doi: 10.1109/TSTE.2012.2212471.

[7] G. Strbac, D. Papadaskalopoulos, N. Chrysanthopoulos, *et al.*, "Decarbonization of electricity systems in europe: Market design challenges", *IEEE Power and Energy Magazine*, vol. 19, no. 1, pp. 53–63, 2021. doi: 10.1109/MPE.2020.3033397.

[8] A. Estanqueiro, *New market model for 100% renewable power systems*. [Online]. Available: https://traderes.eu/.

[9] H. Allcott and S. Mullainathan, "Behavior and energy policy", *Science*, vol. 327, no. 5970, pp. 1204–1205, 2010.

[10] F. Gracceva and P. Zeniewski, "A systemic approach to assessing energy security in a low-carbon eu energy system", *Applied Energy*, vol. 123, pp. 335–348, 2014.

[11] TNO, *The supply of flexibility for the power system in the netherlands, 2015-2050*. [Online]. Available: https://www.tno.nl/media/12356/e17044-flexnet-the-supply-of-flexibility-for-the-power-system-in-the-netherlands-2015-2050-phase-2.pdf.

[12] TU Delft, *Emlab - energy modelling laboratory*. [Online]. Available: http://emlab.tudelft.nl/.

[13] C. Gomes, C. Thule, D. Broman, P. G. Larsen, and H. Vangheluwe, "Co-simulation: State of the art", *arXiv preprint arXiv:1702.00686*, 2017.

[14] M. Pavičević, A. Mangipinto, W. Nijs, *et al.*, "The potential of sector coupling in future european energy systems: Soft linking between the dispa-set and jrc-eu-times models", *Applied Energy*, vol. 267, p. 115 100, 2020, issn: 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2020.115100. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261920306127.

[15] J. Deane, A. Chiodi, M. Gargiulo, and B. P. Ó Gallachóir, "Soft-linking of a power systems model to an energy systems model", *Energy*, vol. 42, no. 1, pp. 303–312, 2012, 8th World Energy System Conference, WESC 2010, issn: 0360-5442. doi: https://doi.org/10.1016/j.energy.2012.03.052. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544212002551.

[16] P. CRAMTON, A. OCKENFELS, and S. STOFT, "Capacity market fundamentals", *Economics of Energy & Environmental Policy*, vol. 2, no. 2, pp. 27–46, 2013, issn: 21605882, 21605890. [Online]. Available: http://www.jstor.org/stable/26189455.

[17] A. B. Jaffe and F. A. Felder, "Should electricity markets have a capacity requirement? if so, how should it be priced?", *The Electricity Journal*, vol. 9, no. 10, pp. 52–60, 1996.

[18] "Acer market monitoring report 2019 - electricity wholesale markets volume", Agency for the Cooperation of Energy Regulators (ACER), Tech. Rep., 2019.

[19] J. E. Duggan, "Capacity market mechanism analyses: A literature review", *Current Sustainable/Renewable Energy Reports*, pp. 1–7, 2020.

[20] M. Bidwell, "Reliability options: A market-oriented approach to long-term adequacy", *The Electricity Journal*, vol. 18, no. 5, pp. 11–25, 2005, issn: 1040-6190. doi: https://doi.org/10.1016/j.tej.2005.03.010. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1040619005000540.

[21] A. N. Kleit and R. J. Michaels, "Reforming texas electricity markets", *Regulation*, vol. 36, no. 2, p. 32, 2013.

[22] M. Petitet, D. Finon, and T. Janssen, "Capacity adequacy in power markets facing energy transition: A comparison of scarcity pricing and capacity mechanism", *Energy Policy*, vol. 103, pp. 30–46, 2017.

[23] L. de Vries and P. Heijnen, "The impact of electricity market design upon investment under uncertainty: The effectiveness of capacity mechanisms", *Utilities Policy*, vol. 16, no. 3, pp. 215–227, 2008, Capacity Mechanisms in Imperfect Electricity Markets, issn: 0957-1787. doi: https://doi.org/10.1016/j.jup.2007.12.002. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957178708000027.

[24] P. Bhagwat, "Security of supply during the energy transition: The role of capacity mechanisms", Ph.D. dissertation, Delft University of Technology, 2016.

[25] P. Cramton and S. Stoft, "A capacity market that makes sense", *The Electricity Journal*, vol. 18, no. 7, pp. 43–54, 2005, issn: 1040-6190. doi: https://doi.org/10.1016/j.tej.2005.07.003. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1040619005000850.

[26] P. Benalcazar and P. Nalepka, "The polish capacity market proposal vs the british model", *Polityka Energetyczna-Energy Policy Journal*, 2017.

[27] F. J. Convery, "Origins and development of the eu ets", *Environmental and Resource Economics*, vol. 43, no. 3, pp. 391–412, 2009.

[28] Y.-J. Zhang and Y.-M. Wei, "An overview of current research on eu ets: Evidence from its operating mechanism and economic effect", *Applied Energy*, vol. 87, no. 6, pp. 1804–1814, 2010, issn: 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2009.12.019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S030626190900556X.

[29] G. Perino and M. Willner, "Procrastinating reform: The impact of the market stability reserve on the eu ets", *Journal of Environmental Economics and Management*, vol. 80, pp. 37–52, 2016.

[30] J. M. Reilly and S. Paltsev, "An analysis of the european emission trading scheme", MIT Joint Program on the Science and Policy of Global Change, Tech. Rep., 2005.

[31] S. Osorio, O. Tietjen, M. Pahle, R. Pietzcker, and O. Edenhofer, "Reviewing the market stability reserve in light of more ambitious eu ets emission targets", 2020.

[32] C. Chaton, A. Creti, and M.-E. Sanin, "Assessing the implementation of the market stability reserve", *Energy policy*, vol. 118, pp. 642–654, 2018.

[33] G. Perino and M. Willner, "Eu-ets phase iv: Allowance prices, design choices and the market stability reserve", *Climate Policy*, vol. 17, no. 7, pp. 936–946, 2017.

[34] R. Gerlagh, R. J. Heijmans, and K. E. Rosendahl, "Covid-19 tests the market stability reserve", *Environmental and Resource Economics*, vol. 76, no. 4, pp. 855–865, 2020.

7

[35] H.-M. Groscurth, T. Bruckner, and R. Kümmel, "Modeling of energy-services supply systems", *Energy*, vol. 20, no. 9, pp. 941–958, 1995, issn: 0360-5442. doi: https://doi.org/10.1016/0360-5442(95)00067-Q. [Online]. Available: https://www.sciencedirect.com/science/article/pii/036054429500067Q.

[36] K. C. Hoffman and D. O. Wood, "Energy system modeling and forecasting", *Annual review of energy*, vol. 1, no. 1, pp. 423–453, 1976.

[37] S. Jebaraj and S. Iniyan, "A review of energy models", *Renewable and Sustainable Energy Reviews*, vol. 10, no. 4, pp. 281–311, 2006, issn: 1364-0321. doi: https://doi.org/10.1016/j.rser.2004.09.004. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032104001261.

[38] C. M. Macal and M. J. North, "Tutorial on agent-based modeling and simulation", in *Proceedings of the Winter Simulation Conference, 2005.*, IEEE, 2005, 14–pp.

[39] P. A. Abrams, "Implications of dynamically variable traits for identifying, classifying, and measuring direct and indirect effects in ecological communities", *The American Naturalist*, vol. 146, no. 1, pp. 112–134, 1995.

[40] P. Hansen, X. Liu, and G. M. Morrison, "Agent-based modelling and socio-technical energy transitions: A systematic literature review", *Energy Research & Social Science*, vol. 49, pp. 41–52, 2019.

[41] Z. Zhou, W. K. V. Chan, and J. H. Chow, "Agent-based simulation of electricity markets: A survey of tools", *Artificial Intelligence Review*, vol. 28, no. 4, pp. 305–342, 2007.

[42] S. A. Harp, S. Brignone, B. F. Wollenberg, and T. Samad, "Sepia. a simulator for electric power industry agents", *IEEE Control Systems Magazine*, vol. 20, no. 4, pp. 53–69, 2000.

[43] C. Macal, P. Thimmapuram, V. Koritarov, *et al.*, "Agent-based modeling of electric power markets", in *Proceedings of the Winter Simulation Conference 2014*, IEEE, 2014, pp. 276–287.

[44] R. Entriken, "Using automated agents with probe: Interface requirements specification", Technical Report 2005. 1012157. EPRI, Palo Alto, Tech. Rep., 2005.

[45] E. J. Chappin, L. J. de Vries, J. C. Richstein, P. Bhagwat, K. Iychettira, and S. Khan, "Simulating climate and energy policy with agent-based modelling: The energy modelling laboratory (emlab)", *Environmental Modelling & Software*, vol. 96, pp. 421–431, 2017, issn: 1364-8152. doi: https://doi.org/10.1016/j.envsoft.2017.07.009. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364815216310301.

[46] L. de Vries, E. Chappin, and J. Richstein, *EMLab-Generation : An experimentation environment for electricity policy analysis*, English. Netherlands: Delft University of Technology, 2013, Version 1.0.

[47] P. C. Bhagwat, A. Marcheselli, J. C. Richstein, E. J. Chappin, and L. J. De Vries, "An analysis of a forward capacity market with long-term contracts", *Energy Policy*, vol. 111, pp. 255–267, 2017, issn: 0301-4215. doi: https://doi.org/10.1016/j.enpol.2017.09.037. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0301421517305967.

[48] J. C. Richstein, "Interactions between carbon and power markets in transition", Ph.D. dissertation, Gildeprint Drukkerijen, 2015.

[49] B. Gurfel, "Model of the economic efficiency of the exploitation of oil shale in comparison with other mineral sources of energy", *Applied Energy*, vol. 5, no. 3, pp. 205–213, 1979, issn: 0306-2619. doi: https://doi.org/10.1016/0306-2619(79)90037-0. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0306261979900370.

[50] A. Arnette and C. W. Zobel, "An optimization model for regional renewable energy development", *Renewable and Sustainable Energy Reviews*, vol. 16, no. 7, pp. 4606–4615, 2012, issn: 1364-0321. doi: https://doi.org/10.1016/j.rser.2012.04.014. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032112002729.

[51] A. Rentizelas, I. Tatsiopoulos, and A. Tolis, "An optimization model for multi-biomass tri-generation energy supply", *Biomass and Bioenergy*, vol. 33, no. 2, pp. 223–233, 2009, issn: 0961-9534. doi: https://doi.org/10.1016/j.biombioe.2008.05.008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0961953408001463.

[52] V. Taseska-Gjorgievska, M. Todorovski, N. Markovska, and A. Dedinec, "An integrated approach for analysis of higher penetration of variable renewable energy: Coupling of the long-term energy planning tools and power transmission network models", *Journal of Sustainable Development of Energy, Water and Environment Systems*, vol. 7, no. 4, pp. 615–630, 2019.

[53] S. Collins, J. P. Deane, K. Poncelet, *et al.*, "Integrating short term variations of the power system into integrated energy system models: A methodological review", *Renewable and Sustainable Energy Reviews*, vol. 76, pp. 839–856, 2017, issn: 1364-0321. doi: https://doi.org/10.1016/j.rser.2017.03.090. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1364032117304264.

[54] D. F. Dominković, R. G. Junker, K. B. Lindberg, and H. Madsen, "Implementing flexibility into energy planning models: Soft-linking of a high-level energy planning model and a short-term operational model", *Applied Energy*, vol. 260, p. 114 292, 2020, issn: 0306-2619. doi: https://doi.org/10.1016/j.apenergy.2019.114292. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261919319798.

[55] C. A. Deane J.P. Gracceva F., "Soft-linking exercises between times, power system models and housing stock models", in *Informing Energy and Climate Policies Using Energy Systems Models*, Springer, Cham, 2015, pp. 315–331.

[56] A. Krook-Riekkola, C. Berg, E. O. Ahlgren, and P. Söderholm, "Challenges in top-down and bottom-up soft-linking: Lessons from linking a swedish energy system model with a cge model", *Energy*, vol. 141, pp. 803–817, 2017, issn: 0360-5442. doi: https://doi.org/10.1016/j.energy.2017.09.107. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544217316274.

[57] VTT Technical Research Centre of Finland Ltd, *Project spine*. [Online]. Available: http://www.spine-model.org/spine%5C_toolbox.htm.

[58] VTT, *Vtt*. [Online]. Available: https://www.vttresearch.com/en.

[59] C. Byers, T. Levin, and A. Botterud, "Capacity market design and renewable energy: Performance incentives, qualifying capacity, and demand curves", *The Electricity Journal*, vol. 31, no. 1, pp. 65–74, 2018, issn: 1040-6190. doi: https://doi.org/10.1016/j.tej.2018.01.006. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1040619017303330.

[60] U. El Mir, "Identification of a validation method for open source simulation tools and application of said method to the mvs: Multi-vector simulator-sector coupled systems", 2020.

[61] S. Schlesinger, "Terminology for model credibility", *Simulation*, vol. 32, no. 3, pp. 103–104, 1979.

[62] O. Balci, "Verification validation and accreditation of simulation models", in *Proceedings of the 29th conference on Winter simulation*, 1997, pp. 135–141.

[63] S. Robinson, *Simulation: the practice of model development and use*. Wiley Chichester, 2004, vol. 50.

[64] R. Pietroń, "Modelowanie symulacyjne", *Wybrane zagadnienia, E-Materiał PWr*, 2013.

[65] B. Mielczarek, "Modelowanie symulacyjne w zarządzaniu", *Symulacja dyskretna, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław*, 2009.

[66] M. Olan, "Unit testing: Test early, test often", *Journal of Computing Sciences in Colleges*, vol. 19, no. 2, pp. 319–328, 2003.

[67] J. Hommes, *Github repository for this project*. [Online]. Available: https://github.com/jimhommes/Spine_EMLab_COMPETES.

A
Documentation

This appendix serves as the documentation for Emlabpy and the SpineToolbox implementation. The coupling implementation in SpineToolbox and Emlabpy can be found on the Github [67]. In contrast to the rest of the report, this appendix will go in-depth on how to install, run and understand the code. This is described in a manner in addition to Chapter 4. If even more detail is desired on the workings of this project, the code itself should be consulted in which documentation is provided.

A.1. Requirements and Installation Instructions

To run and install the soft-linking, the following software is prerequisite:

- Python
- SpineToolbox[57]
- AIMMS (license required, academic license available)
- COMPETES (obtainable in accordance with TNO)
- Microsoft Access (or Microsoft Access Ready Driver)

The source code is openly available and can be downloaded through the Github [67]. Once downloaded, the folder can be opened as a Spine-project in SpineToolbox.

A.2. Running Instructions

To run a case study, first the input data has to be defined. Under the 'resources/data/' folder, template files have been supplied. These should be filled and the 'template_' should be removed from the file names.

If SpineToolbox is showing red exclamation marks, this indicates that there are problems that need to be fixed before running. Most likely, this will be fixed by the following actions:

1. File references should be checked. The blocks should refer to the correct paths, e.g., the 'EMLAB Init Data'-block should refer to the 'EMLAB Init.xlsx' file.

2. The SpineDB databases have to be initialized. This is done by clicking 'New Spine DB' on the 'DB EMLAB', 'DB COMPETES' and 'Simulation Configuration Parameters' blocks.

Once SpineToolbox shows no red exclamation marks, SpineToolbox is ready to be executed. However, to be able to run the full coupling, COMPETES has to be initialized. This is done by launching AIMMS and loading COMPETES. Once loaded, the procedure 'Setup_RESTAPI' has to be run. This procedure enables the HTTP request module from the AIMMS DataExchange library. This enables SpineToolbox to communicate with AIMMS and is necessary to let SpineToolbox run COMPETES.

The soft-linking is now ready to be executed.

The entire loop in SpineToolbox entails one year of execution time. Note that when running multiple years, the import blocks must not be run again. At the time of writing, a looping feature that would enable the automatic runs of multiple years is not available. To run multiple years, the blocks from 'Init EMLAB Clock' should all be run after the first run.

A.3. Emlabpy

In this section, a broad overview is given on Emlabpy and the packages are elaborated on. Package responsibilities and contents are described. If more detail is desired, the code and the comments should be consulted.

A.3.1. Class Overview and UML

Figure A.1 shows the UML diagram of the Emlabpy code. Not all functions and attributes are mentioned, but it gives an overview of the interconnections of packages and classes. In this section, details on the workings and definitions in Emlabpy is given per package.

Util

The 'util' package provides all utility Python files in regard to the operation of Emlabpy. There are three files: repository.py, spinedb.py and spinedb_reader_writer.py.

The repository is central in Emlabpy. It acts as a first point of interaction for all modules in Emlabpy regarding Python objects. It contains all loaded Python objects

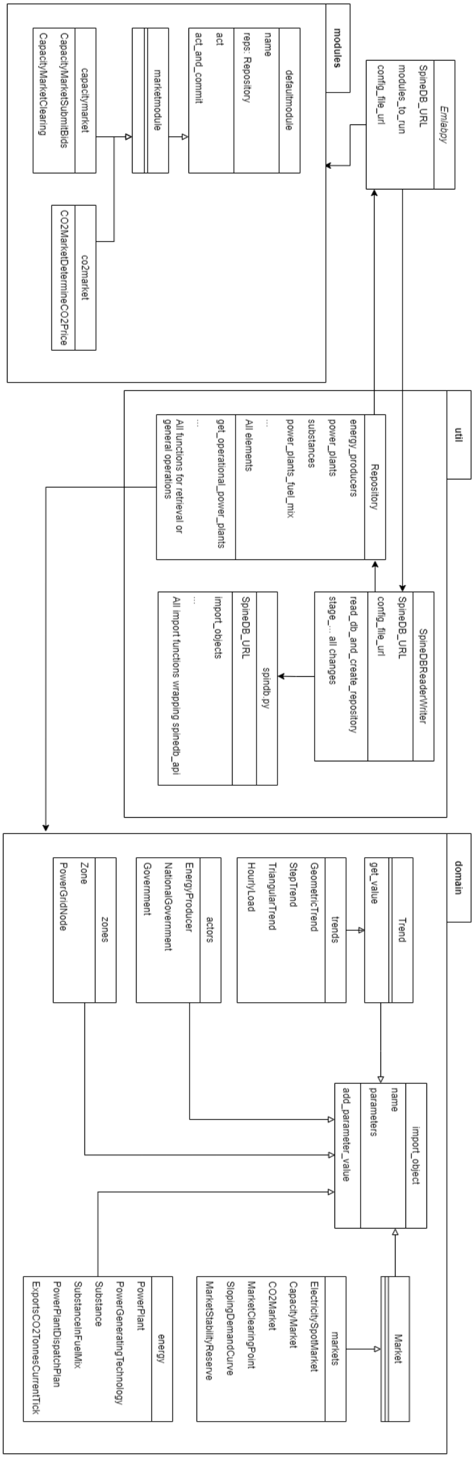


Figure A.1: The UML diagram for the Emlabpy implementation showing the classes, the packaging and the inheritance relations.

and functions that directly extract information from these objects. E.g., there is a list of power plants and a function that retrieves all operational power plants.

The spinedb file is a wrapper for the spinedb_api Python package developed by VTT. It enables interactions between Python and the SpineDBs from the SpineToolbox project.

This spinedb file is used by the spinedb_reader_writer file. As can be seen in Figure 4.1, this file handles all interactions between Emlabpy and SpineDB. The most important function it has is the 'read_db_and_create_repository' function, which reads the SpineDB and fills the repository with the Python object interpretations of the SpineDB values. It does so by using the database parameter definitions found under the 'domain' package per class.

Domain

The 'domain' contains all object definitions and the database-to-object mappings required for the SpineDB reader/writer class. All objects extend the class 'ImportObject' which provides a default interpretation if none is provided.

The objects are all defined per file. The 'actors.py' file, for example, contains all beings involved in the soft-linking, e.g., energy producers and consumers.

Modules

The 'modules' package contains all executions of the Emlabpy module as described thus far. The developed CO2 market and capacity market modules are defined here. The main Emlabpy file executes modules based on arguments provided by SpineToolbox.

The structure in the modules are constructed on top of the repository and the SpineDB reader/writer. The repository is accessed if objects are required and the SpineDB reader/writer is access for exporting results to the SpineDB.

A.3.2. Module Extension

If it is desired to extend Emlabpy with a new module, the existing modules and SpineToolbox's calling of these modules should be studied. Emlabpy contains a sequential execution of modules that depends on arguments provided to the executable. It can be seen in SpineToolbox that in order to call the CO2 market, the argument 'run_co2_market' is provided. The new module should also extend the Module class. This class contains structure and proper logging.

A.4. SpineToolbox

Section 4.2 provides some broad information on SpineToolbox and this implementation. This section goes in-depth on the workings of all blocks found in Figure 4.5.

A

A.4.1. File references, Importing and SpineDB

At the beginning of the entire sequence, all file references, Importer-blocks and SpineDB blocks are called. This entails initialization of the soft-linking: the SpineDBs are filled with initial data defined in the Excel sheets. The Importer-blocks contain the mappings from the Excel sheets to the SpineDB-structure.

At the top in Figure 4.5, blocks regarding configuration can be found. These blocks include the reference, import and storage of the simulation execution configuration and contains information like what years to run and what investment horizon to apply. The Excel files should be altered if a different scenario is desired.

It should be noted that in this SpineToolbox implementation there are two types of mappings that can be found in the COMPETES import blocks. This has to do with the fact that easy export is required in another block. This is further elaborated on in Appendix A.4.7.

For further information on the workings of Data Connection, Data Store or Importer blocks, the SpineToolbox documentation should be consulted.

A.4.2. Clock and Clock Increment

The 'Init EMLAB Clock' and 'Increment EMLAB Clock' Tool-blocks both refer to the same Python script clock.py. As the names suggest, the blocks are responsible for the clock central in the soft-linking that indicates the current execution year. In the EM-Lab SpineDB the object class, object and parameter 'SystemClockTicks' is initialized. The existence of the parameter with a certain value indicates the current execution year.

The increment simply adds the 'Step' defined in the central simulation configuration parameters to the current year. This is exported again to the EM-Lab SpineDB.

A.4.3. Initialization Preprocessing DB EMLAB

There were some structural difficulties in the different data organizations of EM-Lab and COMPETES. The 'Initialization Preprocessing DB EMLAB' script is a programmatic solution to some of these difficulties. 'Initialization' refers to the fact that this script is executed once directly after the Importer-blocks have finished.

The first challenge presents in the naming of fuels throughout COMPETES. In COMPETES, there exist two kinds of fuel-names: a fully capitalized name and a more elaborate normally capitalized name. The connection is described in the entries in the 'FuelMap'. As Emlabpy has no interpretation of these different names, a single name has to be prioritized. The 'replace_power_generating_technology_fuel_names' function is responsible for replacing the fuel names for the PowerGeneratingTechnologies with the correctly capitalized name.

Furthermore, this script extracts the correct VRE capacities and fixed O&M costs.

SpineToolbox's Importer-blocks are not capable of a conditional import. The table containing VRE capacities has entries per year and per country. SpineToolbox is unable to only import the current year and for the Netherlands, for example. This is done programmatically.

A.4.4. Preprocessing DB EMLAB

In contrast to the 'Initialization Preprocessing' block, the 'Preprocessing DB EMLAB' block is run every execution year. Mainly, it contains the script as described in Section 4.3.1 which sets the correct generator statuses.

It also tackles another problem originating from structural differences. In COMPETES, under installed capacities, there are some VRE capacities listed every year. This is due to COMPETES' aggregating of results: as changes are implemented over multiple years, the aggregate will show the current value. The script 'decom_power_plants_and_return_sum' will decommission all of these generators for EM-Lab and implement a single operational generator that works with the aggregate.

A.4.5. Emlabpy

Emlabpy has thoroughly been discussed in this report and this appendix. This section will only describe the way Emlabpy is called through the Tool-blocks of SpineToolbox.

In the soft-linking, Emlabpy is called twice. As mentioned, Emlabpy has a module structure and is able to execute separate modules by calling them through arguments passed to the executable. The blocks 'EMLAB CO2 Market' and 'EMLAB Capacity Market' call the CO2 market and capacity market respectively. They do so by calling Emlabpy with the arguments 'run_co2_market' and 'run_capacity_market' respectively. Emlabpy shows the handling of these arguments.

A.4.6. DB EMLAB to DB COMPETES

The 'DB EMLAB to COMPETES' script is described thoroughly in Section 4.3.2. All functions found will aid in either of the described purposes: export the CO2 price or export the capacity market revenues. As there is a structural difference in VRE and non-VRE properties in COMPETES, the functions to export capacity market revenues also differ in VRE and non-VRE.

A.4.7. Prepare COMPETES

The 'Prepare COMPETES' script prepares the Microsoft Access databases which will be read by AIMMS to run COMPETES. The script works by first copying the empty Access files, that contain the table structure, to the location where COMPETES will access them. The Python package pyodbc is used to execute SQL queries to import all values into the Microsoft Access databases.

As the entire SpineDB structure has to be exported programmatically to Access files, it was decided to build only two types of mapping in the Importer-blocks. Tables with simple value mappings were loaded as Spine objects with single parameter values. Tables with unique values per year or per other values are loaded using a SpineDB Map object. The same structure is used for Spine relationships. The tables and their unique index names have to be defined in 'COMPETES Config File' as the script will output the SpineDB values under that name. These two types of mappings are named Type I and Type II mappings.

This script contains a separate function to export the fuel prices to COMPETES. The reason for this is, in EM-Lab, the fuel prices are determined by a trend. To couple this generated value, this script uses the trend to regenerate these values and export them to COMPETES.

A.4.8. COMPETES

As stated, AIMMS is called through an HTTP request through the AIMMS DataExchange library. The COMPETES blocks call 'aimms_call_competes.py' with the argument which type (dispatch or investment) to run and for which year. A boolean is also passed on whether to include the 'look-ahead', or investment horizon, in the year of execution.

After the initial HTTP request has been sent, the script will send an HTTP request periodically requesting a status update. The status is printed to the terminal until the status mentioned is 'finished'.

A.4.9. COMPETES Output to EMLAB DB and COMPETES DB

Section 4.3.3 describes the main uses of this script. All functions in this script either execute one of the mentioned methods or aids in correctly interpreting the Excel sheet output of COMPETES.

A.5. Debugging

If debugging is desired, this can refer to two levels of depth. First, it could be the case that something is wrong in Emlabpy's execution. This means that it's desired to debug the code, preferably through an IDE. It could also be the case that in the broader sense a module in SpineToolbox is malfunctioning. In this case, a guide is provided towards finding where to look in the soft-linking.

A.5.1. Code

All scripts in the soft-linking can be executed through an IDE, if desired. In SpineToolbox, per Tool-block all arguments provided to the executable are listed. In case a SpineDB URL is referred, this URL can be determined by finding the file path of the SpineDB (found in the Database-block) and prepending this file path with 'sqlite:///'. The scripts describe what arguments are necessary.

When executed, the SpineDB will have to be studied through SpineToolbox. It is possible to study the database through a different reader, however SpineToolbox has an interpretation ready of the SpineDB SQL structure. This makes it the best choice.

A.5.2. Coupling

If there is unexpected output or independent module failure, the soft-linking will have to be studied. It is recommended to always back-track from the point where the soft-linking goes wrong. There are some points to keep in mind when going through SpineToolbox:

- **Check the terminal windows**
  If a script fails for whatever reason, SpineToolbox often does not detect failure of the execution. This means the coupling will continue, even though there might not be an output to continue with. Often this means that there is a cascade of failures through the run. To detect this properly, the terminal windows have to be checked for error messages. They also provide useful info, as information is constantly printed.

- **Check COMPETES output, or try running COMPETES separately**
  As described, the execution of COMPETES is a simple HTTP request in SpineToolbox, awaiting execution. If anything goes wrong on this side, SpineToolbox will not detect it. COMPETES also will not output error and warning messages to its console when it's executed by a HTTP request. However, the status of execution can still be checked. Lastly, it could be beneficial to try to execute COMPETES manually using data from the soft-linking. This way, if there is anything wrong with the data, COMPETES will output this to its terminal.

- **Check SpineDB consistency**
  In the current SpineToolbox implementation, the SpineDB contains all information on the current execution. If data is altered in an attempt to restart the coupling, it might be beneficial to simply reset the SpineDBs.

B

Data Structure

This appendix shows the used data structures in the soft-linking of this project. Both models require their own initialization data set but since there is overlapping initialization data there is also a ”Shared” data structure. In the images in Figure B.1, Figure B.2, Figure B.3 and Figure B.4 the classes with their parameters can be found.

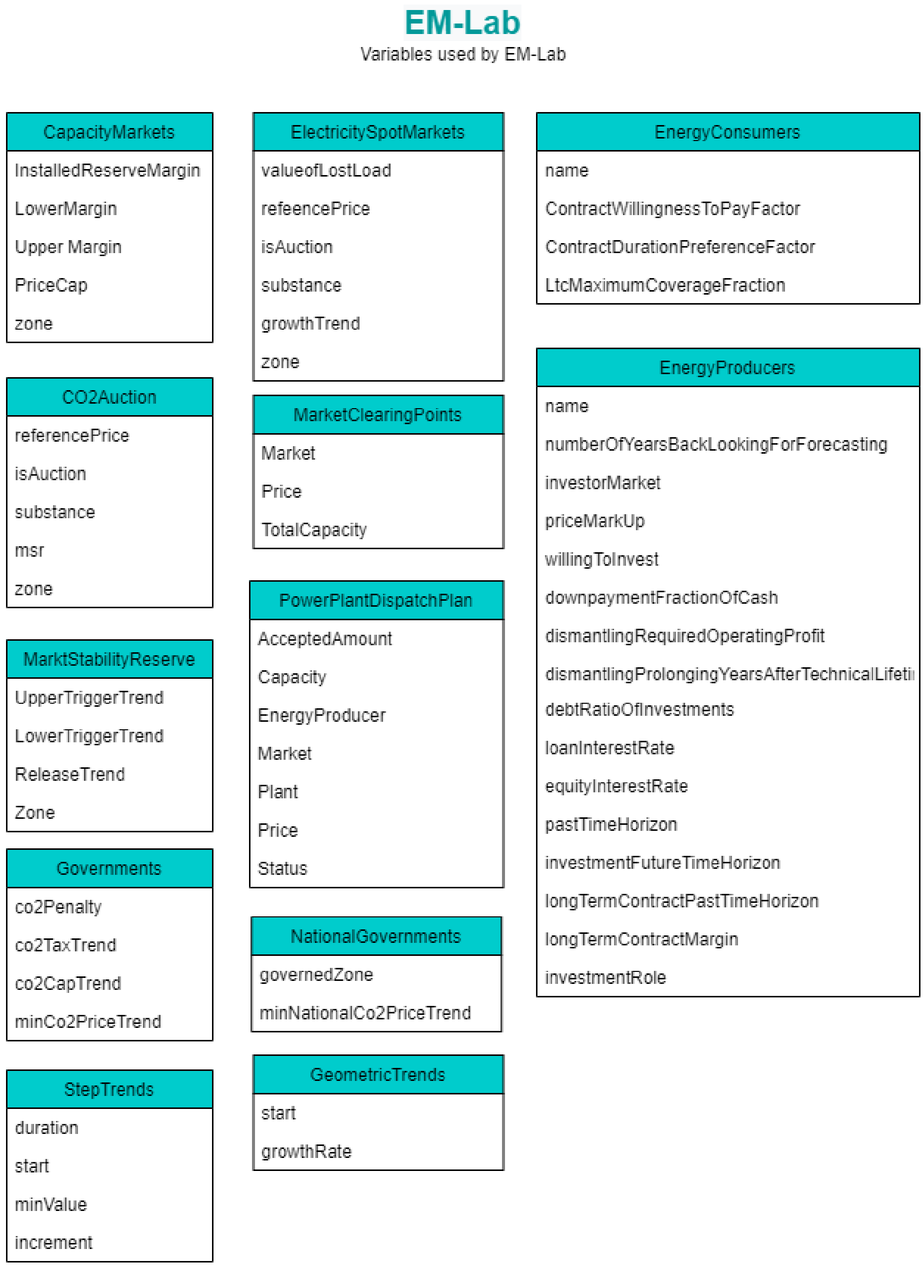


Figure B.1: EM-Lab data used in this project's SpineToolbox implementation.

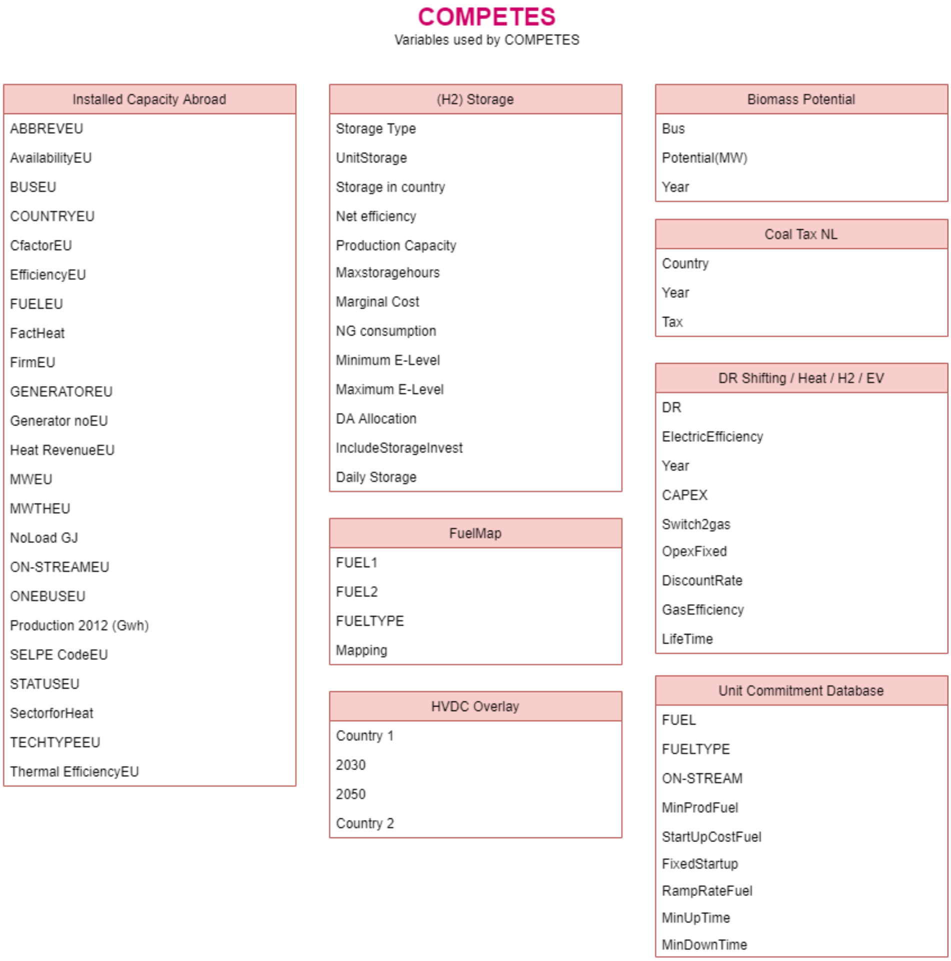


Figure B.2: Part one of COMPETES data used in this project's SpineToolbox implementation.

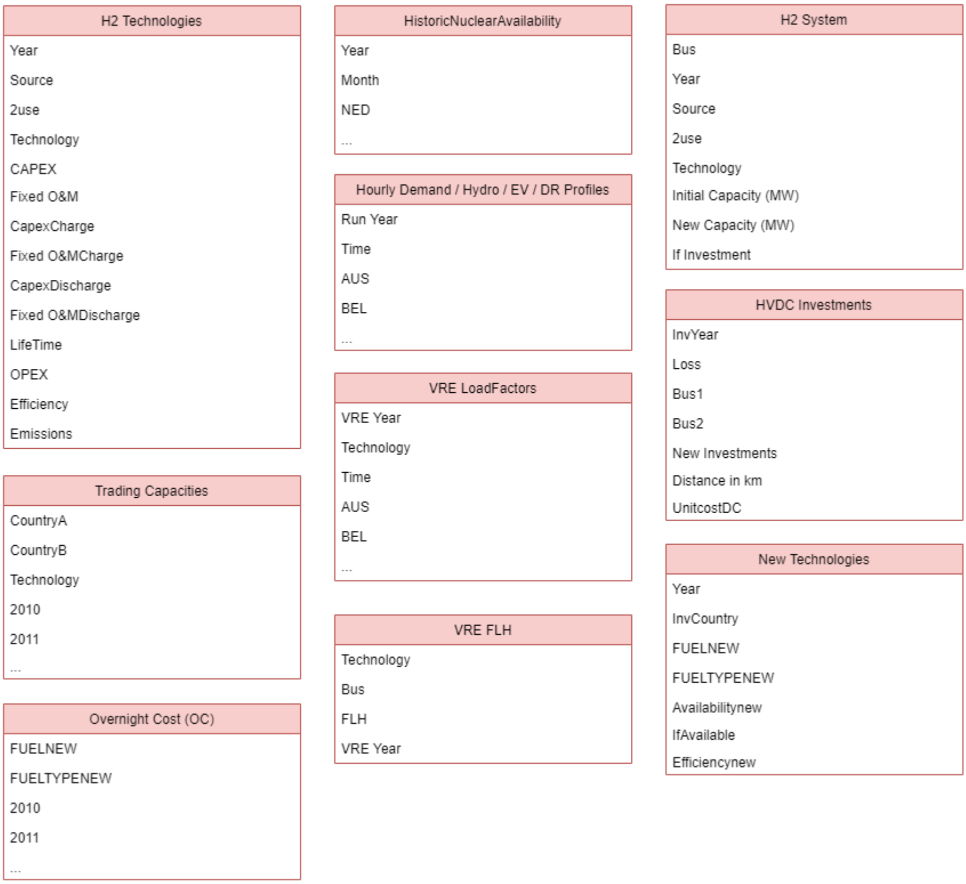


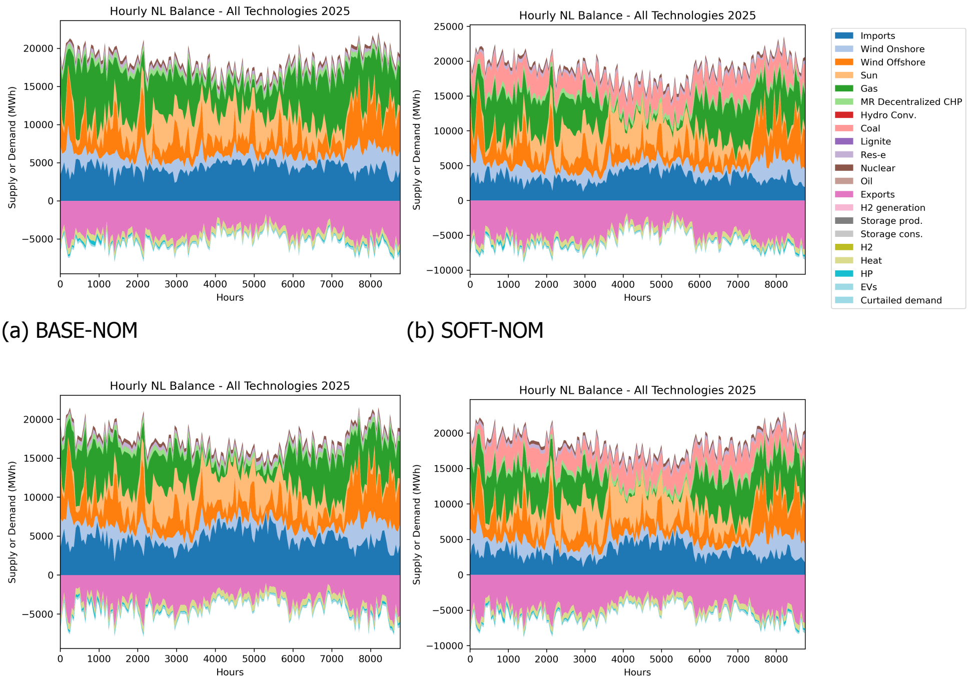
Figure B.3: Part two of COMPETES data used in this project's SpineToolbox implementation.

Figure B.4: Data used by EM-Lab as well as COMPETES in this project’s SpineToolbox implementation.

C

Additional Results

Some of the results were too specific or inconclusive for the report. However, they might give additional insight and have been provided in this appendix.



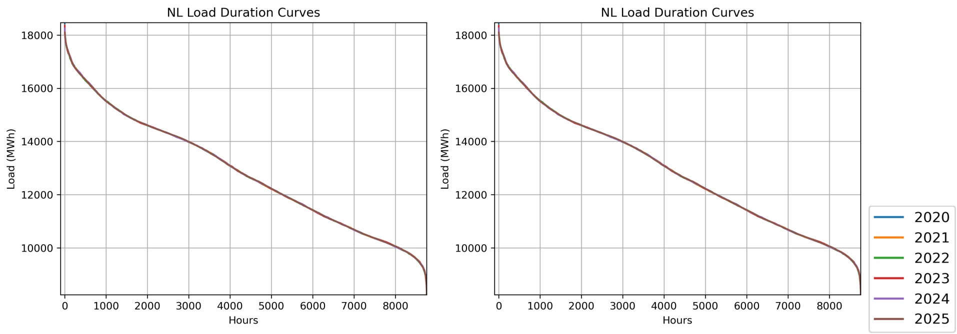
(a) BASE-NOM

(b) SOFT-NOM

(c) BASE-SCAR

(d) SOFT-SCAR

Figure C.1: The Dutch hourly balances per technology. Not included in the original report as it is specific and cluttering.



(a) NOM (b) SCAR

Figure C.2: The load duration curves for the non-scarcity and scarcity scenarios. They are not included in the original report as they only differ slightly.

D
Test Results

Section 5.2 describes the importance of static testing. This appendix elaborates on the static testing executed for this project. To be efficient, the focus lies on the testing of the core of Emlabpy and the coupling scripts (EM-Lab to COMPETES, COMPETES to EM-Lab, EM-Lab Pre-processing and EM-Lab Initial Pre-processing).

The Python testing framework pytest is used. This framework is available to Python as a package and contains all methods and objects necessary in order to do functional testing. Examples that are used are mocking, fixtures and multiple types of assertions.

Because of Spine's database approach, the tests are run using a dummy database prepared separately. This database is copied for every test. This way it's not necessary to define all data through the code. Interactions with the databases are tested through mocking and are matched to expected results from the dummy data. This dummy database is prepared using a separate SpineToolbox project, found under the tests folder.

A total of 96 unit tests were written achieving a total of 74% line coverage. Line coverage is the amount of lines of code that are executed through the written tests. This line coverage takes into account non-elementary functions, database interactions and more aspects which are not tested. These elements are either not practically testable or difficult, and with that costly, to test. Figure D.1 shows the coverage results.

Coverage report: 74%

Module	statements	missing	excluded	coverage ↓
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain__init__.py	0	0	0	100%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\actors.py	26	0	0	100%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\import_object.py	10	0	0	100%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\zones.py	5	0	0	100%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\util__init__.py	0	0	0	100%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\trends.py	75	1	0	99%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\util\repository.py	172	1	0	99%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\markets.py	74	2	0	97%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\util\spinedb_reader_writer.py	145	13	0	91%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\domain\energy.py	176	27	0	85%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\competes_to_emlab.py	216	72	0	67%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\emlab_to_competes.py	149	52	0	65%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\emlab_preprocessing.py	63	25	0	60%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\emlabpy\util\spinedb.py	114	53	0	54%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\emlab_initialization_preprocessing.py	60	29	0	52%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\helper_functions.py	6	4	0	33%
E:\Dropbox\workspace\Spine_EMLab_COMPETES\resources\scripts\spinedb.py	114	85	0	25%
Total	**1405**	**364**	**0**	**74%**

coverage.py v5.5, created at 2021-08-14 23:45 +0200

Figure D.1: This figure presents the test coverage done by unit testing. The percentage indicates how many lines of code have been executed through the use of static unit tests. A total percentage of 74% was achieved. The file paths containing 'Emlabpy' indicate files which contain core programming for Emlabpy. The other files under 'resources
scripts' are coupling scripts.