On the Modeling and Simulation of Future Energy Systems

Dissertation
zur Erlangung des Grades
des Doktors der Ingenieurwissenschaften
der Naturwissenschaftlich-Technischen Fakultät
der Universität des Saarlandes

von
Lukas Exel

Saarbrücken
2021

Tag des Kolloquiums:	5.11.2021
Dekan:	Prof. Dr. rer. nat. Jörn Walter
Berichterstatter:	Prof. Dr.-Ing. Georg Frey Prof. Dr. rer. nat. Marc Deissenroth-Uhrig
Vorsitz:	Prof. Dr. rer. nat. Andreas Schütze
Akad. Mitarbeiter:	Dr.-Ing. Lutwin Klein

LUKAS EXEL

ON THE MODELING AND SIMULATION OF FUTURE ENERGY SYSTEMS

If we will but exert the spirit of determination,
if we will but manfully maintain belief in our own works,
we can adjust our governments and industries to our ideals.

— Preface of Electrical World, Volume 79, 1922

ABSTRACT

DEUTSCH Die Endlichkeit fossiler Energieträger und deren Einfluss auf das globale Klima erzwingen eine Transformation des Energie- und Stromsystems – eine Umgestaltung, die getrieben ist durch neue Technologien, veränderte Zielstellungen der Akteure und politische Rahmenbedingungen, die als Bebauungsplan zu verstehen sind. Ein etabliertes Mittel zur Findung der Rahmenbedingungen sowie der Bewertung von Strategien zur Erreichung individueller Ziele ist die Nutzung von Modellen und deren Simulation. Genau wie die Systeme selbst müssen die Modelle und Methoden angepasst werden, um die Realität hinreichend genau abzubilden.

Auf diese Forschungsfrage zahlt diese Arbeit mit zwei aufeinander aufbauenden Schwerpunkten ein. Zunächst wird, ausgehend von einer holistischen Beschreibung des Energie- und Stromsystems, ein Überblick zum Forschungsstand von Modellierungsansätzen und Herausforderungen gegeben. Im zweiten Teil wird das Framework MOCES entworfen und implementiert. Basierend auf MODELICA erlaubt es eine holistische Simulation des Stromsystems auf Basis der Modellierung des individuellen Verhaltens der einzelnen Akteure sowie der physikalischen Systeme, auf die sie einwirken.

MOCES wird genutzt, um ein deutsches Stromsystem zu simulieren, das mit Speichern modifiziert wurde, die dem Ziel folgen, Strom möglichst lokal zu verbrauchen. Das erstellte Modell erlaubt, u.a., den Einfluss auf die Netzlast und -stabilität, den Strommarkt sowie die individuellen Erträge der Akteure zu bewerten.

ENGLISH The finiteness of fossil fuels and their impact on the global climate call for a transformation of the energy system, in particular the power system. This transformation is driven by new technologies, changing objectives of the actors involved and conditions by political frameworks. A classic means of designing such conditions as well as evaluating strategies for achieving individual objectives is the use of models and their simulation. Just like the systems themselves, the models and the underlying methods are in a state of change, and their applicability must be questioned. As a result, there is a need for new methods and tools for modeling and simulating complex energy systems to represent reality with sufficient accuracy.

The present thesis addresses this gap. First, based on a holistic description of the energy and power system, an overview of the current state of research of modeling approaches and their shortcomings is given. Second, the framework MOCES is designed and implemented. Based on MODELICA, it allows a holistic simulation of the power system based on the modeling of the individual behavior of each actor as well as the physical systems they act on.

Finally, MOCES is applied to simulate the German power system extended by storage systems that pursue the goal to use energy as locally as possible. The model allows, i.a., to evaluate the grid load and stability, the electricity market and the yields of the actors.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

1 INTRODUCTION

1.1 THE TRANSFORMATION OF THE ENERGY SYSTEM

The finite nature of fossil fuels and their unacceptable adverse effects on the global climate require a transformation of the ENERGY SYSTEM into one beyond fossil fuels. Fortunately, this transformation has already begun. It is based on new technologies, political objectives and, last but not least, a changing awareness of all the actors involved.

From the author's point of view, the transformation is driven, respectively made possible, by the following four key aspects:

A. ENERGY CONVERSION AND STORAGE TECHNOLOGIES (ECST). The technological foundation of an ENERGY SYSTEM beyond fossil fuels are technologies such as PHOTOVOLTAIC (PV) systems, wind power plants, energy storage systems and entities that link different ENERGY CARRIERS with each other, such as power-to-gas plants. Wind and PV systems are especially fundamental as they can provide enough electrical energy to supply the overall demand.

   However, those technologies bring a new complexity to the system that must be mastered. A critical aspect that increases the complexity is their dependency on the supply of renewable ENERGY CARRIERS. Wind and PV systems, for example, depend on the time variable and the quantity of wind and solar radiation respectively. Thus, their output is predictable, but their controllability is restricted to the limitation of the power output.

   Storage systems are the key technology to buffer the non-uniform feed-in of renewable power plants and thus enable the reduction of the capacity of conventional power plants. However, new challenges arise in determining an optimal operation strategy and optimal design parameters.

B. INFORMATION AND COMMUNICATIONS TECHNOLOGY (ICT). Regarding information technology, the foundation of the transformation of the energy system are cheap, fast, reliable, bi-directional and secure data connections to the individual entities, equipped with data-collecting sensors and technical facilities that are capable of influencing the energy consumption and production at ENTITY level. They make concepts such as DEMAND RESPONSE possible. These technologies evolve passive loads into flexible and active components of the energy system. They offer a more cost-effective operation of the distribution and the local networks, and they can be a cost-effective alternative to expanding the grid.

c. DECENTRALIZED ENERGY SYSTEMS (DESS) are not restricted to a decentralized energy production anymore. They instead reflect a general desire to balance the generation and consumption of energy as locally as possible, up to the goal to be as self-sufficient as possible from the actual system. This goal is an excellent example for the fact that private end customers, in particular, have targets other than the most cost-effective supply of energy, and that these adjusted targets lead to a change in the overall energy system.

d. MARKET DESIGN AND REGULATORY FRAMEWORK. The structure of a liberalized ENERGY SYSTEM, as well as its operation and further development, is the result of PARTIES within a market that interact with each other in order to achieve their own goals. To simplify matters, we will assume that the result of the interaction is the optimal solution for the given system.

   The ENERGY SYSTEM is strongly regulated by the legislator. Possible interactions are very limited and influenced by the regulatory framework.

   The legislature cannot only restrict action, it also has the possibility of providing incentives for activities through taxes and subsidies. For example, it can internalize external costs such as air pollution and therefore shift the behavior of the involved PARTIES.

The first three aspects are often referred to with the three buzzwords *decarbonization*, *digitalization*, and *decentralization*, respectively. These developments come hand in hand with a changing view of the ENERGY SYSTEM and its primary task. Until a few years ago, this task could be defined as follows:

> *Provide the required* FINAL ENERGY *to the consumer of energy within the different* DEMAND SECTORS.

and is now transforming into:

> *Provide an* ENERGY SERVICE *or* ENERGY-RELATED SERVICE *to the consumer.*

This redefinition detaches itself from the view that the consumers should merely be satisfied with the supply of energy. It targets to address the consumers' actual needs. Note that, for the intrinsic demand, solutions may exist that are only partially based on the provision of energy. Let us take the example of the provisioning of petrol for motorists here. It is not the motorist himself or herself who needs the gasoline. The gasoline is used to cover a demand for mobility that could also be covered by other solutions, such as by an electric car or train. In case of the actual need of the driver not being mobility, but reaching the workplace, then this demand could also be satisfied by suitable technologies for teleworking for example.

1.2 MODELING AND SIMULATION AS A KEY TECHNOLOGY

Depending on the perspective on the changing energy system, two fundamental questions arise. From the perspective of a stakeholder who is a part of the changing system, the primary concern is: "How to interact with the system in the most profitable way?" (i) From an energy policy or academic point of view, the question is: "How to design the system optimally?" (ii) Both questions are looking for an optimal solution - solutions that cannot be found in a straightforward way due to the complexity of the system. In the general case, it is therefore necessary to restrict oneself to weighing different options for action against each other. The underlying questions are then: "How is my performance when I behave in this or that way?" (iii) and "What is the performance of the system like when I design it this or that way?" (iv), respectively.

Whenever we are confronted with such kinds of questions, we use MODELS of the not yet existing system to get answers, or at least insights, as we are not capable of running experiments on the real system. Moreover, we use those answers as a basis for our decisions in the real world. The required type of MODEL depends on the kind of question. For answering the questions (i) and (ii) OPTIMIZATION MODELS are required. For the questions (iii) and (iv), a SIMULATION MODEL is needed.

Modeling means abstraction. The result of the modeling process is a model that is an abstract representation of the real world. If we have done everything correctly, it is valid within a defined context which is derived from the question we would like to answer. Cellier [1] calls this context EXPERIMENTAL FRAME and refers to a concept by Zeigler [2]. The experimental frame defines a set of experiments for which the model is known to be valid. Zeigler consistently requires that the description of the frame is an elementary part of a model and that it can only be used if the question to be answered, an experiment, is within this frame.

Unfortunately, we cannot prove the validity of a model of a system still to be designed. We have to trust that the modeler has chosen a proper level of abstraction. We have to assume that his knowledge of the real system is sufficient to describe its behavior in the context of a specific question. If we understand the system as a set of interacting actors, we must trust that no interaction has been mistakenly neglected in the modeling process. We must believe that the so-called system boundary has been set correctly.

Considering the changes mentioned in Section 1.1, it should be noted that precisely these system boundaries are undergoing dramatic changes as the level of interaction between the individual actors is increasing significantly. This dynamic shift is confronted with modeling and simulation approaches in the field of energy system analysis, which have been developed for decades to answer specific questions. For instance, the first version of the MARKAL model family [3] was established in the early 1980s to determine cost-minimal energy system designs. This ultimately casts doubt on the validity of the models. Pfenninger et al. describe this challenge as follows:

> *[One recommendation] is to rethink whether current methods are appropriate for twenty-first century challenges. [...] The danger is that proven and established methods gain primacy because of their familiarity. Many of the large models used today have existed since before [...] the advent of many of the large-scale changes in the energy sector underway in the early twenty-first century. Both the challenges and the tools available to deal with them are being transformed at an accelerating pace, and energy modelers must be careful not to be left behind. (Pfenninger et al. [4])*

1.3 MAIN CONTRIBUTION

1.3.1 *What this Thesis is About*

The main contribution of the thesis is the development of a modeling framework, called MODELING OF COMPLEX ENERGY SYSTEMS (MOCES), allowing a simulative investigation of an ENERGY SYSTEM and especially its core, the POWER SYSTEM. The framework is developed as a MODELICA library. It uses an approach that allows to extend MODELICA-based simulation models with an agent-based simulation environment but ensures the consistency with the MODELICA LANGUAGE SPECIFICATION (MLS). Models developed with MOCES can therefore be processed with any MODELICA IDE.

The framework spans over the seven CONCEPTUAL DOMAINS of the POWER SYSTEM defined by the NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST) [5]: *Markets*, *Operations*, *Service Provider*, *Transmission*, *Distribution*, *Customer*, and *Generation*. It enables the simulation of the interaction and behavior of the various ENTITIES within and between the different domains. This interaction is based either on a physical coupling or on communication.

MOCES realizes the modeling of ENERGY SYSTEMS with a three layer approach. The PHYSICAL LAYER, modeled in *pure* MODELICA, representing the physical coupling, the BUSINESS PROCESS LAYER, modeled with an agent-based approach, serving the interaction based on communication, and the INFORMATION LAYER providing a consistent environment to the entities.

The modeling approach is not restricted to a specific modeling depth, neither spatial nor temporal nor in the modeling depth of the ENTITIES themselves. However, in the time domain, we focus on a simulation time span of up to one year, in the spatial domain, on models spanning nations with a geographical resolution slightly larger than the NUTS-3-level. The modeling depth is restricted to effects with time-constants larger than 10 s.

Models developed with MOCES have a static structure that does not change during simulation. However, dynamic changes in the structure can be represented to a limited extent by using models that can switch between several predefined structures.

A unique characteristic of MOCES is the explicit modeling of the BALANCE MANAGEMENT PROCESS (BMP), which is the core process ensuring the balance between production and consumption within liberalized energy markets.

For this main contribution, the following six aspects will be discussed in detail:

HOLISTIC DESCRIPTION OF THE ENERGY SYSTEM (Chapter 2) In the introduction, we already pointed out that the validity of an ENERGY SYSTEM model depends on the correct level of abstraction and that the chosen abstraction level has to be reconsidered in light of the transformation of the ENERGY SYSTEM. Motivated by that, Chapter 2 contributes a holistic description of the ENERGY SYSTEM, and in particular the POWER SYSTEM. It elaborates the core principles on which the reliability of the system is based and which can be assumed to continue to be its backbone in the future. The focus on the POWER SYSTEM results from the expectation that it will be the backbone of the ENERGY SYSTEM beyond fossil fuels.

The holistic description of the ENERGY SYSTEM is based on the NIST CONCEPTUAL MODEL, the ENTSO-E ROLE MODEL and the description of the BMP as provided in related documents by the EUROPEAN NETWORK OF TRANSMISSION SYSTEM OPERATORS (ENTSO-E).

Finally, given the holistic description of the ENERGY SYSTEM, we will derive the challenges for an ENERGY SYSTEM MODELING AND SIMULATION FRAMEWORK (ESMSF) that is capable of representing the current and upcoming characteristics of the POWER SYSTEM. These challenges are summed up by Pfenninger et al. [4] with the key phrases: *resolving details in time and space*, *uncertainty and transparency*, *complexity and optimization across scales* and *capturing the human dimension*.

EVALUATION OF EXISTING MODELING METHODS AND TOOLS (Chapter 1.5 and 3) The holistic description and the challenges that have been worked out in Chapter 2 serve to evaluate existing methods and modeling approaches. We consider two different established modeling approaches. The OPTIMIZATION MODEL APPROACH and the SIMULATIVE APPROACH. Both methods are used in different contexts (EXPERIMENTAL FRAMES) to answer a wide variety of questions but differ in their purpose. The OPTIMIZATION MODEL APPROACH uses OPTIMIZATION MODELS to determine an optimal configuration of a system. The SIMULATIVE APPROACH uses SIMULATION MODELS and is limited to forecast the behavior of a system.

We will not attempt a more detailed grouping of the different modeling approaches and models in this thesis. This grouping overview would not be useful, as most models are developed for a specific application and are therefore hardly comparable.[1] We therefore concentrate on describing some classical modeling approaches using the example of established models. These established models are evaluated in the context of this thesis. We work out their advantages and disadvantages concerning the identified challenges.

DESCRIPTION OF MOCES (Chapter 4) The third aspect is the description of the framework MOCES and the decisions that led to the chosen design, both based on the *holistic description* and the *evaluation of existing approaches*. The result is the already mentioned design of MOCES, which follows the SIMULATIVE APPROACH and is built on top of the modeling language MODELICA.

DESCRIPTION OF THE PROCEDURE FOR DEVELOPING A MODEL INSTANCE IN MOCES (Chapter 4.5) As MOCES is a modeling framework, an additional step is required to build a concrete model instance for a given context. A context could be, for example, *Model the German power system to investigate the increasing share of renewable energy systems*. An abstract procedure will be developed and described for this step.

EXEMPLARY APPLICATION OF MOCES (Chapter 5; running example in Chapter 4) We contribute a concrete application of MOCES and the aforementioned procedure for the example of the following context: *Modeling the German power system to investigate the effect of decentralized flexibility management*. The resulting model instance is based on the ELMOD-DE[2]

1 This fact has been indirectly substantiated by numerous review papers which limit themselves to a pure listing and description of the models, cf. [6], [7].

2 OPEN SOURCE ELECTRICITY MODEL FOR GERMANY

model provided by the DIW[3] and data provided by the DWD[4], ENTSO-E[5] and the four German TSOs[6]. The chapter gives an insight into the depth of detail of MOCES and discusses the insights gained by the simulation.

RELATED TOPICS (Chapter 3.3.3) Further important aspects are touched upon in this thesis but are not discussed in detail. One such example is the challenge of handling large MODELICA models. We will show that this problem does not originate in the modeling language MODELICA itself but in the way models are processed by standard MODELICA IDES. Furthermore, we will point to some promising approaches for improving model processing.

1.3.2 *What this Thesis is not About*

Even though the goal of MOCES, and thus the present thesis, is to develop a framework that allows the holistic modeling and simulation of the POWER SYSTEM, it is indispensable for a clear scope to set boundaries for the aspects of the system that are taken into account. In the following, we will mention these aspects and justify the choices and restrictions that have been made.

A. ICT-DEPENDENT MONITORING, CONTROL, AND PROTECTION SYSTEMS: ICT technologies, one of the key aspects that contribute to the transformation of the energy system, have a particular impact on how grid operators run the TRANSMISSION GRIDS and DISTRIBUTION GRIDS. In this context, *operation* refers to the three primary responsibilities of a GRID OPERATOR: *monitoring*, *control*, and *protection*, and related sub-responsibilities such as (local) voltage control or wide-area damping control. The use of ICT-based systems to perform the associated tasks leads to several challenges. The following ones are named by Mueller et al. [8], for example:

   - Effect of latencies, packet loss or failure to the control application.
   - Impact of cybersecurity threats on the reliability of the POWER SYSTEM.
   - Choice of ICT infrastructure to fulfill the requirements.

   Those challenges refer to fundamental issues which have been discussed generally or in other specific contexts since decades. They

3 DEUTSCHES INSTITUT FÜR WIRTSCHAFTSFORSCHUNG (GERMAN INSTITUTE FOR ECONOMIC RESEARCH)
4 DEUTSCHER WETTERDIENST (GERMAN METEOROLOGICAL OFFICE)
5 EUROPEAN NETWORK OF TRANSMISSION SYSTEM OPERATORS
6 TRANSMISSION SERVICE OPERATOR

are subject of research activities, such as HYBRID SYSTEMS, TIME DELAY SYSTEMS, and NETWORKED CONTROL SYSTEMS.

One of the main methods used to solve the aforementioned challenges is modeling and simulation. In this case, detailed modeling of the grid and the communication network is required. Therefore, a typical methodology is to use a co-simulation approach interlinking a network simulator, such as ns-3[7] with a grid simulator, such as PSLF[8]. An overview of different approaches is given in Mueller et al. [8].

For a clear and manageable scope, the present thesis does not deal with the questions in relation to ICT technologies mentioned above. MOCES-based models can therefore not be used to answer research questions in this area. We are convinced that, in order to address those challenges, no holistic modeling and simulation of the energy or power system is required. Instead in-depth modeling and simulation of the specific sub-parts can be used, assuming fixed boundary conditions and scenarios. Those boundary conditions and scenarios in turn may be derived from more holistic models and/or based on worst-case considerations.

B. STRUCTURAL CHANGES AND LONG-TERM CONSIDERATIONS: The structure of the systems considered in MOCES is assumed to be static and defined. It is therefore not suitable for investigating long-term developments over years or decades as it is possible with models like LEAP[9] or MARKAL[10].

C. INVESTIGATION OF REAL SYSTEMS: The focus of this thesis is on the development of a modeling framework that allows studies of the POWER SYSTEM, not on an examination of an existing system, such as the German one, in the context of a concrete research question. Accordingly, this thesis does not contain reliable recommendations for the development of the German energy system. The findings presented in Chapter 5 have to be understood in this context.

1.4 DEFINITION OF KEY TERMS

In the following section, we will introduce central terms and define how they are used in this thesis. We do so because there exist very different interpretations of the respective terms.

7 See `https://www.nsnam.org/` (last accessed: 3 June 2021).
8 Commercial power system analysis software provided by GENERAL ELECTRIC.
9 LONG-RANGE ENERGY ALTERNATIVES PLANNING SYSTEM
10 MARKET ALLOCATION

1.4.1 *Energy System*

From a traditional perspective, the term ENERGY SYSTEM is defined as:

> *process chain [...] from the extraction of primary energy to the use of final energy to supply services and goods., (Pfenninger et al. [4])*

A recent definition states the following:

> *[...] a system primarily designed to supply energy-services to end-users., (Wikipedia.org [9])*

Both definitions imply that an ENERGY SYSTEM spans across nations, even continents. The first definition is characterized by an image of the energy system in which the demand for energy is met by a complex process, starting with the extraction of fossil fuels followed by several conversion and transport processes. The second definition focuses on the final state of this transformation chain and removes the mandatory reference to PRIMARY ENERGY and fossil fuels.

Both definitions suggest that the ultimate goal of an ENERGY SYSTEM is to provide ENERGY SERVICES. This is an interesting aspect, as considerations of ENERGY SYSTEMS often end with the examination of the USEFUL ENERGY or even FINAL ENERGY required by the different ENERGY DEMAND SECTORS. At first glance, this distinction seems to be hairsplitting. However, it leads to a shift in the SYSTEM BOUNDARY of the ENERGY SYSTEM and an immense increase in complexity.

This increase in complexity can be observed in the resulting different main use cases of the ENERGY SYSTEM. Either it is *"to provide natural gas to a domestic house"*, or it is *"to provide heat to the house"*, or even *"to provide thermal comfort to the residents of the house"*. Looking at the second use case from a service provider perspective, the possibilities to fulfill the demand increase dramatically compared to the first case, in which the provider is limited to specific ENERGY CARRIER. In the third case, the provider could also offer warm clothes to the residents –to give a somewhat stupid example–, or provide better heat insulation –to provide a better example–.

This example shows that some ENERGY SERVICES can also be replaced by alternative solutions, which are not anymore restricted to deliver energy. Especially these alternative technologies no longer fit into the process-oriented first-named definition. We, therefore, prefer the second definition, which detaches itself from this restriction and does not imply a concrete design.

Unfortunately, the term *Hybrid Energy System* is highly unrelated to the term ENERGY SYSTEM as it is mainly used to describe stand-alone systems that include some DES. We will not use this term later on.

1.4.2 *Power System*

Based on the definition of the ENERGY SYSTEM given in the previous subsection, we define the POWER SYSTEM as the subset of all elements of the ENERGY SYSTEM that are directly linked to the ENERGY CARRIER *electricity*. This linkage is explicitly not limited to the technical units directly connected to the grid. It implicitly includes all ACTORS whose behavior influences those technical units. It also includes a set of rules that restrict or prescribe the interaction between the ACTORS. In particular, it involves the processes that ensure the balance between production and consumption of electrical energy and the ELECTRICITY MARKET.

1.4.3 *Smart Grid*

The term SMART GRID is not uniformly defined. Following the OFFICE OF ELECTRICITY DELIVERY AND ENERGY RELIABILITY (OE), it

> *[...] generally refers to a class of technology people are using to bring utility electricity delivery systems into the 21st century, using computer-based remote control and automation. (Energy.gov [10])*

This definition reflects the American view on its transport and distribution network, which has suffered from a lack of investment in recent decades that have resulted in increased failure rates.

In the European, and especially in the German, community, the term SMART GRID is used in a broader sense. The BDEW[11] gives the following definition:

> *Ein Smart Grid ist ein Energienetzwerk, das das Verbrauchs- und Einspeiseverhalten aller Marktteilnehmer, die mit ihm verbunden sind, integriert. Es sichert ein ökonomischeffizientes, nachhaltiges Versorgungssystem mit niedrigen Verlusten und hoher Verfügbarkeit.*
>
> *A smart grid is an energy grid that integrates the consumption and feed-in behavior of all market participants connected to it. It ensures an economically efficient, sustainable supply system with low losses and high availability.*

This definition reflects the view of the German energy industry, which sees the POWER SYSTEM as the backbone and driver of the so-called *Energiewende* that addresses the entire ENERGY SYSTEM. However, this definition is rather a general objective for the ENERGY SYSTEM, and it remains

11 BUNDESVERBAND DER ENERGIE- UND WASSERWIRTSCHAFT (GERMAN ASSOCIATION OF ENERGY AND WATER INDUSTRIES)

unclear how exactly the *energy grid* differs from the ENERGY SYSTEM and its status quo.

We will therefore use the term SMART GRID as defined by the OE to encompass the evolvement of the grid.

1.4.4 *Energy Sector*

The use of the term ENERGY SECTOR varies widely. Typically, it is used in combination with a list of *sectors* that are to represent the entire or at least the core of the ENERGY SYSTEM. Hardly any definition is coherent in itself. In the German context, the following three sectors are mentioned in particular: electricity, heat, and transport, e.g., by Quasching [11, p. 8]. Lund et al. [12] add the sectors cooling, industry and buildings. These definitions of the sectors are unfortunate, as transport is an ENERGY DEMAND SECTOR, electricity is an ENERGY CARRIER, heat is either a carrier or a USEFUL ENERGY DEMAND, industry is again an ENERGY DEMAND SECTOR, and energy consumed by buildings belongs to several ENERGY DEMAND SECTORS.

A more consistent definition of the term ENERGY SECTORS is used in the context of classical ENERGY SYSTEM MODELS, such as MARKAL. Here, the ENERGY SECTORS, often referred to as ENERGY DEMAND SECTORS, describe different non-overlapping consumer groups. In Germany, typically the following four sectors are given: household, industry, transport, and commerce trade and services.

In general, it can be noted that the term sector is used to cut out and name a certain, context-dependent part of the energy system. We will use the term ENERGY SECTOR to encompass all ENTITIES directly connected to a specific ENERGY CARRIER such as electricity –to remain constant with the fuzzy usage of the term.

1.5 PRIMER ON MODELING, SIMULATION, AND OPTIMIZATION

As SYSTEM, MODEL, SIMULATION and OPTIMIZATION and their related notions are essential terms within this thesis, we would like to make some introductory remarks on them to clarify how they are used.

The term SYSTEM can be defined in a number of ways. Even within our context, one might refer to different definitions, depending on the view on the ENERGY SYSTEM one is considering. The definition that fits best with respect to how the ENERGY SYSTEM is understood in this work, and that at the same time also characterizes how we approach the topic of modeling the ENERGY SYSTEM is the following:

> *a regularly interacting or interdependent group of items forming a unified whole. (Merriam-Webster.com [13])*

Let us consider this definition from a high-level perspective and accept that there are no completely isolated items. We then have to conclude that there is no boundary of the system that divides the system from the rest, and that there are not various systems; there is only one system –the universe. Such a conclusion might be astonishing, given the definition of a system by Brian Gaines: "A system is what is distinguished as a system" (cited from Cellier [1]). Cellier provides an interpretation of this statement and accompanying explanations as follows:

> *Whenever we decide to cut out a piece of the universe [...] we define a new 'system'. A system is characterized by the fact that we can say what belongs to it and what does not, and by the fact, that we can specify how it interacts with its environment. (Cellier [1])*

This contradictory statement sheds some light on how scientists and engineers deal with the complexity of systems. They cut an infinitely small piece out of the universe, by defining the SYSTEM BOUNDARY, and by specifying how the system interacts with its environment beyond that boundary, typically by defining inputs and outputs. Moreover, with the help of this small piece, they try to understand the behavior of this piece and how it interacts with the surrounding –that is what scientists do. If they conclude that they have understood its behavior, they start to *design* small pieces and put them in a dedicated place of the universe to influence it –that is what engineers do.

The potential pitfall is obvious. If one cuts a system, one will loose interactions and characteristics. In the best case, one knows exactly where to cut and how to avoid loosing relevant interaction and properties, but in general this cannot be guaranteed.

What scientists and engineers actually do when they cut the world into pieces, is to transfer a part of the system into a representation which is manageable, a MODEL. They model a part of the system to investigate it and they use a model to design a part of the system. This approach can work perfectly, but it can also fail dramatically if the part which is designed based on a model is interacting with the rest of a system through inter-dependencies not considered in the model.

For all that, we will use the term SYSTEM as it is defined by Cellier, a definition from which it follows that each MODEL is a SYSTEM. This definition allows us to use the terms *system design*, *system analysis* or *system theory* as they are typically used in an engineering context. Nevertheless, we emphasize that there are one or typically several transformation steps required between the designed system and the system which is put into the real world and vise versa.

We should internalize this finding when we sum up the goal of all research effort dedicated to energy systems: to design an optimal energy system, or at least to transform the current one in a better performing

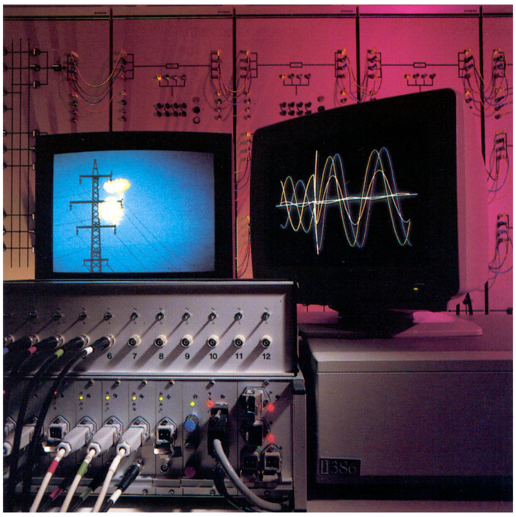


Figure 1: Photograph of a physical grid model (in the background) and measurement equipment (in the foreground) as used during the 90s at the Chair of Energy Supply at Saarland University.[15]

one. We should also internalize that we need a MODEL for the system that includes all necessary dependencies, otherwise we will fail.

The term MODEL is not restricted to mathematical models. It also includes other types of models. Aside the mathematical, there are three more types of models [14]:

- *Physical* models: a physical object that mimics an excerpt of the behavior of the system, e.g., a model of a wind turbine blade used for aerodynamic tests or a model of the electric grid as shown in Figure 1.
- *Mental* models: a representation of the external world and its behavior that we as humans develop during our lifetime. Be it to catch a ball that is coming towards us or to make decisions about our future professional life.
- *Verbal* models: a more concrete statement about the behavior of the system, e.g., the efficiency of a PV-plant descends by 0.5% per degree centigrade.

In this thesis, the term MODEL is used as synonym for a mathematical MODEL, as mathematical concepts are used to describe the component parts of the model and their interaction. More precisely, we will mostly use dynamic, nonlinear, and implicit models. Accordingly, we can assume that a direct (mathematical) analysis of the models will fail, due to their complexity, and that we will not be able to find a closed formed solution. Therefore, we need another method to extract the dynamic behavior of the model for a set of inputs, we will use SIMULATION.

Before looking into the method SIMULATION, we will first have a look at the different *quality levels* of models. Roughly speaking, the *quality level* is a quantity for how suitable the MODEL is for SYSTEM design purposes. Models that can be used for design purposes have a special characteristic. They can be combined to build a system and one can assume that it is a valid representation of the system one is designing.

To give an example: Mathematical models for electronic circuits have a very high quality level. We can combine models for the individual components, such as a resistor or capacitor, to build a model of a novel system never built before. In this case, we can safely assume that the model is a valid representation of the system.

At the other side of the spectrum, there are mathematical models which cannot be used for design purposes at all. In the last decade, engineers suspiciously eyed computer scientists: they have had enormous success in modeling complex systems with generic methods, such as deep neural networks, computing power, oodles of data and often without any knowledge of the modeled system at all. The crucial point for our context is that such models can be used to predict the behavior of an existing system, but they cannot be used for system design. Usually, they do not even attempt to provide a realistic model of the individual components of the system and their interplay, but instead merely model, more or less directly, the complex relation between input and output of the system. For example, consider a machine translation model that has been trained to translate German sentences into English sentences. It can be used to predict a translation for a new German sentence that has never been seen before by the model. However, despite such a model potentially being inspired by how the human brain works, it does usually not model how humans translate sentences and does not refer to human language processing. It cannot be modified to translate into a language that does not exist yet.

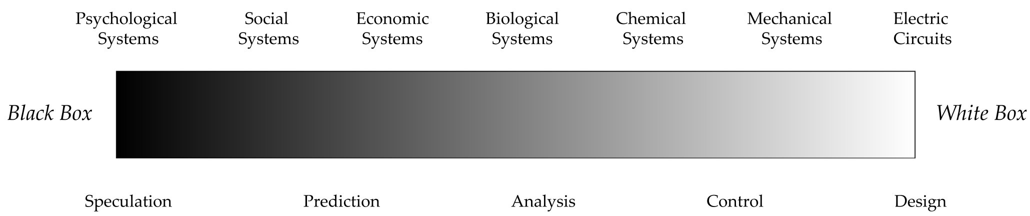


Figure 2: Spectrum of Modeling and Simulation, adapted from Cellier [1].

This *spectrum* of modeling and simulation was introduced by Karplus [16] at the end of the 70s. Cellier took it as the basis for his version of the spectrum in the 90s, which is shown in Figure 2. He used the terms *white* and *black box* to indicate how well the respective systems are understood and how adequately the internal structure of the model represents the internal structure of the system. We notice that the MODELS for the

ENERGY SYSTEM span from the right end almost to the left end of the spectrum.

SIMULATION is a method to determine the dynamic behavior of a model given a defined environment. This definition is highly related to the term *experiment*. However, *experiment* typically indicates that a *physical model* rather than a *mathematical model* is used. Therefore, the term *computer experiment* is sometimes used as a synonym for SIMULATION. In general, the outcome of a single SIMULATION performed by a SIMULATOR is the prediction of the behavior of the model, typically in two dimensions: time and space.

The concepts described so far and their logical sequence, also known as MODELING AND SIMULATION (MS)-chain, are illustrated in Figure 3 (gray blocks). If MS is used to find the optimal design of a system or to optimize the model of a system, the MS-chain is extended by an optimization loop and embraced by an outer loop representing the actual modification of the SYSTEM, e.g., a new version of an internal combustion engine or a modified POWER SYSTEM.

As Figure 3 shows, there are two possible pathways of the optimization loop: in the inner loop only parameters of the model are changed, while in the outer loop also the structure of the model is modified. However, a clear distinction between these two modifications is rarely possible. If one considers, e.g., the RC circuit given in Figure 6, the distinction seems clear. The modification of a parameter would be the adjustment of the capacitance, the modification of the structure would be the addition of another electrical component, e.g. an inductance. Often, however, a MODEL can be created that represents several structures implicitly and also includes a parameter allowing to change the structure. In the example of the RC circuit, this would be a switch which deactivates parts of the circuit.

The loops refer to a method called simulation-based optimization. In this context, the question how to tune the parameters or redesign the entire system in a systematic way arises. Depending on the complexity of the model, different methods can be used. Barton and Meckesheimer [18] formulate the optimization problem as:

$$\min_{\mathbf{p}\in\Omega} g\left(p_1, p_2, \ldots, p_n\right) \tag{1.1}$$

with each p_i being one of the design parameters $\mathbf{p}$ and Ω the set of possible values for $\mathbf{p}$. $g(\cdot)$ is the response function of the simulation. Typically, the response function is a user-defined function $h(\cdot)$ based on measured system performances $f(\mathbf{p})$, such as the mean of a specific value. In general, $g = h(f(\cdot))$ is not known and its value has to be obtained by performing a simulation run. If the model includes stochastic elements, a sufficient number of simulation runs is required. Depending on the characteristic of $g(\cdot)$ and Ω, different optimization strategies

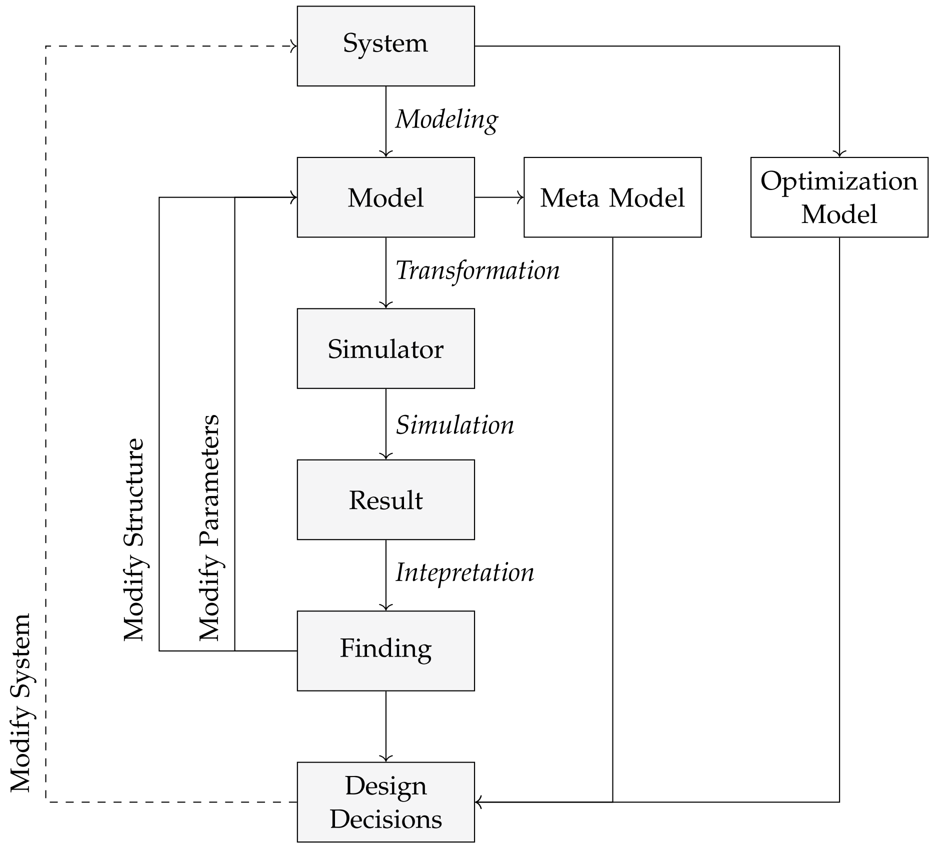


Figure 3: MODELING AND SIMULATION chain and its relation to different options to find an optimal system design, extended version of the original in Cellier and Kofman [17].

can be performed [18], including random search, meta-heuristics, and gradient-based strategies.

One strategy that can be applied to optimization problems if Ω as well as $g(\cdot)$ is continuous is to use a METAMODEL. A METAMODEL is an approximation of $g(\cdot)$, typically deterministic and cheap to calculate, e.g., a simple regression model that is linear in its model parameters β as given by Kleijnen [19] as:

$$\begin{aligned} \widetilde{g}(\mathbf{p}) = \beta_0 &+ (\beta_1 p_1 + \ldots + \beta_n p_n) && (1.2a)\\ &+ (\beta_{1,2} p_1 p_2 + \beta_{1,3} p_1 p_3 + \ldots + \beta_{n-1,n} p_{n-1} p_n) && (1.2b)\\ &+ \left(\beta_{1,1} p_1^2 + \ldots + \beta_{n,n} p_n^2\right) && (1.2c) \end{aligned}$$

$$g(\mathbf{p}) = \widetilde{g}(\mathbf{p}) + e \qquad (1.3)$$

The main idea of METAMODEL-based optimization is to find an appropriate approximation function $\widetilde{g}(\cdot)$ for $g(\cdot)$ that minimizes the error e

and use this function instead of the original one to determine the optimal design parameters **p**. The non-trivial challenge is thus to determine and validate an appropriate function $\widetilde{g}(\cdot)$. The optimization methods that can afterwards be applied to $\widetilde{g}(\cdot)$ to determine the optimal design parameters **p** are again dependent on the form of $\widetilde{g}(\cdot)$.

Using a METAMODEL that is linear in its *model* parameters, such as the model given in Equation (1.2), allows an easy fitting of the model. If the model is also linear in its *design* parameter **p**, such as the one in Equation (1.2) without the quadratic terms in (1.2b) and (1.2c), efficient optimization algorithms like the SIMPLEX METHOD can be used to determine the optimal design parameters.

This special class of models is also known as OPTIMIZATION MODELS. Compared with the models considered before, their primary focus is not to determine a time-dependent behavior, but to be helpful in a decision problem. These types of models, especially LINEAR OPTIMIZATION MODELS, are the backbone in the area of ENERGY SYSTEM models. Contrary to the METAMODEL, they are typically not built as a simplified version of a complex model. Scientists build the OPTIMIZATION MODEL in a direct way for the considered SYSTEM and restrict themselves to the chosen class of optimization problem, such as the aforementioned linear OPTIMIZATION MODEL. This alternative approach to design an optimal system is also depicted in Figure 3. Contrary to MODELS as defined so far, OPTIMIZATION MODELS do not necessarily describe the temporal and spatially resolved behavioral of the modeled system in an explicit way. Typically, these two dimensions, space and time, exist only at a very abstract level set by the modeler. To distinguish between MODELS that require a SIMULATOR to be interpreted and OPTIMIZATION MODELS, we strive to use the terms SIMULATION MODEL and OPTIMIZATION MODEL in a consequent manner.

At first glance OPTIMIZATION MODELS and SIMULATION MODELS seem to be highly separated from each other, but mixtures of both are also possible and common. SIMULATION MODELS can include OPTIMIZATION MODELS, and OPTIMIZATION MODELS can include SIMULATION MODELS.

1.6 PRIMER ON ACAUSAL MODELING

Looking at a SYSTEM, it is intuitive to see it as a system of systems, or, alternatively as elements that are linked together by connecting their initial interfaces to the environment and thus forming a new system, cf. Figure 4.

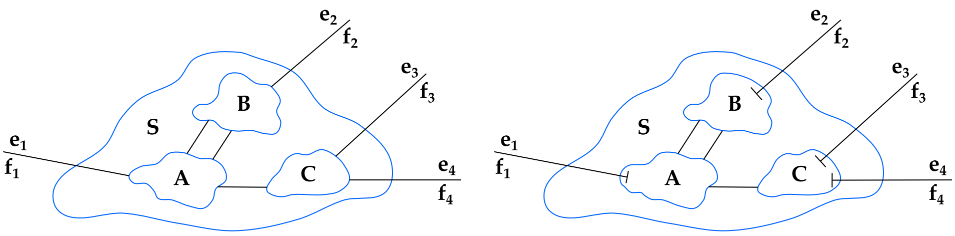
In the year 1961, Paynter and Briggs [20] called the process leading to the described view of systems *reticulation*[12], not modeling. *Reticulation* addresses the decomposition of a larger system into interconnected elements, and thus also the composition of elements to a larger system, and

12 *Reti* is Latin for (fish) net [20].

emphasizes the perspective that a system is a network of interconnected elements

In the simplest case, the individual sub-systems have dedicated inputs and outputs and, if several are linked, the CAUSAL DIRECTION between them is defined. Thus, it is determined which sub-system issues the *cause* and which one consumes it and reacts to it with an *effect*.

Since causality thus ultimately indicates which variables depend on which variables and determines a calculation sequence, knowledge of the CAUSAL DIRECTION is helpful in modeling and in many cases even indispensable. Unfortunately, in some cases the causal direction is not known. For example, if we consider an electrical resistance, there are no inputs or outputs, and the causal direction is unclear —Is it the voltage drop that causes a current flow, or is it the the current flow which causes the voltage drop? The causal direction is no inherent property of the resistance and it is not even a property of the mathematical model we typically use, Ohm's Law. However, as we will see, if we look at the resistance as an *element* embedded in a complete circuit, we *can* determine the causal direction.



(a) Causal direction not known (b) Causal direction partially known

Figure 4: Abstract depiction of a system of systems connected by *energy bonds* represented by bond graphs, adapted from Paynter and Briggs [20].

Interfaces of physical systems, such as an electric pin or a mechanical flange, are therefore designed in a different way, based on two fundamental aspects of physical systems named by Paynter and Briggs [20]. The first aspect is concerned with the types of linkages. Linkages of physical systems can be described by so-called *energy bonds* [20]. An *energy bond* represents a linkage between two elements, whose CAUSAL DIRECTION is not yet known. To describe this type of linkage, two variables are needed, one representing an *effort* and the other one representing a *flow*. An abstract example of a system description using energy bonds is shown in Figure 4a. As the name suggests, an *energy bond* represents the transfer of energy between systems, which can be calculated based on the two variables: *effort* and *flow*. If the *causal direction* is known, respectively added to the representation of the bond, as partially done in Figure 4b, the *energy bond* is equivalent to a *bilateral signal flow*, cf. the notation given in Figure 5.

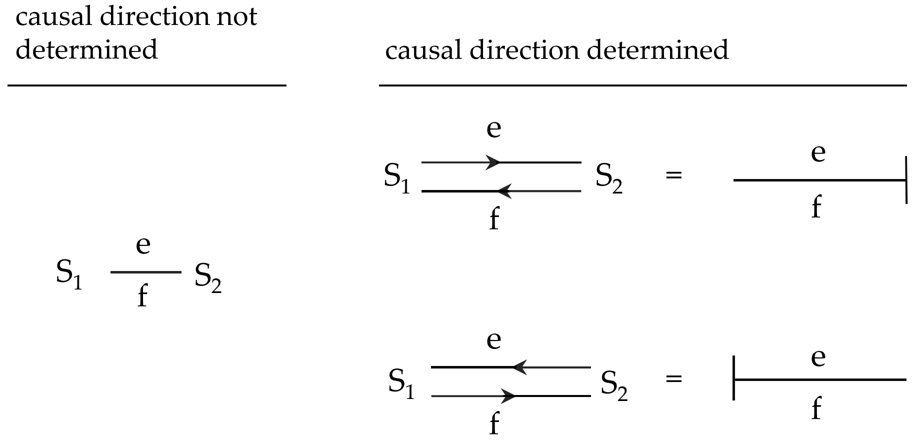


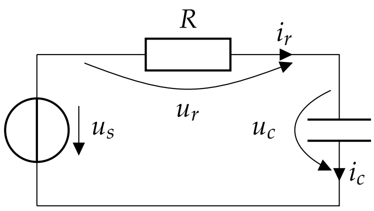
Figure 5: Notation for bond graphs, adapted from Paynter and Briggs [20].

The second aspect recognized by the authors is the finding that a system whose model includes *energy bonds* without specified CAUSAL DIRECTION cannot be *fully analyzed* in contrast to a system with known causal structure and must therefore be transformed into a network only including *signal flow* linkages. Paynter and Briggs do not give a detailed description of what is meant by *fully analyzed*, but we can assume that they allude to analog simulation and the first approaches of digital computer-based simulation as presented, e.g., in Åström et al. [21].

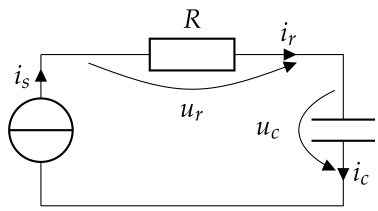
In the late 1990s, the terms *causal* and *acausal* models were introduced to distinguish between models in which the CAUSAL DIRECTION is determined or not yet determined, respectively. In particular, the terms are used in conjunction with modeling tools that support either only signal flow linkages, such as SIMULINK, or signal flows and energy bonds linkages, such as OPENMODELICA. As Janschek [22] states, the terms *causal* and *acausal* are common but misleading as *causality* is defined differently in system theory[13] and typically all physical systems are causal. Janschek [22] therefore recommends to distinguish between models with determinate or indeterminate causal structure. We would even suggest to use the term *not yet known* instead of indeterminate, because the model typically implies a causal structure that is not yet known to the modeler.

From the modeler's point of view, a modeling tool that supports *energy bonds* has great advantages. One can model the system without knowledge of the causal structure and can also maintain the structure. In addition, individual elements of the overall model can be reused in other models. This is possible even if the causal structure of the new model differs fundamentally from that of the original model.

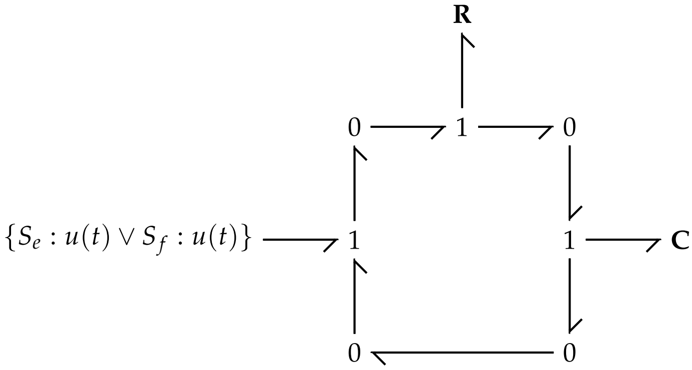
13 In system theory, a system is called causal if, at any point in time, the output of the system is only effected by the evolution of the input up to this point in time. Otherwise it is called acausal.



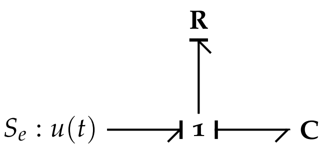
(a) RC circuit connected to ideal voltage source.



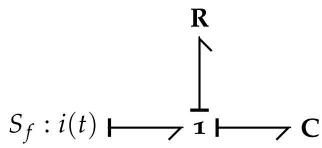
(b) RC circuit connected to ideal current source.



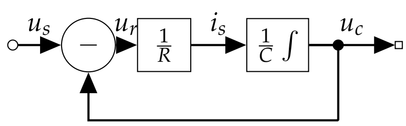
(c) Not yet simplified bond graph for RC circuit with ideal current or ideal voltage source. Causal direction is not determined.



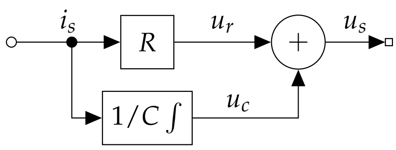
(d) RC circuit connected to ideal voltage source represented as bond graph.



(e) RC circuit connected to ideal current source represented as bond graph.



(f) RC circuit connected to ideal voltage source transformed into signal flow diagram.



(g) RC circuit connected to ideal current source transformed into signal flow diagram.

Figure 6: RC circuits in different model representations.

We will provide a simple example to clarify the differences between the two modeling approaches by walking through Figure 6 which shows RC circuits in different model representations. In Figure 6a, the RC circuit is connected to an ideal voltage source, in Figure 6b it is connected to an ideal current source. The models maintain the structure of the circuit. Figure 6c shows the same systems, but in a not yet simplified bond graph. Please note that the half arrow tips do not define the energy flow direction but only a reference direction. Simplifying the bond graph and determining the CAUSAL DIRECTION[14] results in the graphs given in Figure 6d and 6e. We would like to draw attention to the causalities of resistance. While in 6d a voltage drop causes a current flow, in 6e a current flow causes the voltage drop. The different directions of causality are particularly evident in the signal flow diagrams given in Figure 6f and 6g, which can be derived from the respective bond graphs. While systems depicted in this way have little or no relation to the structure of the system, as represented in 6a and 6b respectively, their further analysis with simulation is straightforward, as already stated by Paynter and Briggs [20].

14 A precise description of how causality can be determined is dispensed with.

2 FROM THE TRANSFORMATION OF THE ENERGY SYSTEM TO THE CHALLENGES REGARDING MODELLING AND SIMULATION

The introduction briefly outlined the fundamental changes in the energy system, often summed up with the three keywords *Decarbonization*, *Digitalization*, and *Decentralization*. In the Sections 2.1 and 2.2, we will systematically examine these fundamental changes. Section 2.1 starts with a high-level description of the energy system and sums up the expected transformation of it within the next decades which are required to achieve the climate action targets. We elaborate that one subsystem, the power system, is playing a key role in transforming the energy system. Therefore, we will describe the current state of this part of the system in more detail in Section 2.2. The Subsections 2.2.7 and 2.2.8 describe in detail the most important actors in the electricity system, their roles and the processes used to ensure a stable energy supply.

Based on an overview of the challenges of the evolution of the power system in Section 2.3, we will derive the challenges for an ESMSF capable of performing reliable statements about the performance of a future energy system in Section 2.4.

2.1 TRANSFORMATION OF THE ENERGY SYSTEM

2.1.1 *Overview on the Current State*

Figure 7 shows the German energy system in 1995, given as a Sankey diagram. The representation is ideal for describing the German energy and climate action targets and the necessary transformation of the energy system to fulfill them. The illustration is consistent with pre-renewable energy systems in which the demand for energy is met by a complex process, starting with the extraction of fossil fuels followed by several conversion and transport processes. If the abstract representation of the system is linked to real technical units, the following can be noted:

Different energy carriers are converted into each other via technical units, often accompanied by a loss of energy. This conversion is typically only possible in one direction. Apart from power plants that produce electricity, we can assume that higher demand for a specific energy carrier is accompanied with a need to expand the conversion units that supply this energy carrier. This consideration arises from the fact that all energy carriers, except electricity, can be stored to the extent that

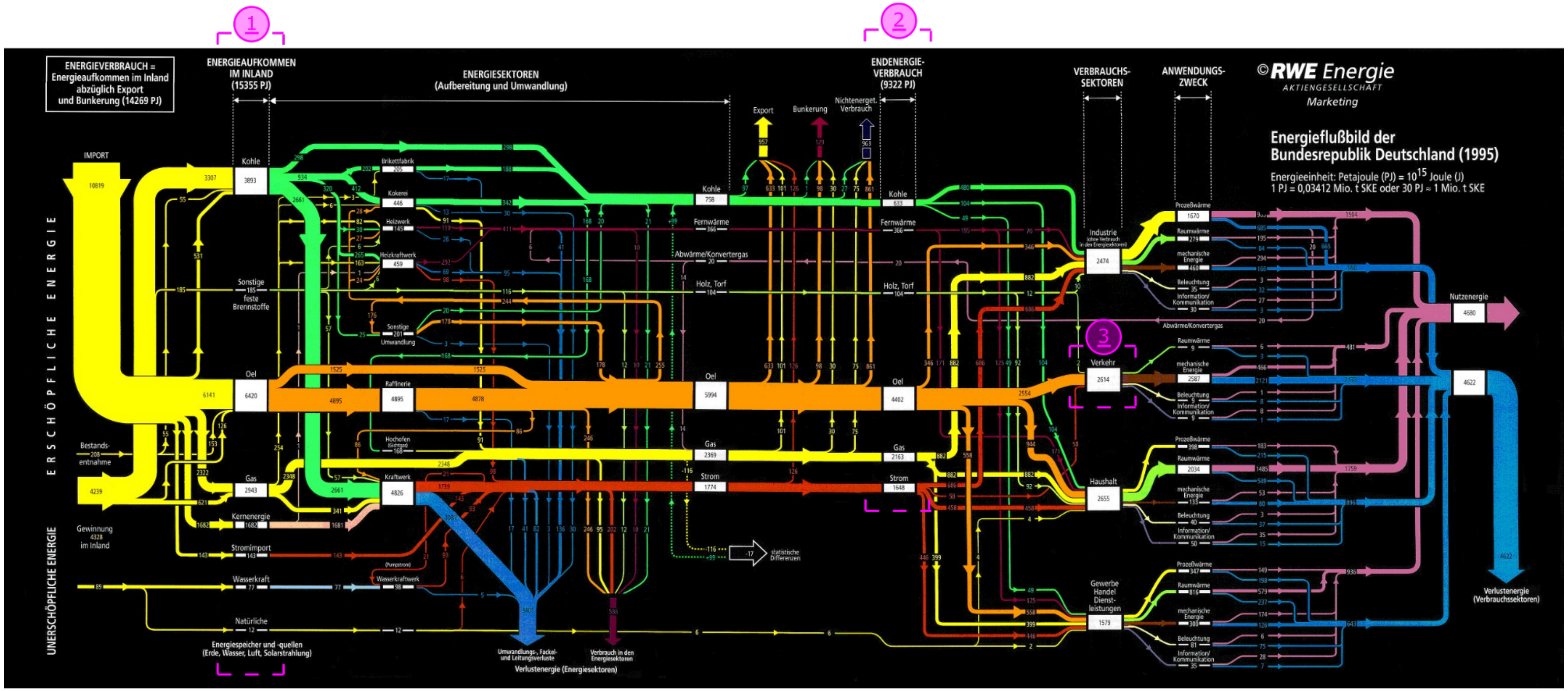
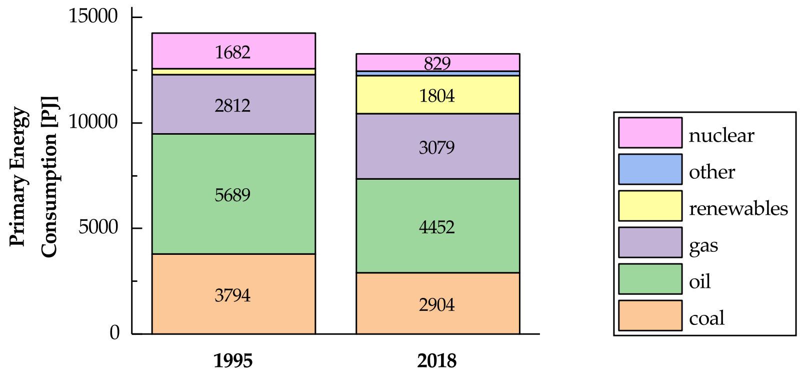
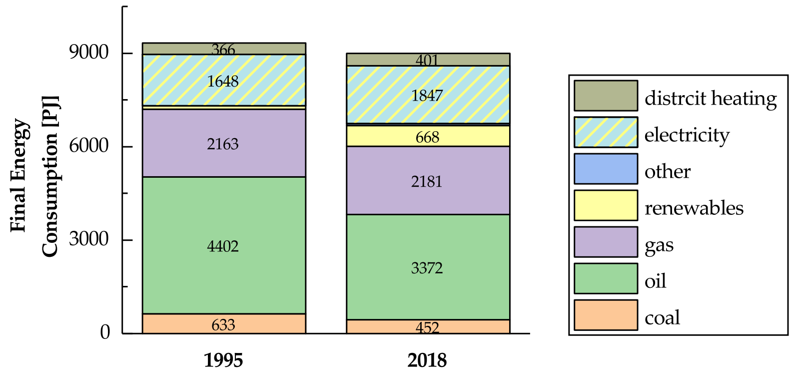


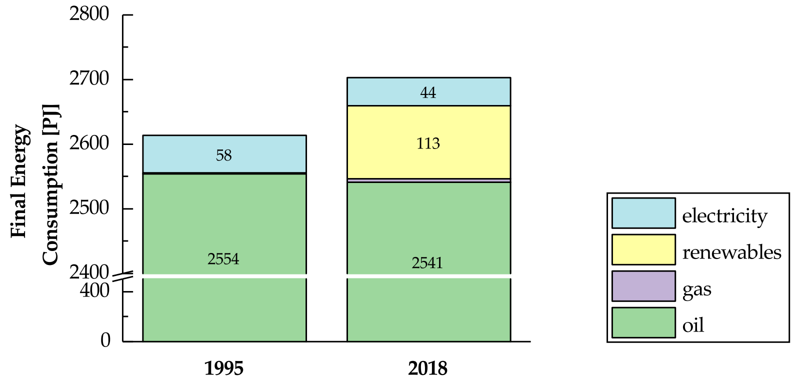
Figure 7: Overview of the energy flows of the German ENERGY SYSTEM in 1995, cited after [23, p. 4]. The areas marked in pink, respectively their development between 1995 and 2018, are highlighted in Figure 8.



(a) Changes in PRIMARY ENERGY consumption, cf. ① Fig. 7.



(b) Changes in FINAL ENERGY consumption, cf. ② Fig. 7. Electricity is yellow-hatched as in 2018 it is partially based on renewable energy sources, cf. Fig 9.



(c) Changes in FINAL ENERGY consumption for the DEMAND SECTOR transport, cf. ③ Fig. 7.

Figure 8: Selected changes for energy quantities given in Figure 7 between 1995 and 2018. Data for 1995 and 2018 is taken from AG Energiebilanzen [24] and differ minimally from those in Figure 7.

allows for a continuous and high utilization rate of the conversion units even if demand is dynamic.

The energy flows of the different ENERGY CARRIERS require a transport infrastructure, usually tailored to the respective ENERGY CARRIERS. The expenditure for this infrastructure can hardly be deduced from the amount of energy it transports due to different costs for different carriers. Even for a specific energy source, the infrastructure does not necessarily scale with the amount of energy. Especially in the case of electricity, the transport distance and the maximum generation level have a decisive influence.

The four DEMAND SECTORS each group a large number of different technical units, which in turn convert an ENERGY CARRIER into USEFUL ENERGY. Except for the transport sector, cf. Fig 8c, all DEMAND SECTORS are connected to the various ENERGY CARRIERS.

2.1.2 *Climate Targets*

The action targets for Germany and the year 2050 can be summed up as follows [25]:

- Reduction of greenhouse gas emissions by 80–95% compared with 1990[1],
- Increase of the share of renewable energies in gross final energy consumption to 60% (at least 80% for gross electricity production) compared with 1990,
- Reduction of primary energy consumption by 50% compared with 2008.

Looking at the system in 1995, cf. Figure 7, which is based mainly on fossil fuels, and the slightly less fossil fuel-dominated system of 2018, cf. Fig 8, this target seems at least ambitious.

2.1.3 *Challenges*

Achieving the goals named in the previous section can only be made by drastically reducing the use of lignite, hard coal, petroleum and, ultimately, natural gas. If one accepts that savings in final energy consumption are limited[2], a shift towards renewable electricity or renewable fuels

1 In addition to the ENERGY SYSTEM, agriculture also contributes to not negligible emissions. To achieve this goal, therefore, not only changes in the energy system are necessary.

2 In DEMAND SECTOR transport, there has been no reduction in FINAL ENERGY consumption since 1995. Demand in the industry sector is even increasing by 3%. Private households −13%, commerce, trade, and services −10%.

is inescapable. If we look at this conversion from the perspective of the individual consumption sectors and use the very abstract presentation in the Figures 7, the transformation appears to be feasible for the DEMAND SECTORS industry, household, and trade, commercial, and services. They are already linked to all ENERGY SECTORS and shifting to other ENERGY CARRIERS 'only' requires a different weighing of the possible paths through the ENERGY SYSTEM. A greater emphasis on the electricity path therefore inevitably raises the question of the impact of this weighting on the infrastructure of the ENERGY SECTOR electricity. In particular, if one considers the relatively low proportion of energy that flows via the ENERGY CARRIER electricity to the DEMAND SECTORS. In combination with the supply-dependent, and therefore fluctuate character of PV and wind power plants, a multitude of questions arise. Especially as the average supply is not evenly distributed geographically. There is instead an imbalance between load centers and regions with high supply. In this context, two partially opposing developments can be observed: a solution at European level, closely linked to the goal of a nation-wide OVERLAY GRID, or a solution in which the primary premise is a local balance between production and consumption. In both pathways the ENERGY CARRIER electricity is the backbone of the transformation of the ENERGY SYSTEM.

The transport sector has a crucial role to play in meeting climate action targets, but there is hardly any significant improvement. Looking at the energy flows in Figure 7, the dominance of the ENERGY CARRIER oil is evident. This dominance has not changed in the last 20 years, and the forecast for this DEMAND SECTOR is negative concerning achieving the action targets, cf. 8c. There are basically two pathways to overcome the dominance of fossil fuels. The first one is based on renewable synthetic fuels that substitute fossil fuels. Here, it is possible to have substitutions that do not require any changes of the combustion engines, such as biodiesel, or substitutions that need certain but no fundamental changes in the design of the combustion engines and related systems, such as hydrogen.

The second pathway is the substitution of the combustion engines by electric motors and the resulting electrification of the transport sector. Both paths, with certain limitations, are already technically possible today, but play a minor role in the existing system. Mainly due to higher costs or due to limited service quality, e.g., regarding range (electric cars) or coverage of stations. Especially with the second option, the importance of the electrical sector increases. But many of the synthetic fuels are also based on hydrogen produced by electrolysis.

Looking at the given diagrams of the ENERGY SYSTEM, the novelty of the so-called *Sektorenkopplung*[3] appears to be questionable at first glance.

3 German for coupling of ENERGY SECTORS.

With the exception of the DEMAND SECTOR transport, all DEMAND SECTORS are linked to all ENERGY SECTORS. Even the electrification of the transport sector is 'only' the integration of a new process chain into the energy system. The same holds for technologies named with terms such as Power-to-Heat, Power-to-Ammonia, or Power-to-Gas. The sector coupling is ultimately the electrification of the energy system in which the efficiency losses of the conversion processes from and to the carrier electricity can be accepted since the volatile generation of electricity from renewable energy plants has no influence on the emission of greenhouse gases and has no MARGINAL COSTS.

Beside that electrification of the ENERGY SYSTEM, the *Sektorenkopplung* also leads to a more dynamic system in which the paths are traversed depending on the current situation.

2.2 TRANSFORMATION OF THE POWER SYSTEM

2.2.1 *Intrinsic Characteristics*

The POWER SYSTEM has some intrinsic characteristics that we would like to start with as they are essential to understand the key figures in Section 2.2.2 and the further explanations in this section.

ELECTRICITY CAN HARDLY BE STORED: The supply and consumption of electrical power must be balanced at all points in time. The system itself has no significant inherent storage capacity. The only inherent storage capacities are the rotating masses of the synchronous generators connected to the grid. Their capacity is so small that even a slight deviation of just a few percentages between production and consumption would cause the system to collapse in a matter of seconds[4]. Therefore, the balance must be ensured, among others by the use of continuous controllers that regulate the output of conventional power plants. The lack of storage capacity within the power system is closely tied to the fundamental difficulty of storing electrical energy. Therefore, electrical energy storage systems have played and still play only a minor role.

THERMAL POWER PLANTS ARE SLUGGISH: The time constants of thermal power plants are about two orders of magnitude larger than the system time constant of the electrical grid, respectively the turboset mentioned above. [5] The start-up of power plants from zero

---

4 A typical acceleration time constant of a synchronous generator is 10 sec. It is defined as the time required to accelerate the turboset of the power plant from nominal speed to standstill with nominal torque. In an approximate way, it can therefore be stated that the energy stored in the grid is sufficient to supply it for 10 sec.

5 This statement is based on the typical maximum load gradients of conventional power plants in partial load operation.

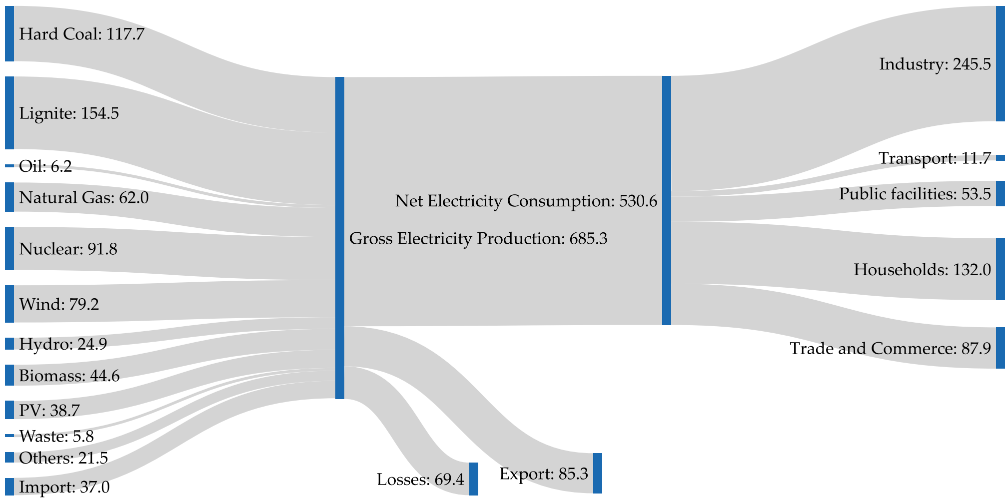


Figure 9: Electricity production and consumption in TWh in Germany for the year 2015; data is taken from [26].

production takes up to several hours depending on the initial state. The ratio of these two time constants determines that the POWER SYSTEM cannot be operated purely reactively with respect to the current situation. The planning of the production requires a certain LEAD TIME.

ELECTRICITY IS A COMMODITY: Electricity is traded on markets. It is produced because a customer is willing to pay a fixed price for an agreed amount of energy, provided at a specified time. Trading volumes and rates result from the interaction of a large number of players at marketplaces. Some of the players are not even capable of consuming or producing electricity themselves.

2.2.2 *Key Figures*

In order to provide an insight into the German POWER SYSTEM, we would like to take a more detailed look at the electricity path of the ENERGY SYSTEM shown in Figure 7. To that end, Figure 9 presents the overall production and consumption of electrical energy for the year 2015. It shows the individual ENERGY CARRIERS used to generate electricity on the left side, and the breakdown into the different DEMAND SECTORS on the right side. According to Federal Ministry for Economic Affairs and Energy (BMWi) [26] and as shown in Figure 9, in 2015 about 30 % of the electricity production was generated from renewable energy sources (Wind, Hydro, Biomass and PV). This share was 3.4 % in 1990, 6.25 % in 2000 and 16.51 % in 2010.

Due to the dynamic consumption and as electricity is difficult to store at a large scale, the production capacity of all power plants must be con-

	Raw data		Corrected data		
Energy carrier	Full-load hours	%	Full-load hours	%	Used avail. factor
Hard Coal	3537.6	40.4	**3879.0**	**44.2**	91.2
Lignite	6623.6	75.6	**6950.2**	**79.3**	95.3
Oil	2474.9	28.3	**2591.5**	**29.6**	95.5
Natural Gas	2286.1	26.1	**2203.2**	**37.7**	70
Nuclear	8081.9	92.3	**8462.7**	**96.6**	95.5
Wind	1778.7	20.3	**1872.3**	**21.3**	95
PV	984.1	11.2	**1000.4**	**11.5**	98
Hydro	2412.7	27.5	**6031.8**	**68.7**	40
Biomass	5264.6	60.1	**6729.0**	**76.8**	90

Table 1: FULL-LOAD HOURS for different types of power plants using the given ENERGY CARRIERS in Germany for the year 2015. The *corrected data* columns provide the full-load hours for available power plants, where available means not in planned or unplanned outage due to maintenance or other reasons. The percentage numbers ($\frac{full-load-hours}{yearinhours}$) are just a different representation of the FULL-LOAD HOURS. The data is taken from [26], availability factors are taken from [27]. Please notice that availability factors found in the literature differ.

sidered separately. An interesting measure for this are the FULL-LOAD HOURS of the different types of power plants. This measure, which describes the degree of utilization in conventional power plants, indicates how many GW nominal capacity are *used* to generate 1 GW h of electricity. As we can infer from Table 1, the full-load hours differ very much for the different energy carriers, and the described factor ranges from about 9 (PV) to about 1.1 (Nuclear). The high factor for VARIABLE RENEWABLE ENERGY SOURCE (VRE)-based power plants is a logical consequence of the dependence on wind and radiation. For conventional power plants, however, the factor is also high for some energy carriers, indicating a certain overcapacity. In general, there is much more capacity available than would be necessary with a constant load. The reasons for this are manifold. The two most important aspects are, on the one hand, the dynamic load which cannot be influenced, and, on the other hand, the problem that PV and wind power plants do not provide secured capacity and, therefore, have to be backed by conventional power plants.

The latter aspect is illustrated in Figure 10. It shows the distribution of electricity from VRE-based generation as a share of the total load. While on average 35 % of the load is covered by renewable energy, the generation falls below 10 % in about 0.67 % of the considered period, or 58 h per year. On the other side of the distribution, there are 10 hours a year in which more than 90 % of the load is covered by renewable energy.

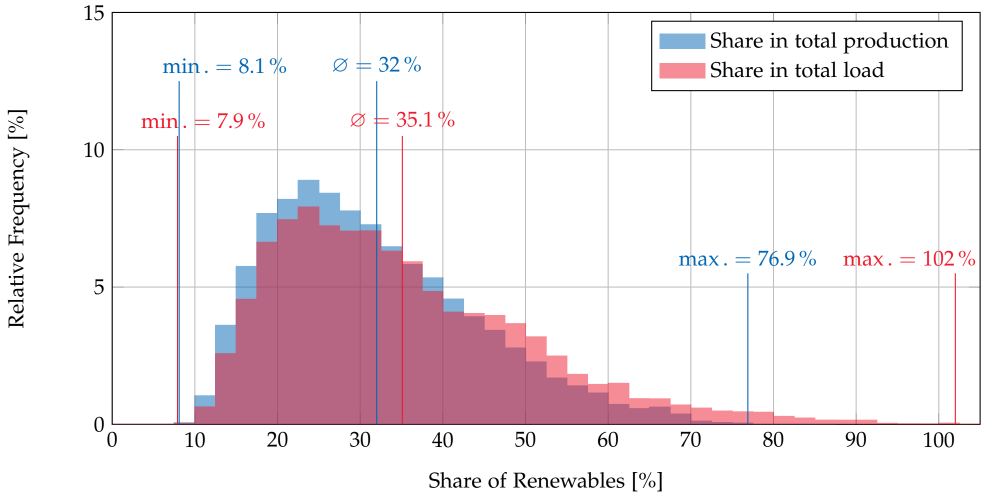


Figure 10: Relative frequency of the share of electricity from renewable sources for each quarter hour in the year 2016; data provided by the BUNDESNETZAGENTUR (GERMAN FEDERAL NETWORK AGENCY) (BNETZA), [28]

Due to the interconnected grid of continental Europe, taking the ratio between the German load and the German generation is not quite correct, since there are periods in which electricity is exported from the German grid as well as periods in which electricity is imported. Therefore, Figure 10 also shows the distribution as a share of the production. Once can observe that that share is lower on average, and it never rises above 80 %. Electricity from VRE-based power plants thus does not necessarily replace electricity from German conventional power plants, but it is also exported abroad.

The presented variable generation from VRES is, together with the price-inelastic demand and the MARGINAL COSTS of the different conventional power plants, the decisive factor for the pricing of electricity at the WHOLESALE MARKET. The distribution of the prices at the EPEX SPOT SE spot market for the German market zone is given in Figure 11. It is a cumulative representation which displays the portion of the hours of the year 2015 for which the price is above the respective x-axis value. The chosen representation allows to include the FULL-LOAD HOURS for the individual types of CONVENTIONAL POWER PLANTS, given in Table 1 (corrected values), as horizontal lines into the diagram and to derive a rough estimation of the lower price limits of the offers placed at the market as the crossing points of the introduced horizontal lines with the accumulated frequency curve. This estimation is based on the assumption that a CONVENTIONAL POWER PLANT produces energy only if the market price is above its MARGINAL COSTS. A comparison of these values with the respective marginal costs, provided as vertical lines, shows a mismatch, especially for lignite and gas. This means that lignite-fired power plants

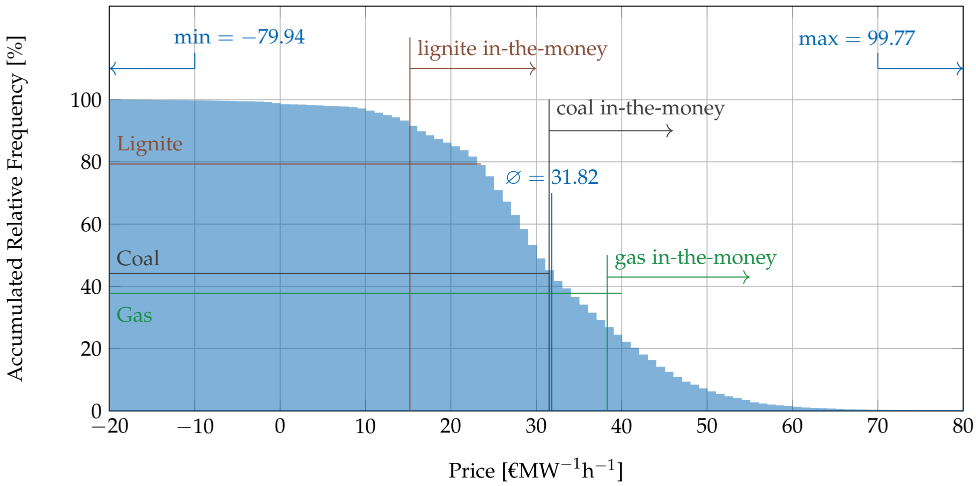


Figure 11: Distribution of the prices for electricity at the EUROPEAN POWER EXCHANGE SE (EPEX SPOT SE) day-ahead spot market for Germany and 2015, data is taken from German Federal Network Agency [28], data for MARGINAL COSTS (vertical lines) is taken from Energiewende [29].

produce less often than expected from the price distribution, while gas-fired power plants produce more often. In other words, gas-fired power plants produce electricity even when they are not *in-the-money*, i.e. the cost of the fuel is higher than the income from the electricity sold.

It thus becomes clear that it is not only the prices at the wholesale market that decide on the use of the power plants, but that there are other influencing factors as well. In the case of the gas-fired power plants, one reason is that many plants do not operate purely driven by the electricity market, as they are used for heat supply in district heating networks or deliver industrial customers directly.

In any case, it would be misleading to consider prices on the wholesale market alone, as these are only one, relatively small, price component of the cost of electricity from the consumers' perspective. Figure 12 therefore shows the end-user prices for electricity and the different price components. For our purpose, the exact numbers are irrelevant; we are mostly interested in the proportions of the categories. The decisive factor for the end-user prices is the high proportion of grid fees, taxes and apportionments, which are distributed differently to end customers of different types due to legal regulations. The high proportion of grid usage costs can be explained by the fact that in Germany the costs of operating and building the electricity grid are passed on solely to the users of electricity. The shown price components are static but some differ from place to place, or from grid provider to grid provider.

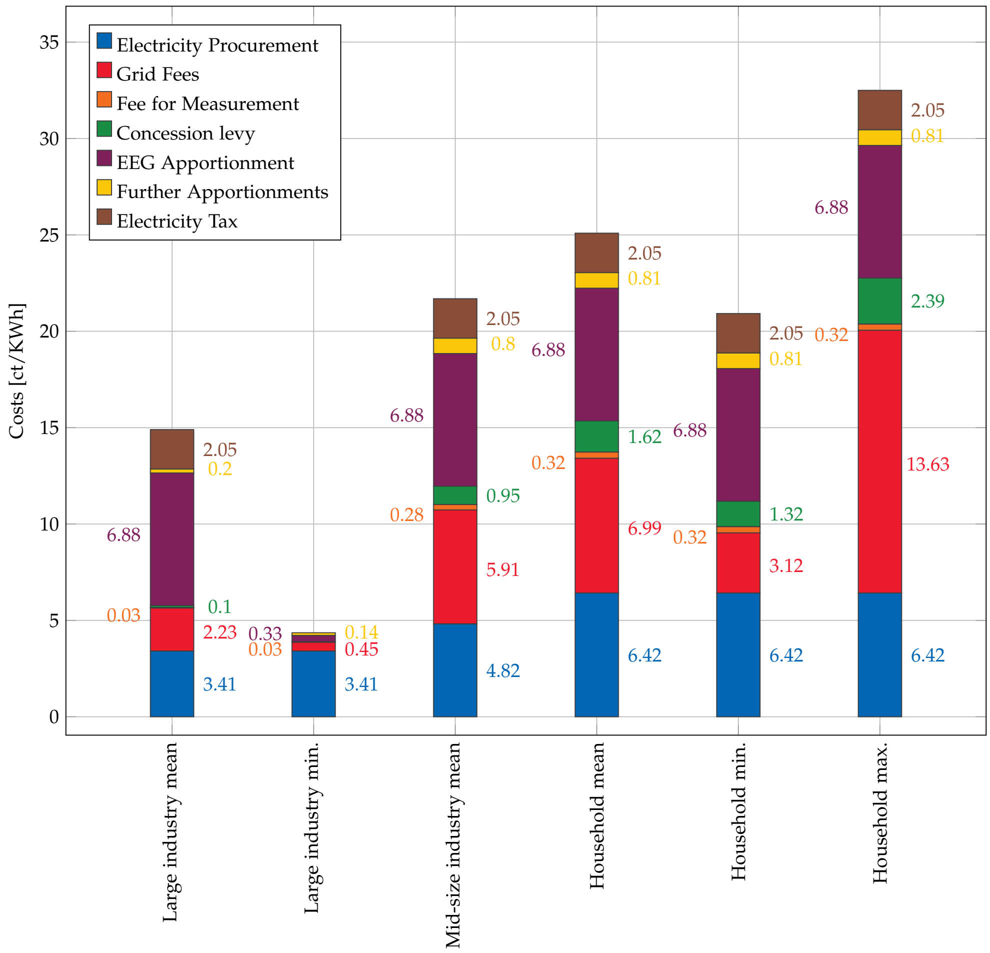


Figure 12: End-user prices for electricity in Germany, and their composition without VALUE ADDED TAX (VAT). Unless otherwise stated, the values are arithmetic means. *Large industry* refers to an end-user with a yearly demand of 24 GW h, *mid-size industry* to an end-user with a yearly demand of 50 MW h, *household* to an end-user with a demand between 2500 kW h and 5000 kW h. The values for *large industry min.* are theoretical values for an end-user who meets all conditions for a possible reduction of the respective fees, levies and apportionments. The values for *household min.* and *household max.* represent the maximum price fluctuations due to regional differences in prices for grid usage costs. The values shown here only occur very occasionally. Nevertheless, grid usage fees differ strongly from region to region. Concession levy differs from rural areas (low) to metropolitan area (high). Data is taken from [30] and refers to 2017.

2.2.3 *Different Perspectives of the Power System*

The POWER SYSTEM can be observed from different angles. We will briefly refer to two prominent perspectives, each of which has a different focus.

2.2.3.1 *System Actor Perspective*

A typical perspective of the POWER SYSTEM is based on the physical structure as given in Figure 13. As it focuses on the technical systems (SYSTEM ACTORS) directly connected to the grid, we will refer to this perspective as the SYSTEM ACTOR perspective. The most typical traditional actors are shown in the figure: conventional power plants connected to the high voltage grid, large and mid-size industries in the high and medium voltage levels and small consumers in the lower voltage level.

The core of this description is the hierarchical structure of the grid as given by the different VOLTAGE LEVELS and their tasks related to transportation and distribution. It is based on the traditional perspective that electrical energy is produced by conventional power plants connected to the TRANSMISSION GRID, transported by the TRANSMISSION GRID and the DISTRIBUTION GRID to the consumer where the electrical energy is utilized. As shown in Figure 13, the different VOLTAGE LEVELS are interconnected by substations transforming the voltage levels.

In a certain way, this picture also reflects the view that the primary task of the system is to meet an uninfluenceable demand for energy through large generation plants. In particular, the DISTRIBUTION GRID, is understood as a passive, static and mostly non-intelligent system.

Besides the above-mentioned traditional SYSTEM ACTORS, the figure also shows DESS, mainly utilizing VRES that are mostly connected to the grid at lower voltage levels. This high number of installations on lower levels leads to an increased reversal of the current flow direction in the transformers which interconnect the different VOLTAGE LEVELS.

2.2.3.2 *Business Actor Perspective*

The technical structure-driven description of the POWER SYSTEM lacks the crucial aspect that the time-dependent behavior of the system is not only affected by the SYSTEM ACTORS, but also by BUSINESS ACTORS that are *somehow* connected to the SYSTEM ACTORS and interact with other BUSINESS ACTORS within business processes and at markets. This aspect is especially essential for liberalized POWER SYSTEMS. Some of the ACTORS are not even indirectly connected to the grid, such as banks trading on ELECTRICITY MARKETS.

To view the POWER SYSTEM from a BUSINESS ACTOR perspective, the representation of the system as a value chain is suitable; Figure 14 provides a typical visualization. The value chain shown is based on the classic

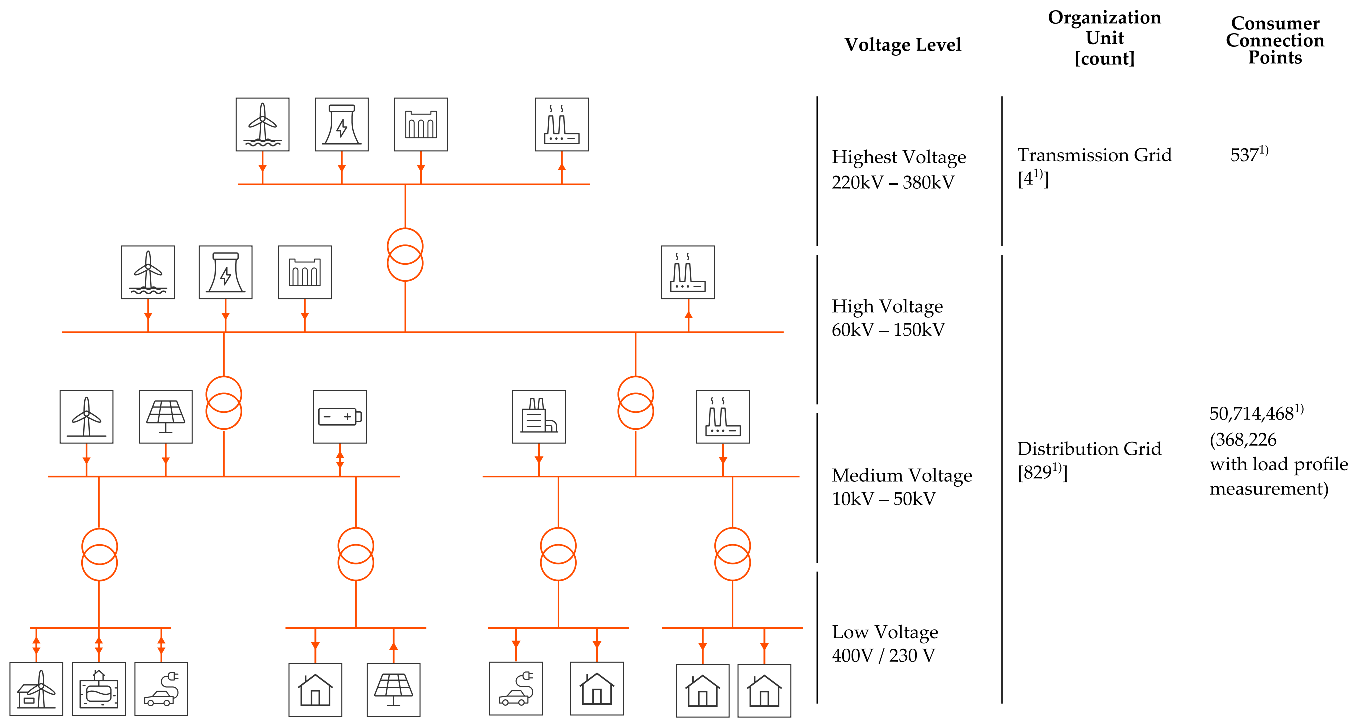


Figure 13: POWER SYSTEM from the SYSTEM ACTOR perspective. The given data is based on the following sources: 1)[30], [31].

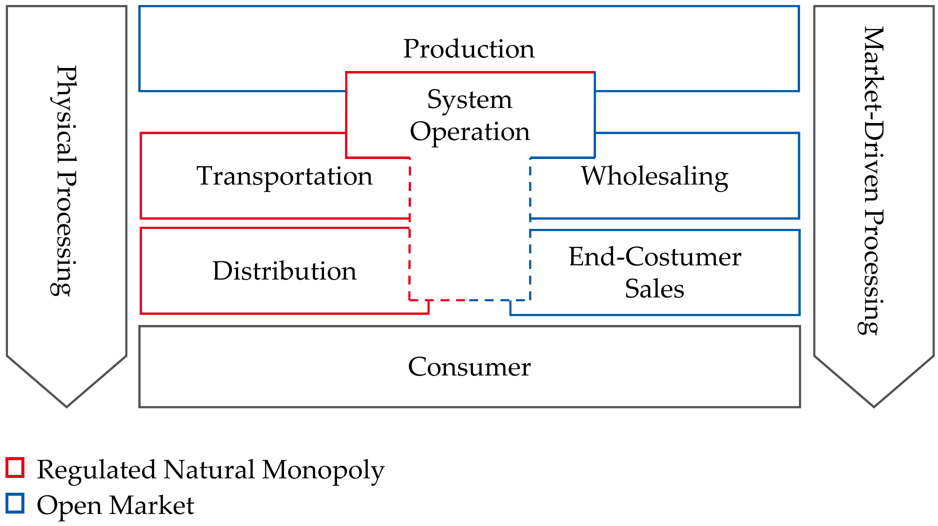


Figure 14: Value chain of the electricity power industry, adapted from Crastan [32] and modified.

view of a POWER SYSTEM as a process that starts with the generation of electricity and ends with consumption. On the one hand, the chain therefore shows the technically required activities, which are already addressed by the SYSTEM ACTOR perspective: generation, transport, distribution, and consumption. On the other hand, two additional activities are incorporated: wholesaling and the end-costumer sales. Those are the activities that take place in a completely liberalized systems within a free market. As the two activities transport and distribution are carried out by natural monopolists in liberalized systems, they must be regulated.

2.2.4 *Organizing the System*

The division of the overall system into an open market on the one hand and regulated monopolies on the other hand leads to a strong dependence of the players in the free market on the monopolists. For example, for the fulfillment of transactions, they depend on the monopolists providing the network infrastructure.

A graphical representation of the fully liberalized system is given in Table 2 in the rightmost column. It is important to note that, in a fully liberalized system, despite being able to choose his supplier freely, the end customer is typically bound to precisely one supplier over an extended period of time. For comparison, a monopoly is represented in the leftmost column. The center shows an intermediate stage between the two system versions. Although it has a wholesale market for electricity, there is only a single buyer, which in turn supplies energy to end customers.

The successful organization of such a liberalized system, which leads to competition as well as high security of supply, is based on the following cornerstones, according to Crastan [33, p. 110]:

A. A functional market for electricity (wholesale) producers. This requires a sufficiently high number of producers (horizontal unbundling), the avoidance of oligopolistic conditions and the prevention of cross-subsidies. To avoid any discrimination regarding the grid usage, an unbundling of production, transportation and distribution is required (vertical unbundling).

B. A functional market for the end-consumers of electrical energy. This means that consumers are free to choose their suppliers and have incentives to do so. It must, therefore, be possible to conduct commercial transactions between producers, intermediaries, and consumers.

C. The operation of the grid, including the responsibility for a stable operation of the grid, remains as a natural monopoly, which must be regulated by the government. Network operators are compensated for their services and must guarantee free grid access.

A system that follows the cornerstones mentioned above typically allows market participants to carry out trading transactions without considering the current or upcoming grid status. Accordingly, there is a separation between the two areas grid and market, and the respective value chains, which span the path between electricity production and consumption as shown in Figure 14.

In the authors' opinion, the activity *system operation* cannot be clearly assigned to one of the categories. While the responsibility is given to a monopolist or a group of regional monopolists, the services needed to fulfill the obligations, e.g., control reserve, are provided by market participants.

The dependence of trading operations on the existence of sufficient grid capacity and technical feasibility is abstracted by appropriate mechanisms, which are derived from the actual local[6] system configuration, fundamental technical restrictions, and political guidelines. These mechanisms, which limit all possible options for market players to the technically feasible ones and allow for implementation by network operators, are summarized under the term MARKET RULES. These define the different ROLES, their rights and obligations (cf. *responsibility*, 2.2.6) as well as the processes required to orchestrate the overall system and are the sum of all acts, ordinances, and provisions on a legal or contractual basis.

6 In this context, local refers to the national framework.

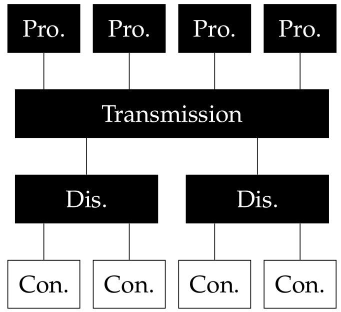
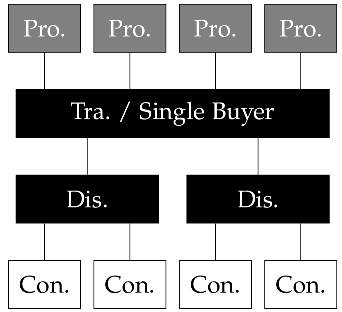
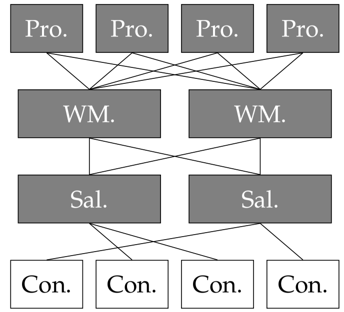
	Monopoly *monoplies at each level*	**Single Buyer** *market for producers*	**Fully Liberalized** *market at each level*
 **Structure**			
Type	unilateral	unilateral	multilateral
Market for Producers	no	yes	yes
Market for Resellers	no	no	yes
Market for End Customers	no	no	yes
Needed Level of Unbundling	no unbundling needed	horizontal unbundling at producer level	full unbundling

Table 2: Fully liberalized POWER SYSTEM compared with a monopoly, adapted from Crastan [32].

In the German context, the ENWG[7] defines the legal framework for the energy system. Its content is further specified in a large number of ordinances[8], such as the STROMNZV[9]. These regulations, in turn, are supplemented by provisions of the BNETZA[10]. It defines, for example, the balancing group contract which regulates the contractual relationships between BALANCE RESPONSIBLE PARTIES (BRPS) and TSOS, and related automated processes. These definitions are typically set up within the framework via a procedure involving the stakeholders concerned. Besides the ENWG, further acts are addressing different energy-related subjects, such as the EEG[11] and its related ordinances.

The German MARKET RULES lead to a BUSINESS ACTOR perspective, in which the geographical relationship between production and consumption has no effect on the advantage of trading transactions. The MARKET RULES thus seek to design the electricity system into a PERFECT MARKET for the market participants.

Within Germany, there is only one price zone, and there are no restrictions regarding transfer capacities, as they (again) exist at the borders with neighboring countries. This independence of the generation of electrical energy from the proximity to electrical loads does not only exist in trading transactions, but also in investment decisions for power generation plants. However, this does not imply that investment decisions are made independently of their geographical location. This applies in particular to power plants using VRE, as their supply heavily depends on the location. This design has been facilitated by a transmission grid with high transport capacities and a historically grown distribution of the power plants, where the local availability of fuels was accompanied by regions with high loads. A prominent example is the lignite-fired and coal-fired power plants near the load centers of the Ruhr region.

2.2.4.1 *Building Blocks of the Organization*

In addition to the organizational principles and the legal framework presented so far, there are a handful of technical building blocks that enable the implementation of the processes and the interaction of market participants.

BATCH-BASED PATTERN Basically, the orchestration of the BUSINESS ACTORS uses a batch-based pattern. This means that the time-variable be-

7 ENERGIEWIRTSCHAFTSGESETZ (GERMAN ENERGY INDUSTRY ACT)

8 The official website of the BUNDESMINISTERIUM FÜR WIRTSCHAFT UND ENERGIE (GERMAN FEDERAL MINISTRY FOR ECONOMIC AFFAIRS AND ENERGY) (BMWI) list 18 ordinances `https://www.bmwi.de/Redaktion/DE/Publikationen/Energie/gesetzeskarte.html` (last accessed: 3 June 2021).

9 STROMNETZZUGANGSVERORDNUNG (GERMAN ELECTRICITY GRID ACCESS ORDINANCE)

10 BUNDESNETZAGENTUR (GERMAN FEDERAL NETWORK AGENCY)

11 ERNEUERBARE-ENERGIEN-GESETZ (GERMAN RENEWABLE ENERGY SOURCES ACT)

havior of the actors for a longer period, typically one day, is coordinated at once and not for each individual TIME PERIOD.

FIXED TIME INTERVAL The shortest TIME INTERVAL of the TIME PERIOD used for the organization is 15 min[12]. The orchestration is therefore based on energy quantities. If power values are selected for description, they represent average power values for a given period of time and thus also describe energy quantities. In most cases, the behavior of an ACTOR within a quarter of an hour is not evaluated. This TIME INTERVAL is also known as IMBALANCE SETTLEMENT PERIOD. The beginning of the time periods are harmonized with the beginning of every 1/4 hour.

SCHEDULES The time-variable behavior is mapped using so-called SCHEDULES. A SCHEDULE describes the planned or actual behavior of an actor, and it is defined as "a reference set of values representing the generation, consumption or exchange of electricity for a given time period" ([34]). In its core, it is a time-series with fixed time steps. In addition, it holds a set of meta information, especially regarding network topological localization but also concerning TRANSACTION TIME, related business type, receiver and sender of the schedule, and further information. The point in time to which an individual element of the schedule refers is also called TIME OF DELIVERY. A SCHEDULE typically covers a TIME PERIOD with an interval of 24 h. From a market perspective, a schedule specifies a transaction between two market participants. Due to the central role, the format of the schedules is subject to (international) standardization. The format currently used in the German context, cf. [35], is based on the version 2.3 of the ENTSO-E SCHEDULING SYSTEM (ESS) [36]. Schedules are mainly used for the description of mean active power values and active energy quantities, but can also be used for other quantities such as reactive energy. In a slightly modified version, e.g., they are also used for weather forecasts.

LEAD TIMES The forth building block of the organization is the definition of LEAD TIMES which oblige market players to make their decisions regarding their behavior for a particular point in time up at the latest at an earlier point in time specified by the market rules. This point in time is also known as GATE CLOSURE TIME (GCT). The resulting LEAD TIME enables the evaluation of the actions from the point of view of the system operators and the utilization of conventional power plants. More precisely, the GCT defines the latest possible TRANSACTION TIME in relation to the TIME PERIOD covered by the SCHEDULE. Due to batch-based planning, the lead differs for the individual time of deliveries. In addition,

12 To the authors' best knowledge, there are no MARKET RULES using shorter TIME INTERVALS.

the lead time depends on the action of the BUSINESS ACTOR. Some activities, such as international energy trading, have longer lead times, while others, such as trading within a market zone, have shorter lead times.

For the description of TIME PERIODS in relation to a TIME OF DELIVERY t_0 at a day d_0, the terms *day-ahead*, *intraday*, and *day-after* have become established. Their end times define the respective GCTS. We will introduce the terms briefly, as they will be used later. In the German context, they are defined in [35]:

DAY-AHEAD previous month to $d_0 - 1\,\mathrm{d}$ at 14:30,

INTRADAY from $d_0 - 1\,\mathrm{d}$ at 18:00 to $t_0 - 15\,\mathrm{min}$ ... $120\,\mathrm{min}$[13], depending on the action of the BUSINESS ACTOR,

DAY-AFTER from $d_0 + 1\,\mathrm{d}$ at 00:00 to $d_0 + 1\,\mathrm{d}$ at 16:00.

The terms are also used with respect to different markets of the electricity exchange, where they describe slightly different periods of time, cf. 2.2.9.

2.2.5 *Nist Conceptual Model*

Surprisingly little scientific work focuses on a holistic description of the POWER SYSTEM and therefore we will draw on work developed by US and European institutes for standards. The central goal of the NIST was not the holistic description of the POWER SYSTEM but "[...] to coordinate development of a framework that includes protocols and model standards for information management to achieve interoperability of Smart Grid devices and systems [...]" [37]. Fortunately, this goal required the development of an abstract but holistic description of the POWER SYSTEM, which is known as NIST CONCEPTUAL MODEL. This model has been taken up and extended by the European standardization institutions. Some of these enhancements have been incorporated into the further development of the NIST model.

An overview to the NIST CONCEPTUAL MODEL is shown in Figure 15. It is published as part of NIST's "Framework and Roadmap for Smart Grid Interoperability Standards" whose first version [37] was released in 2010. The third version of the model [5] was published in 2014. This conceptual view of the POWER SYSTEM is somehow detached from the actual physical implementation of the system and even from the logical structure. At the core of this view is the question of the ACTORS, and *applications* needed to satisfy the demand for ENERGY SERVICES and ENERGY-RELATED SERVICES.

13 According to [35], *intraday* ends at d_0 24:00. From the view of the authors this is misleading, since different GCTS apply for the individual hours of the day. The definition has therefore been adjusted.

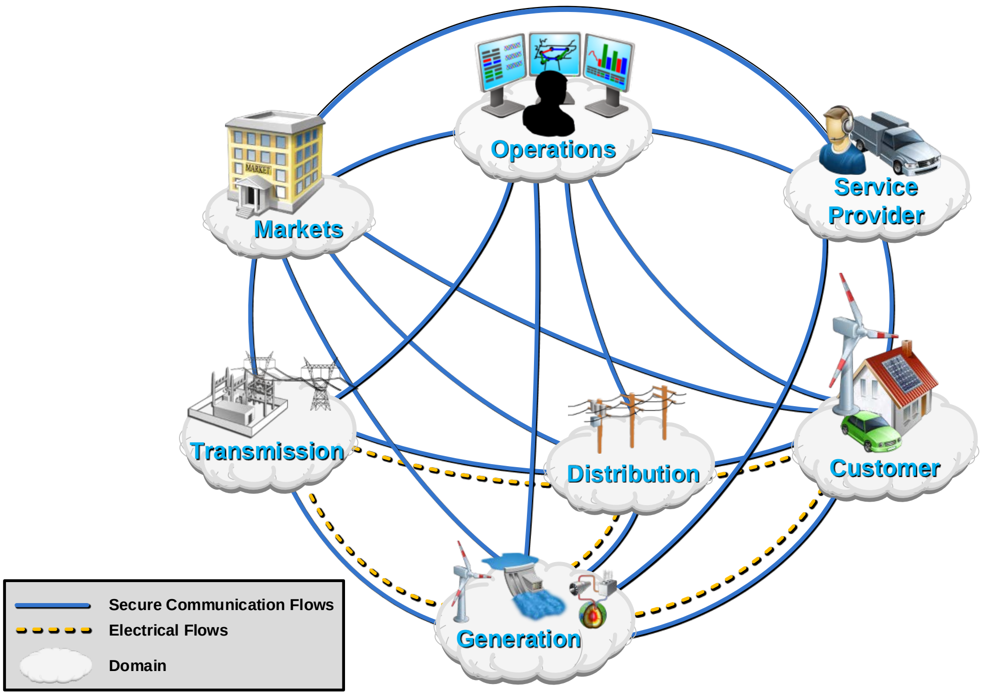


Figure 15: NIST NIST CONCEPTUAL MODEL, taken from [5].

However, as Figure 15 and Figure 16 indicate, this conceptual view is heavily inspired by the current design of the POWER SYSTEM.

The NIST divided the overall system into seven so-called CONCEPTUAL DOMAINS which are listed in Table 3. Following the formulation of the NIST, the domains encompass ACTORS and *applications*[14].

The different *domains* and the different ACTORS within one *domain* are connected by associations enabled through defined interfaces. These *associations* represent logical connections that are either electrical connections or communication connections. And it is obvious that only a certain part of the ACTORS have an electrical interface and thus a direct connection to the electrical network. In this context, the term ACTOR is used very abstractly. An ACTOR can be a person, a device, a software program or an organization. The decisive factor is the ability to make decisions and to interact with other ACTORS via an interface. The term *Applications* is also used in a very abstract manner. It refers to tasks that are performed by one or a collaboration of several ACTORS within a domain.

14 We decided to use the original definition from the first version [37] and not the updated version three [5], where the new version uses the terms ROLES and *Services* that have been added to the concept. The term ROLE seems to have been introduced here due to harmonization with the SGAM Framework. However, the descriptions and illustrations of the individual domains have only been adapted half-heartedly to the new terms. In the author's opinion, the original definition is more consistent, using the deliberately abstract term ACTOR, which covers everything that interacts with other ACTORS and the environment, rather than the term role, which describes a typical, expected and, in the context of the POWER SYSTEM, often defined behavior.

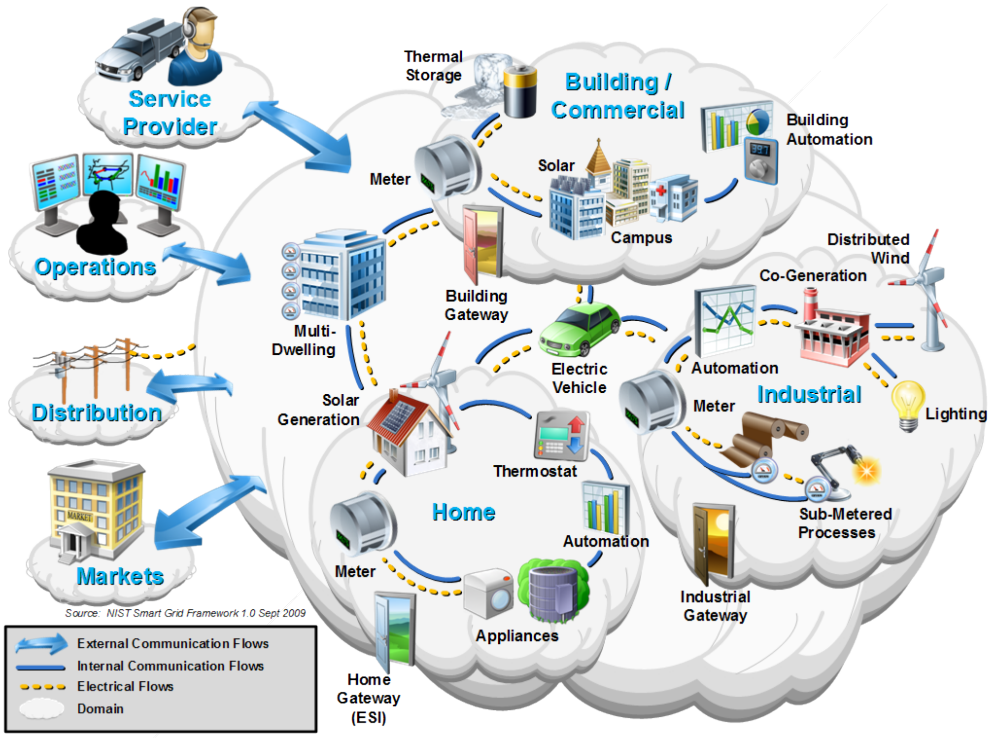


Figure 16: Customer domain of the NIST CONCEPTUAL MODEL, taken from [5].

The following paragraphs outline the different domains and are based on [37] and present a "near-term view" of the POWER SYSTEM.

CUSTOMERS The customer is the core of the POWER SYSTEM around whose need for ENERGY SERVICES the entire system has been built. As the Figure 16 shows, the domain has further sub-domains, which are related to the four DEMAND SECTORS, respectively to those electric sub-sectors. The domain also includes DES such as residential PV plants. The domain is connected to almost all other domains. Electrically to the distribution, communication-based to all domains expect the transmission domain. Two interfaces are given as the boundaries of the domain, the meter at the GRID CONNECTION POINT and an ENERGY SERVICE INTERFACE (ESI). While the first interface, even though its functionality has been very limited, is a standard for decades, the second interface is not yet common. It has the objective to enable an ACTOR-consumer interaction that includes a bridge to the customers' ENERGY MANAGEMENT SYSTEM (EMS). Typical, future-oriented applications, are home automation and residential storage systems.

MARKETS In the market domain, assets related to the POWER SYSTEM are traded. From a classical perspective, the primary asset traded on the market is active power, either as a standardized product at the electricity stock exchange or OVER THE COUNTER (OTC). Markets have a systemically

Domain	Description	Example Application
Customer	End-user of electricity. The domain is further divided into the sub-domains commercial, residential, and industrial. Also includes DES and local storage systems.	Micro-generation
Markets	Providers or participants in energy markets.	Trading
Service Providers	Providers of services to customers such as billing or installation, but also service not yet known.	Home Management
Operations	Managers enabling smooth operation of the POWER SYSTEM, includes application such as monitoring.	Network calculation
Generations	Generators of electricity. Includes conventional power plants, such as coal or nuclear, as well as DES. High overlap with costumer domain in the area of DES.	Protect against black and brownouts
Transmission	Carriers of electricity over long distances.	Measurement and control of grid state
Distribution	Distributors of electricity from and to the costumer.	Substation control

Table 3: Domains of the NIST CONCEPTUAL MODEL and their description, taken and slightly modified from [5].

important role in the electricity system; they balance supply and demand. A dysfunctional market leads to blackouts[15] a non-optimal market design to an inefficient matching of demand and supply. The domain is interlinked with all domains via communication flows, especially to the *Operation* and *Transmission* domain. Current markets are widely limited to active power and operating reserve; both traded at wholesale markets designed for bulk generation. Future markets will take more account of the large number of, mainly volatile, DES and the ACTORS of the customer domain. Changes are also expected for the tradable products, e.g., there will be products for FLEXIBILITY and the locality of the asset will be a more important aspect. Typical applications in the domain are market management and operation as well as retailing and aggregation.

15 Such as the famous California energy crises.

SERVICE PROVIDERS The service domain encompasses all ACTORS that provide services to ACTORS of other domains. [37] misses a definition of the term SERVICE, but the description of the domain suggests that it is understood as a functionality governed by an ACTOR provided through a well-defined interface. In the NIST CONCEPTUAL MODEL, the interface is restricted to the communication connections. Services may offer established standard functionalities such as billing and customer management. However, the service domain is understood as the domain in which new ACTORS may pop in to provide innovative services not yet known, but which are expected to make the electricity system more efficient. The key challenge named by [37] is to develop an ecosystem that enables new ACTORS to participate in the system easily but also secures the high standards of the POWER SYSTEM regarding safety, security and reliability. Applications, respectively *Services* in this domain are: account management and billing as well as building and Home management.

OPERATIONS The Operation domain comprises the ACTORS that are responsible for the operation of the power system. Typically, many applications in this domain are performed by a regulated utility, e.g., by the TSO having a natural monopoly. Applications are network monitoring, network control or fault management. The operation domain is heavily interlinked with the transmission, market and generation domain, but only via communication interfaces. It is expected that several of the applications are moving into the service or market domain, but that the core applications remain in the operation domain.

GENERATION ACTORS in the generation domain produce electrical energy. In the past, the domain was dominated by conventional bulk generation connected to the transmission grid. But, as the importance and number of DES have risen sharply, they now play an essential role. As a consequence, the domain is now also interlinked electrically to the domains customer and distribution, where most of the DES are located. Communication to the other domains is assumed to be critical. Mainly as it provides information on the variable sources wind and solar and allows the operation and market domain to handle insufficient supply.

TRANSMISSION The transmission domain encompasses all ACTORS enabling transport of energy over a long distance. It is electrically linked with the generation and distribution domain and on a communication-based linkage also with the operation and market domain. A transmission grid is typically operated by one TSO whose main task is to maintain the stability of the grid. The domain focuses on physical ACTORS such as substation meters and power quality monitors that allow to control and operate the transmission from the operation domain.

DISTRIBUTION The distribution domain links the customer domain with the transmission domain electrically. Historically, especially the lower voltage grid levels of the distribution grids were designed as a passive system mainly without measurement equipment and not equipped with systems capable to influence the state of the system. The enormous increase of DES connected to the distribution adds new requirements to the distribution domain. It is expected that ICT-based 'smart' systems enable cost-effective alternatives.

2.2.6 *EU Modification and Mapping to Role Model*

The NIST CONCEPTUAL MODEL was adopted and slightly modified by the three European standardization organizations EUROPEAN COMMITTEE FOR STANDARDIZATION (CEN), EUROPEAN COMMITTEE FOR ELECTROTECHNICAL STANDARDIZATION (CENELEC), and EUROPEAN TELECOMMUNICATIONS STANDARDS INSTITUTE (ETSI) within their reports fulfilling the Smart Grid Mandate M/490 [38]. This mandate was given to the three organizations by the European Commission in order to support the European SMART GRID development. The result is described in [39]. To distinguish the European model from the NIST model, we refer to it as the EUROPEAN CONCEPTUAL MODEL. For the European version, the NIST CONCEPTUAL MODEL has been harmonized with the ENTSO-E ROLE MODEL [40], which is a terminology that was defined by the ENTSO-E to enable a dialogue between market participants of different countries by defining atomic ROLES. This procedure was motivated by the need to describe business processes independently from the PARTIES and ACTORS that represent them, as different PARTIES assume different sets of roles in different national contexts.

The EU flexibility concept [41] was also incorporated into the design of the EUROPEAN CONCEPTUAL MODEL. The flexibility concept is also an output of the Smart Grid Mandate M/490 and addresses the challenge to manage distributed flexible supply and demand, and to integrate these elements into the electricity market and the framework for balancing supply and demand. FLEXIBILITY in this context is defined as "the changes in consumption/injection of electrical power from/to the power system from their current/normal patterns in response to certain signals, either voluntarily or mandatory." ([41])

Due to the alignment with the ENTSO-E ROLE MODEL, the EUROPEAN CONCEPTUAL MODEL groups so-called ROLES rather than ACTORS. The relation between the terms is given in Figure 17. As we will use these terms to describe the core processes of the POWER SYSTEM, a short introduction to the terms and their relations, adapted from [39], is provided in the following.

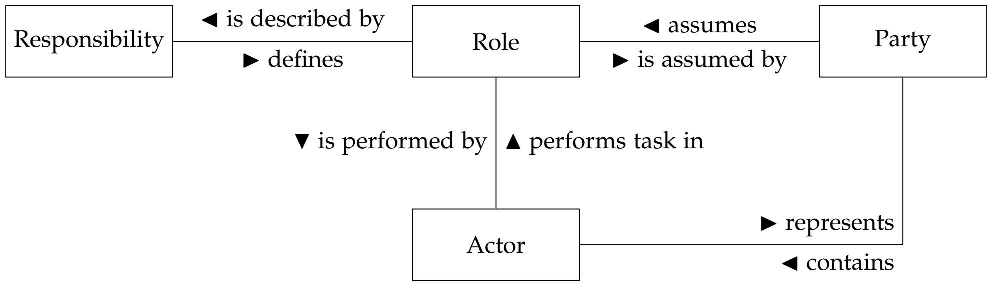


Figure 17: The terms ROLE, ACTOR, *responsibility*, and PARTY, and their relations according to [39].

A ROLE represents the intended external behavior of a PARTY. A ROLE cannot be shared between several parties and should be atomic. Examples are BRP, consumer or weather analyzer.

A PARTY is a legal entity, either a natural person or an organization. Examples are TENNET or Henri Oliveres.

AN ACTOR represents a PARTY. It performs tasks in one or typically a multitude of different ROLES. An ACTOR is an aggregation of a BUSINESS ACTOR and a SYSTEM ACTOR. We define a BUSINESS ACTOR as an ACTOR that participates in a business process, such as a customer, a trader or a trading system, whereas a SYSTEM ACTOR engages in a physical process, such as a transmission line or a HOUSEHOLD ENERGY MANAGEMENT SYSTEM (HEMS).

A RESPONSIBILITY defines the responsibility or external behavior of a ROLE. Examples are "control the grid frequency", "maintain the meters" or "consume energy".

The mapping of the ROLES defined by the ENTSO-E ROLE MODEL to the domains of the EUROPEAN CONCEPTUAL MODEL is given in Figure 18. The domains are grouped slightly differently than in the NIST CONCEPTUAL MODEL. Transmission and distribution are not represented as dedicated domains, as the dominating part of the NIST domains transmission and distribution is only one ACTOR, the TSO respectively the DISTRIBUTION SYSTEM OPERATOR (DSO). The domain grid users subsumes the NIST domains generation and customer.

The domains markets and energy services are almost congruent with the respective NIST domains. However, both are further subdivided into sub-domains related to the three aspects energy market, grid capacity, and flexibility market. Contrary to all other ROLES, the ROLES within the domains operation and grid users are linked with physical assets of the POWER SYSTEM.

2.2.7 *Core Roles and Domains*

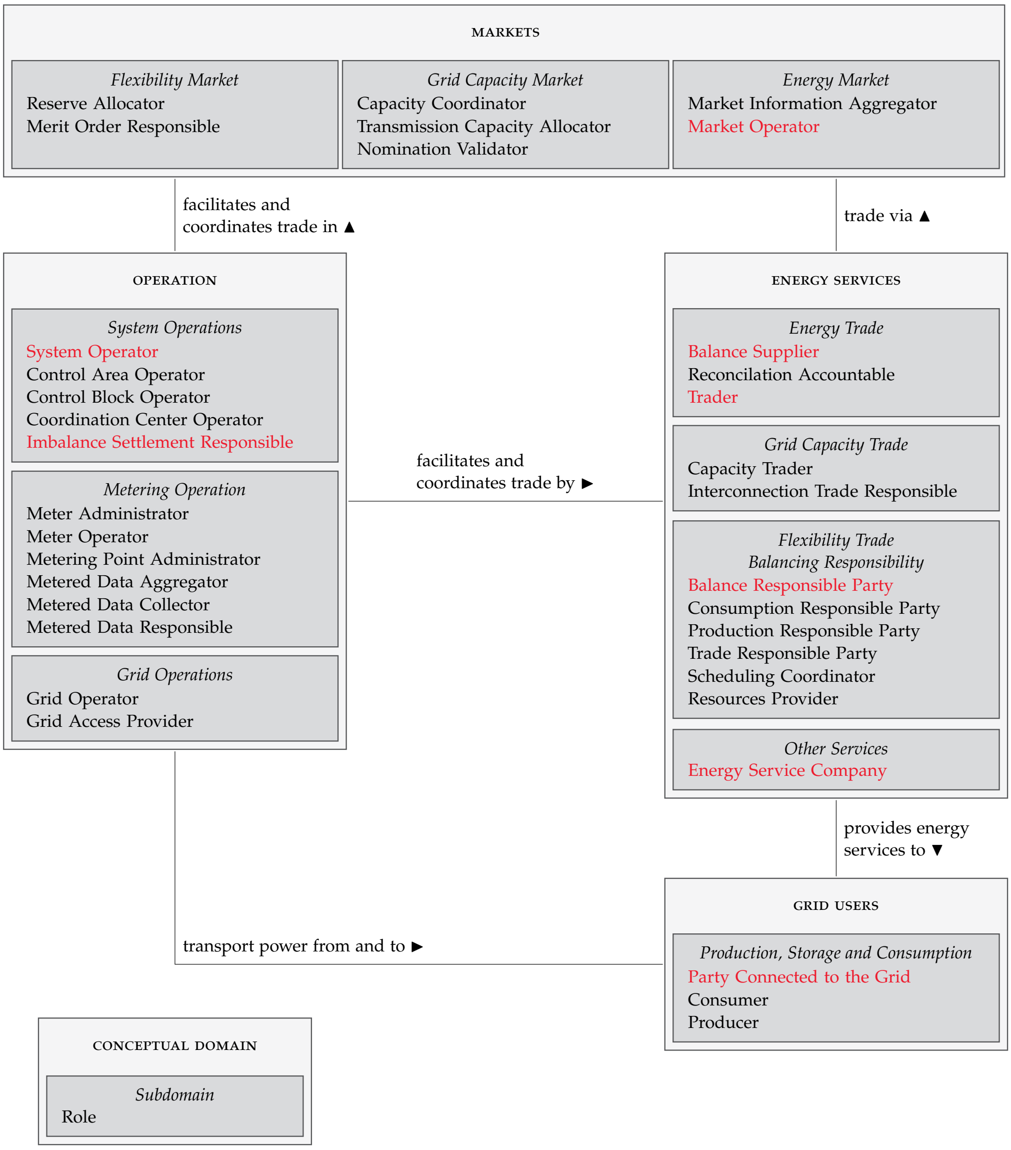


Figure 18: Overview of the domains of the EUROPEAN CONCEPTUAL MODEL and the mapping of the ROLES of the ENTSO-E ROLE MODEL to the respective domains. ROLES given in red are discussed in detail. The figure is based on [39].

The goal of MOCES is to model and simulate the interactions between the different actors in their roles within processes. In the following, we will therefore introduce the crucial roles involved in the most important process, the BMP, cf. Section 2.2.8. Figure 18 highlights these roles in red. We describe them based on the definition of the ENTSO-E ROLE MODEL [42] and establish a reference to the German market design based on the information given by the BDEW in [43]. In addition to the core roles, we will also describe the core domains. In the context of the ENTSO-E

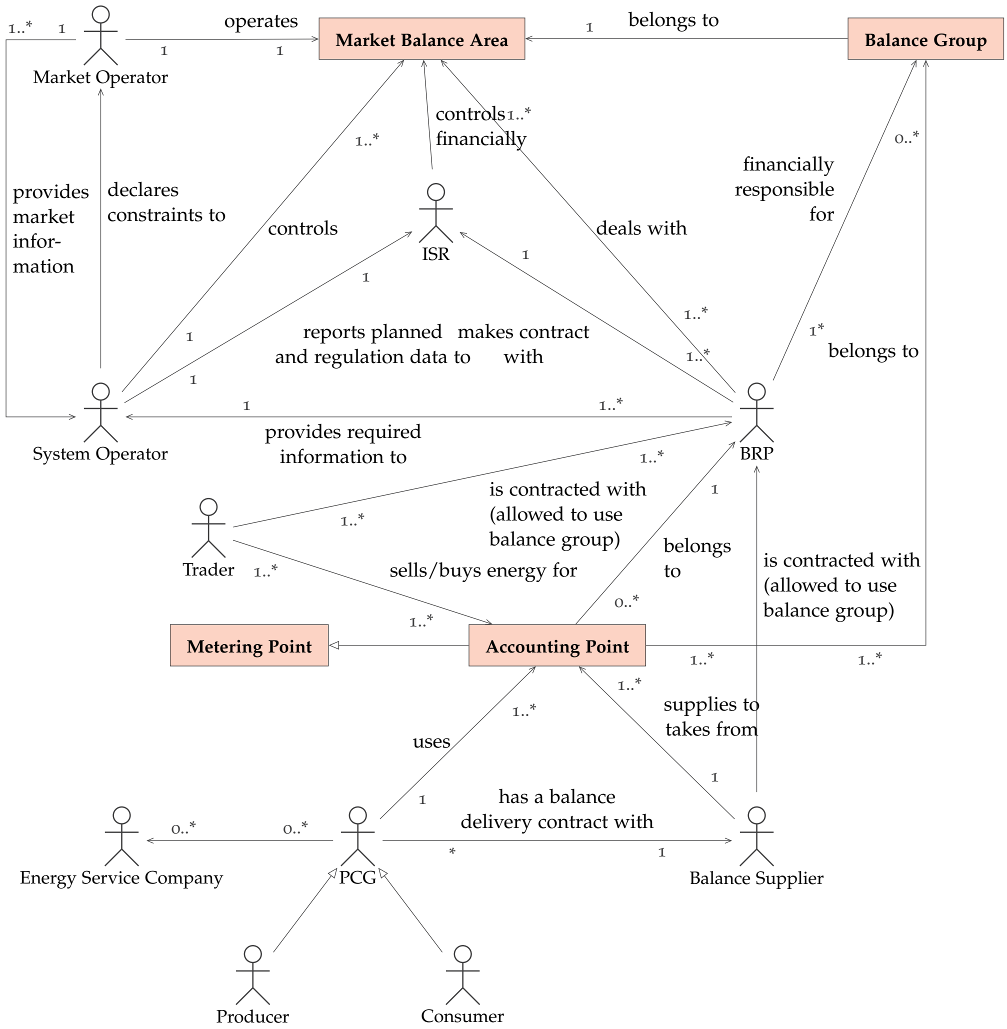


Figure 19: Condensed version of the ENTSO-E ROLE MODEL given in [42], visualizing the relations between core roles (stick figures) and domains (boxed).

ROLE MODEL, a domain is an "abstract objects used [...] necessary for the management of various processes, resources or areas." [42]. The relation between the core roles and domains is visualized in Figure 19.

2.2.7.1 *Core Roles*

PARTY CONNECTED TO THE GRID (PCG) : A PARTY that consumes or supplies electricity at an *Accounting Point*. Typically it has the type of a *Producer* or *Consumer*. It has a relation with exactly one *Balance Supplier*, the form of which is defined with a contract. Even a residential consumer is a PCG. Here, the contract typically specifies consumption-independent and fixed prices per energy quantity. In addition, relations to a multitude of different *Energy Service Companies* are possible.[16]

BALANCE SUPPLIER: "A party that markets the difference between actual metered energy consumption and the energy bought with firm energy contracts by the PCG." [42]. A typical supplier is a vendor who has a supply contract with an end customer. It is contracted with a BRP whose *Balance Group* is assigned the energy consumed at the *Accounting point*. In the German context, the roles BALANCE SUPPLIER and BRP are mostly assumed by the same party.

BALANCE RESPONSIBLE PARTY (BRP): A PARTY that is responsible for balancing its BALANCE GROUPS. It is therefore responsible for ensuring that the feed-in and consumption of all *Accounting points* assigned to the BALANCE GROUPS sum up to zero at each point in time. The BRP has a central role in the energy market as it is responsible for forecasting the demand of all *Consumers* and the feed-in of all *Producers* assigned to its *Balance Group*. It is the only role that "allows a PARTY to nominate energy at the wholesale market" ([42]). Thus, if a customer is to be supplied with electricity, PARTIES with the roles BRP, BALANCE SUPPLIER and TRADER are typically involved. Usually, PARTIES, therefore, assume all three roles. Transactions, and accordingly trades, are possible between BALANCE GROUPS within the same MARKET BALANCE AREA as well as between BALANCE GROUPS belonging to different market areas. However, network physical conditions can restrict these transactions.

TRADER: A PARTY selling or buying energy, mostly at the wholesale level. It is contracted with a BRP that allows the trader to use its BALANCE GROUP. Similar to the role BALANCE SUPPLIER, the role *Trader* is typically performed by one ACTOR together with the roles BRP and BALANCE SUPPLIER.

IMBALANCE SETTLEMENT RESPONSIBLE: The PARTY responsible for the settlement of the difference between planed and actual quantities of the BRP within a *Market Balance Area*. The *Imbalance Settlement Responsible* is not responsible for invoicing, but the same PARTY may perform the responsibility for invoicing.

SYSTEM OPERATOR: The PARTY that is responsible for a stable POWER SYSTEM. That includes the organization of the physical balance in

16 The n:1 relation between the PARTY CONNECTED TO THE GRID and the BALANCE SUPPLIER is expected to develop into an n:n relation in the near future. This development can for example be observed in the resolution BK-17-046 [44] of the German Federal Network Agency. The corresponding decision introduces the role *Aggregator* as an additional party directly linked to the PARTY CONNECTED TO THE GRID and its energy consumption. The *Aggregator* can be understood as a special form of a *Balance Supplier*.

cooperation with the BRPS. It is the central role in the BMP as it has to assess the expected state of the grid based on the information given by the BRPS and knowledge of the grid structure. If required, it intervenes in the behavior of loads and generators, cf. REDISPATCH.

MARKET OPERATOR: A PARTY that operates a power exchange for all PARTIES allowed to place bids and offers at the ELECTRICITY MARKET, mainly the BRPS, and determines the prices resulting from bids and offers. Usually, there is only one *Market Operator* for a *Market Balance Area*.

ENERGY SERVICE COMPANY: A PARTY that provides energy-related SERVICES. In the definition given by the ENTSO-E ROLE MODEL, the user group of the services is restricted to PCGS. This restriction, however, contradicts the basic idea of SERVICES. In reference to the SERVICES of a CONCEPTUAL DOMAIN, cf. Section 2.2.5, we understand a *Service Company* as a PARTY that provides a SERVICE through a well-defined interface to any other ACTORS to support those activities. *Energy Service Companies* are expected to adopt ROLES in the future that are currently not yet known.

2.2.7.2 *Core Domains*

ACCOUNTING POINT: An accounting point for energy under the financial responsibility of precisely one BRP and belonging to a *Balance Supplier* and a PCG. The *Accounting Point* is a particular type of a *Metering Point* that represents the asset meter that measures or calculates energy flows. The *Accounting Points* can be interpreted as the inputs and outputs of the *Balance Group*.

BALANCE GROUP: An account under the responsibility of one BRP; it belongs to exactly one *Market Balance Area*. It sums the power flows of the linked *Accounting Points*. Apart from the relation to the *Accounting Points*, that have a dedicated physical placement, and the assignment to a *Market Balance Area*, BALANCE GROUPS abstract away from the grid topology.

MARKET BALANCE AREA: A geographic area with common market rules and a single price for imbalances. In addition, the ability to meter the feed-in, consumption, and exchange with other areas is required. *Market Balance Areas* typically span larger regions. In Germany, for example, there are four zones, which are congruent with the territories of the four TSOS. In most cases, these four areas behave like a single area.

2.2.8 *Core Processes: the Balance Management Process*

The respective local market rules include a large number of defined processes for organizing the POWER SYSTEM. In the German context, the details of these processes are determined by decisions of the Decision Chamber 6 of the Federal Network Agency. The decisions are results of procedures involving the actors concerned.

An overview of the processes of the German system can be found in a policy paper of the BDEW, published in the context of further developing the existing processes [45, p. 29]. The majority of those processes address "real-world problems" arising from the heterogeneity of the ACTORS involved and their IT-systems. However, many of these processes are not expected to have any significant impact on the efficiency of the entire system. Examples are the concrete design of business processes for supplying end customers with energy, and associated processes such as cancellations or supplier changes, both defined by the GPKE[17]. Other processes in turn describe the procedure regarding the exchange of power generation data, e.g., the MPES[18], which concretely specifies which quantities must be measured in which resolution and to which ROLES these measured values must be made available in which way. This process is an example of solving problems that indeed occur within a real SYSTEM. However, the problems would not occur in a model of the system that is used to analyze the system, as the provisioning of the required and correctly measured values in a defined format to the ACTORS can be guaranteed within a simulation without any problems.

Accordingly, and with reference to the focus of the modeling framework MOCES (cf. Section 1.3.1), we will restrict ourselves to the description of those processes and their design which have a concrete impact on the performance of the POWER SYSTEM. These are all the sub-processes that are directly linked to a process called BMP, whose design in turn is linked with the design of the electricity markets.

The BALANCE MANAGEMENT PROCESS (BMP) is the core process that enables a liberalized energy market and thus an open market. At the same time, its design must enable a reliable power supply. Although the national arrangements differ in detail, the process is always based on the fundamental idea that each individual market participant is obliged to balance its BALANCE GROUP at all times which results in a balanced system. *At all times* normally refers to the amount of energy within time slices of 15 min length. Deviations are punished financially, with penalties calculated in such a way that it is unfavorable for the ACTOR to ac-

17 GESCHÄFTSPROZESSE ZUR KUNDENBELIEFERUNG MIT ELEKTRIZITÄT (BUSINESS PROCESS FOR SUPPLYING CUSTOMERS WITH ELECTRICITY)

18 MARKTPROZESSE FÜR ERZEUGENDE MARKTLOKATIONEN (STROM) (BUSINESS PROCESS FOR ACCOUNTING POINTS PRODUCING ELECTRICAL ENERGY)

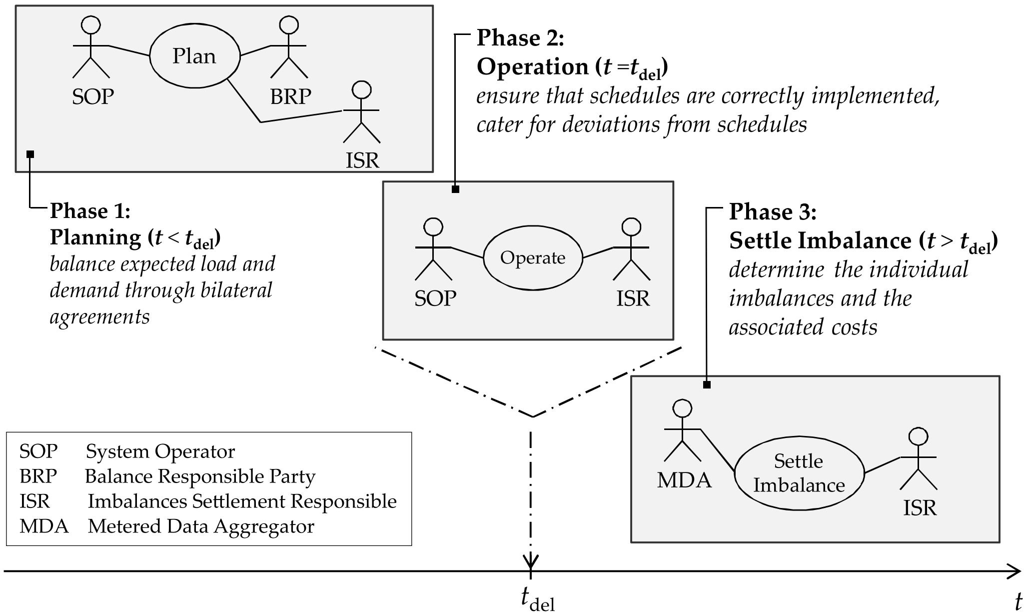


Figure 20: Main phases of the BMP as use case diagrams. Figure is based on [46].

cept this punishment instead of ensuring a quarter hour performance balance by other means. Using the batch-based process pattern, cf. Section 2.2.4.1, the process steps within the BMP typically cover longer TIME PERIODS; usually, the processed TIME PERIOD starts at 00:00 each day and covers 24 h.

The overall context of the BMP is given by the use case diagrams shown in Figure 20. In the following, we will give a detailed description of the three sub-processes: planning, operation and settle imbalance. The description is based on the respective documents developed by the ENTSO-E, [46], [47] and [48], and the particular German configuration, described in [35], [49], and [50]. These national configurations are based on the legislative regulations given by STROMNZV.

Basically, all three use cases shown in Figure 20 run continuously and in parallel. In relation to a specific TIME OF DELIVERY t_{del}, however, the three consecutive phases are correlated. The planning phase ends at the latest at the TIME OF DELIVERY, the operation takes place at exactly this point in time, and the settlement phase begins at the earliest after this point in time.

Section 2.2.4 already discussed that the design of the MARKET RULES has an impact on the performance of the overall system. After having introduced the BMP, we will now give some concrete examples for that. Weißbach [51] shows that the typical one-hour trading products with constant output lead to an increased demand for balancing power. Borggrefe and Neuhoff [52] underline the importance of intraday markets with short lead times to optimally integrate wind energy into the overall system. A decisive aspect there is the negative correlation of the forecast quality with the forecast horizon. At the same time, however, Borggrefe and Neuhoff [52] point out that shortened lead times can also lead to problems, since only a few conventional generation units are technically capable of reacting sufficiently quickly to the desired performance changes. Markets with short lead times therefore exclude sluggish generation units and their flexibility in terms of power generation.

Additionally, when new concepts in the area of LOCAL ENERGY SYSTEM and DEMAND RESPONSE are developed, they must be harmonized and compatible with the BMP and the essential characteristics of an liberalized energy market. However, this condition is often neglected when developing and presenting new concepts. For example, in the study by Eid et al. [53], four concepts are introduced, but none of them is directly compatible with the BMP. For three of the presented concepts, the authors at least assume that they are fundamentally compatible with the basic principles of the liberalized energy system. Nevertheless, they would require the definition of new roles and the modification of existing roles.

2.2.8.1 *Planning*

In the *Planning* Phase, the expected energy production and consumption are planned by means of quarter-hour SCHEDULES. This phase refers to interactions on the markets, but also to the network-physical assessment of the planned production by the SYSTEM OPERATORS (SOPS), which are responsible for the stability of the grid. The covered TIME PERIODS are *Day-Ahead* and *Intraday*.

In the following description, we will restrict ourselves to the interactions of the BRP with the SOP, assuming only one MARKET BALANCE AREA. The processes required to conduct trading transactions across MARKET BALANCE AREAS and their transfer capacity restrictions are not described here, as we will not model them in the later applications of MOCES in Chapter 4 and 5. For the same reason, we also omit a description of the processes related to transmission capacity rights.

A detailed, yet simplified representation of the planning phase from the BRP perspective is given in Figure 21. It is a based on the overview diagram given in [46] and is extended by the author to underline aspects not in the scope of [46]; these are those aspects that relate to the internal activities of the BRP. We further assume the typical configuration where one ACTOR performs the ROLES BRP, BALANCE SUPPLIER, and *Trader*.

The core of the process in Figure 21 is the transmission of the schedules from the BRP to the SOP that has to be done before GCT of the TIME PERIOD *Day-Ahead*, cf. upper part of the blue fragment in Figure 21. Before the SOP accepts the SCHEDULES, it checks their validity concerning three aspects: their inner coherence independently from other SCHEDULES; the existence of a complementary SCHEDULE matching this

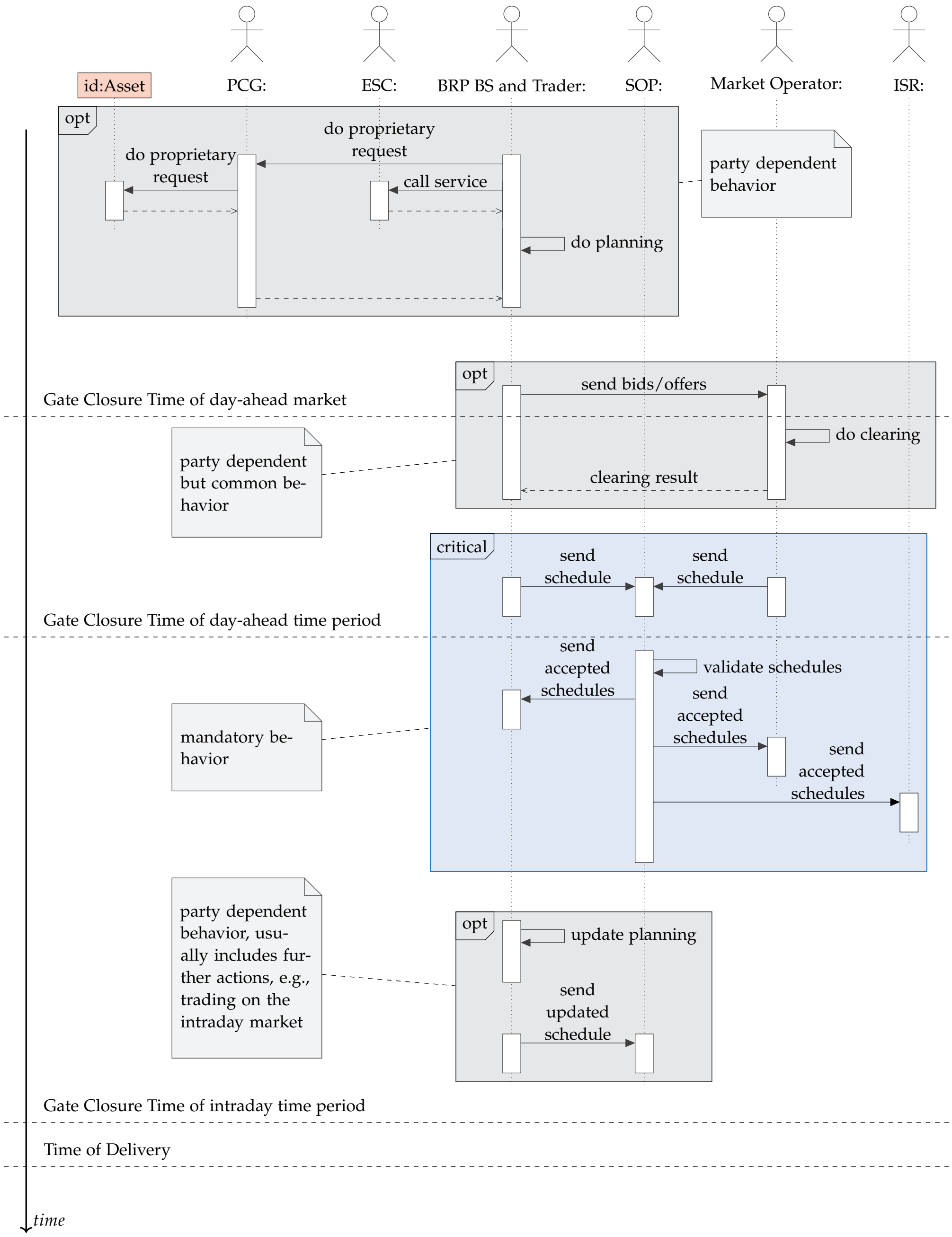


Figure 21: Sequence diagram of the planning phase of the BMP. The diagram is based on [46] and extended in relation to the actor-dependent behavior given in the two upper fragments.

SCHEDULE; and the compatibility of the totality of all SCHEDULES with the MARKET RULES. The transmission of all SCHEDULES to the SOP enables the network-physical assessment of the totality of all actions of the BRPS by the SOP. This step is essential as the BRPS are not capable of including grid transfer restrictions into their decisions. However, if the SOP finds that the totality of all transactions may lead to an unacceptable network condition, this conclusion does not necessarily lead to the rejection of SCHEDULES. The SCHEDULES can still be confirmed, and the SOP will carry out actions in a separate process to counteract grid congestion[19].

As indicated in the sequence diagram (Figure 21), the BRPS are allowed to update their schedules after the GCT of the *Day-Ahead* period within the *Intraday* period, cf. bottom gray fragment. This opportunity for updates is particularly significant for VRES, as it allows to update the schedules based on improved feed-in forecasts, whose quality-level increases with shorter time spans between TRANSACTION TIME and TIME OF DELIVERY.

The party-dependent activities of a BRP, given in the gray fragments of the sequence diagram, differ depending on the strategy of the BRP and the type of the PCGS associated with it. The activities in the figure are prototypical but not necessary, and they can vary from PARTY to PARTY. If the associated PCGS are of the type of a VRE-based producer, the most important activity is the elaboration of a feed-in forecast for all PCGS associated to the BALANCE GROUP. For this purpose, the BRP typically communicates with one or more service providers who provide weather or feed-in forecasts. These forecasts form the basis for the bids on the day-ahead market of the electricity market. If the assets are not controllable, the BRP is forced to place "sell at any price" bids on the trading exchange. This obligation arises from the due diligence of the BRP for an equalized BALANCE GROUP. The BRP thus has to accept the case of paying money for selling electrical energy; according to [30], in 2016, 62.3% of all offers at the EPEX SPOT SE day-ahead market were price-insensitive. The LEAD TIME of the day-ahead market is up to 36 h. Typically, the BRP therefore uses the continuous intraday market, that has much shorter LEAD TIMES, to react to changing feed-in forecasts. Energy is procured when the original forecast for the day-ahead market was too high, and energy is sold when the forecast was too low. All in all, for PCGS of the type VRE-based producer, the BRP has relatively little scope of action. The actions are mainly a reaction to the forecasts of the uninfluenceable behavior of its associated feed-in systems.

The options for actions are often also similarly limited if the associated PCGS are of type consumer. In the typical constellation where the connected assets cannot be controlled and the consumer is contracted with a fixed-price tariff, the BRP can neither intervene in the load nor is

19 See the process defined by the BNETZA for REDISPATCH, cf. BK-11-098 [54].

it able to provide financial incentives for changing behavior. In this case, the planning activities are therefore limited to the best possible forecasting of the loads and cost-optimal procurement of energy to equalize the BALANCE GROUP. Ultimately, however, the BRP must accept any price in order to meet the demand for electrical energy.

Thus, the interaction of the BRP with the PCG, or associated asset, depicted in the upper gray fragment of Figure 21 often does not occur in reality. However, if it does take place during the daily planning procedure, it can improve the behavior of the BRP. For example, if the PCG is of the type of a VRE-based production unit, data exchanges of the actual feed-in values of the previous day can positively influence the quality of the forecasts, and thus ultimately also the yield of the marketing at the power exchanges. If the PCG is of the type of a CONVENTIONAL POWER PLANT, and, thus, it is possible to control its power output, the main task of the BRP is to determine optimal bids to be placed at the day-ahead market. Finding an optimal decision then typically requires the application of optimization problems or similar methods, especially if technical restrictions, such as ramping limits, add constraints to the bidding strategy. An approach addressing a set of PCGS that are either conventional power plants or hydro storage units is presented by Nowak et al. [55]. This contribution also shows that the problems to be solved are similar to the traditional UNIT COMMITMENT PROBLEM (UCP), cf. Section 3.2.3.1. The complexity of the task to be handled by the BRP is reduced by the concrete design of the markets and the products tradable there. This applies in particular to CONVENTIONAL POWER PLANTS and their technical restrictions, which are reflected in order types such as block orders and linked blocks, cf. [56] and Section 2.2.9.

2.2.8.2 *Operation*

In the *Operation Phase*, the BRPS have to ensure that their SCHEDULES are correctly implemented and the SOP takes care of unavoidable deviations between production and consumption. A schematic description of this phase, taking place at TIME OF DELIVERY, is given in Figure 22.

The extent to which the BRP can actually intervene in the load or production behavior of its associated PCGS heavily depends on the type of the PCG. It lies between the two following extremes: Small 'loads', such as a PCG of the type household, usually do not provide any type of interface that would enable the BRP to affect their behavior. In fact, even knowledge about the actual behavior is very limited for this setup: typically, it is nothing more than the annual consumption and the category of the load, such as household or office building. The other extreme is a PCG of the type of a controllable generator or load that is capable of following a dynamic trajectory provided by the BRP in REAL-TIME. Typ-

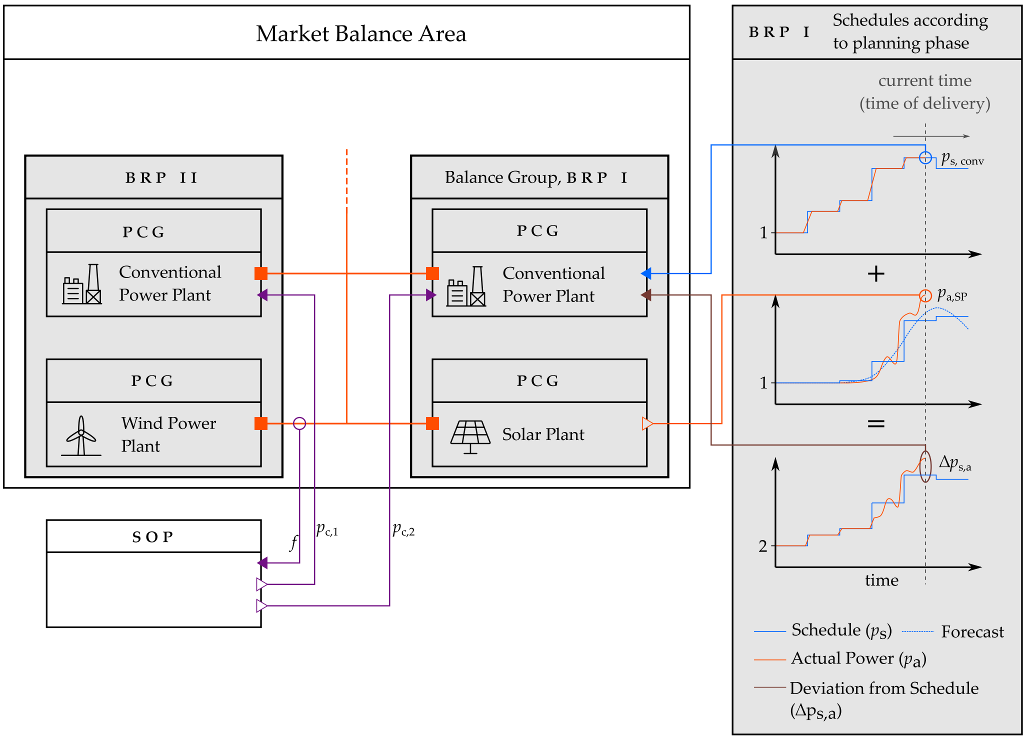


Figure 22: Schematic overview of the operation phase of the BMP, emphasizing the connection to the SCHEDULES which are the results of the planning phase, and the mechanisms controlled by the SOP to ensure the grid balance by activating control power.

ically, measured values of the actual behavior are also available in this setup.

The BRP I in Figure 22 shows another setup, for the BALANCE GROUP of a BRP. In this configuration, one part of the PCGS, represented by the CONVENTIONAL POWER PLANT, can be controlled, another part, represented by the solar plant, makes its actual behavior available to the BRP. These characteristics are represented by the signal flows. Since the BRP is responsible for the sum of the deviations of all its associated PCGS –represented by the bottom diagram on the right side–, and not for each individual PCG, the BRP can levelize its entire BALANCE GROUP by controlling only a subgroup of all its assets. In addition, the length of the IMBALANCE SETTLEMENT PERIOD of 15 min facilitates to recoup the balance group, since intra-quarter-hour deviations are not taken into account. In Figure 22, this possibility is represented by the additional set-point ($\Delta p_{s,a}$) for the CONVENTIONAL POWER PLANT, which is derived from the difference between the nominated SCHEDULES and the actual values .

Deviations from the planned SCHEDULES of the individual PCGS can offset each other within a BALANCE GROUP or across all PCGS connected to the grid. However, deviations are inevitable due to erroneous load and production forecasts and unplanned power plant outages, and they are also systematic due to 1-h-schedules with constant power traded on market places which do not fit the ramp-like load curve of all consumer; see Weissbach and Welfonder [57] or Hirth and Ziegenhagen [58] for a detailed discussion.

To level these deviations, the SOP is obliged to allocate controllable idle power capacity in reserve, known as OPERATING RESERVE, and to implement a balancing system that compensates the deviations continuously by utilizing the reserves. In Germany, the reserved capacity is about 6% of the annual peak load.

The technical design of the system that maintains the balance is also sketched in Figure 22, cf. the SOP and its connections (in purple). It is based on the advantageous characteristic of an AMPLIFYING CURRENT (AC) system with synchronous generators in which the grid frequency, cf. f in Figure 22, is a direct measure of the balance of production and consumption and can be measured at any point of the grid.

The rotating masses of the generators, which rotate in a fixed ratio with the grid frequency, thus represent inertial masses which are decelerated when the load is larger than the generation and accelerated when the generation is larger than the load. In simplified terms, the balancing system is a closed control loop in which the nominal frequency is the set-point value and the actual grid frequency f is the process variable. The control variable determined centrally by the SOP is distributed to the individual reserved capacities acting as actuators ($p_{c,1}$, $p_{c,2}$). To provide the required power as effectively as possible, different classes of available power have been defined that complement each other: *primary*, *secondary* and *tertiary control power*. While *primary control power* must be available immediately, the accepted latency for the others is significantly higher, in the range of seconds to minutes. The *tertiary control reserve* must be available over a longer period, in case of doubt permanently.

In the liberalized energy market, the SOP is obliged to procure the needed reserve from a market, which is known as *control reserve market*, where related products such as *primary*, *secondary* and *tertiary control reserve* are traded. On this market, the providers of the named products place offers with self-chosen prices. The SOP, who is the single buyer at this market, is obliged to choose the offers with the lowest prices, see Section 2.2.9.2.

2.2.8.3 *Settle Imbalance*

The final phase, called *Settle Imbalance*, ensures that the costs incurred by the SOP are distributed among the BRPS that originally caused them. It takes place after the TIME OF DELIVERY in the *day-after* TIME PERIOD. On overview can be found in Figure 23.

The central role in this process is the IMBALANCE SETTLEMENT RESPONSIBLE (ISR) that is responsible for the orchestration of the following three

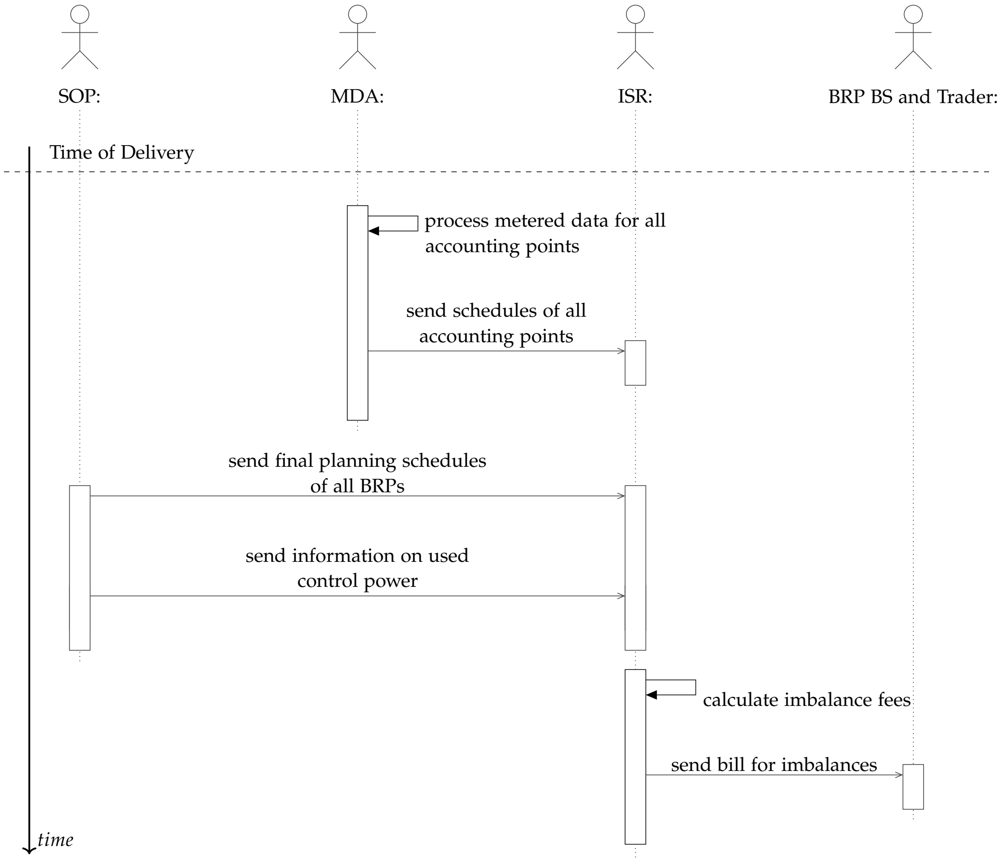


Figure 23: Sequence diagram of the settlement phase of the BMP. ISR: Imbalance Settlement Responsible, MDA: Metered Data Aggregator

basic activities of this phase: The first activity is the processing of all BRP schedules by the SOP and their forwarding to the ISR. In addition, the SOP also determines its expenditures for balancing the grid and provides this data to the ISR as well. The second activity is the determination of the current metered values for each BRP and the respective ACCOUNTING POINTS. Despite being shown after one another in Figure 23, the chronology of these two activities does not play a role. Based on the data determined in the two first activities, the ISR identifies the imbalances for each BRP and determines the respective imbalance fees. These will be charged to the BRP.[20] The specific calculation rules are subject to local market rules. In Germany, they are defined by the decision BK-12-24 [59] of the BNETZA and the related documents provided by the four German TSOS [50].

2.2.9 *Core Markets*

Table 4 provides an overview of the existing energy markets at the wholesale level. More precisely, it is an overview of the German energy market,

20 For simplification, we ignore the role of the billing agent and assign its responsibility to the ISR.

which is typical for the European context, but does not have a general CAPACITY MARKET. The market is divided into the ENERGY-ONLY MARKET (Section 2.2.9.1) and the market for operating reserve (Section 2.2.9.2). An impression of the market volumes of the different energy markets for the German market zone can be gained as well from Table 4.

	Volume [TWh]	**∅ Price** [€/MWh]	**Comment**
Future Markets			
EEX Future Market	1466		Only Phelix future contracts traded at EEX
EEX OTC-Clearing	1367		Only Phelix OTC-Clearing provided by EEX. Overall volume is much higher, roughly 5000 TWh.
Spot Markets			
EPEX SPOT SE Day-Ahead	235	28.98	Mean value of Phelix Day Base
EPEX SPOT SE Intraday	41		
Operating Reserve			
Capacity	73.25	2.70	Market volume is approximated by the sum of the mean values of the tendered capacities for primary, secondary, and tertiary reserve multiplied with the TIME PERIOD of one year. The mean value is the quotient of the total costs for reserving primary, secondary, and tertiary reserve, and the trading volume. Mean values for the individual types of operating reserve differ significantly.
Working	2.1	96.19	The mean value is the quotient of the total costs for BALANCING ENERGY1) and the activated secondary and tertiary operating reserve. Mean values for the individual types of operating reserve differ significantly.

Table 4: Overview of the market volumes for electricity for the year 2016. Data is taken from [30], value for 1) is based on information provided by BNETZA (http://tiny.cc/bnetza_ba_energy_2016 (last accessed: 3 June 2021)).

2.2.9.1 *Energy-Only Market*

In general, the energy market distinguishes between OTC trading and trading with standardized products on electricity exchanges. At the OTC market, products, that are typically not standardized, are traded directly between two parties. In contrast, standardized products are traded on the electricity exchange anonymously and without credit and counterparty risk.

It is therefore not possible to provide an overview of OTC trading and its volumes, as related data is not published. In the following, we will thus limit ourselves to a description of the different markets and products on the power exchanges. A detailed description of the products of the spot market is given in [56], more information on the contracts at the future market can be found in [60].

FUTURE MARKET: The primary purpose of products traded at the future market is to hedge against price-related risks. Therefore, the contracts typically do not involve a physical fulfillment and are purely financial products. The high market volumes in Table 4 must be put into perspective against this background.

SPOT MARKET: Contracts at the spot market usually include a physical fulfillment, meaning that electricity is actually delivered. The place of delivery is limited to the four German TSOS, which form one MARKET AREA.

DAY-AHEAD MARKET: It is organized as a double-sided blind auction that takes place at 12:00 pm daily. Based on the individual bids and offers of the participants, the market is cleared, resulting in an own market-clearing price for every single hour of the day but unique for all market participants and independent from their own bids and offers. The day-ahead market allows complex orders, giving the participants the possibility of interlinking the execution of orders for different hours of the day with so-called block orders. This type of order enables the placing of offers for specific hours that are only executed when all the individual orders are in-the-money. This is particularly important for the marketing of electricity from conventional power plants with ramping limits and high start-up costs.

INTRADAY AUCTION: The intraday auction market is complementary to the day-ahead market. It is also organized as a double-sided blind auction that takes place after the clearing of the day-ahead market. In contrast to the day-ahead market, orders refer to a quarter-hour. The TIME INTERVAL covered by the contracts is therefore equal to the typical TIME INTERVAL of the SCHEDULES. Block-orders are not allowed. The intraday auction is tailored for market participants faced with significant intra-hour ramps, e.g. participants marketing renewable energy.

INTRADAY CONTINUOUS: The continuous intraday market is organized through an open order book where bids and offers are continuously matched. The order book provides information regarding the details of the last trade: price, quantity, and time, as well as a list of all buy and sell limit orders. The minimum TIME INTERVAL covered by contracts is 15 min, block orders are possible. Traders can place different types of orders at the market to influence the execution of them, such as "fill-or-kill" orders that are either entirely executed immediately or canceled.

2.2.9.2 *Operating Reserve*

The market volume of the operating reserve market is relatively small compared to the one of the spot market. In addition, the high value given for the market volume of *operating reserve capacity* given in Table 4 (73.25 GW h) is not directly comparable with the market volume of the day-ahead market that represents an energy amount to be *fulfilled* and not only *reserved*, as in the case of operating reserve. The market for operating reserve is designed as a single buyer market, in which the four German TSOs act as joint buyers in their role as SOPs. All products require a physical fulfillment. The participation in the market is limited to ACTORS that have proven that their technical units are capable of performing the required service level given in [49, Annex D].

The design of the market follows the regulations of the BNETZA as defined in BK-10-097 [61], BK-10-098 [62], and BK-10-099 [63] for the three types of operating reserve, primary, secondary, and tertiary, respectively. All markets are designed as a *pay-as-bid* auction; meaning that market participants get their own price if their offer is accepted. The individual products of the different types of operating reserve differ in the effective direction of the provided power and the covered TIME PERIODS. Effective direction means that either additional power is feed into the grid (positive control power), or consumed from the grid (negative control power).The different types of operating reserve are:

PRIMARY CONTROL is the power that is provided instantaneously by power plants as a function of a deviation from the nominal frequency of the grid. There is only one product for primary control, and it has symmetric effective direction, meaning that positive and negative primary control power cannot be offered individually. Accepted providers for primary control energy are paid by their individual bid and independently of the actually requested energy, which is not paid extra. The product is tendered weekly with a LEAD TIME of about six days.

SECONDARY CONTROL is the power that has to be provided within seconds on request of the SOP. The four related products differ in the effective direction and the TIME PERIOD covered. The products are tendered weekly with a LEAD TIME of about five days. Accepted providers are paid for the provisioning of the power, even if no power is requested, and for the actually requested energy.

TERTIARY CONTROL is the power that has to be provided within minutes on request of the SOP. The overall twelve different products differ in the effective direction and the TIME PERIOD covered. Just like with the secondary control, providers are paid for the provisioning and the requested energy. The products are tendered on each weekday with a LEAD TIME of one day, respectively two or three for the weekend.

FURTHER PRODUCTS were introduced in the younger past. We do not describe them here due to their limited importance.

2.3 CHALLENGES IN TRANSFORMING THE POWER SYSTEM

So what are the crucial challenges in further developing the POWER SYSTEM such that it fits the requirements of an energy system beyond fossil fuels? We would like to approach this question from two directions. In Section 2.3.1, we will discuss it against the background of the objective of an ENERGY SYSTEM based on renewable energies. We will additionally focus here on the possible and probably necessary introduction of components putting a value to locality into the system. In Section 2.3.2, we will discuss additional aspects that have motivated the development of MOCES (cf. Chapter 4).

2.3.1 *Fundamental Challenges in Moving to a VRE-based System*

The fundamental challenge in the further development of the power system is simply formulated. It is the transformation into a system in which energy-related needs of consumers are satisfied without the use of conventional energy sources. On the one hand, this requires a reduction in energy demand, and on the other hand, the development of a system with which the remaining energy demand can be covered on the basis of VRE-based power plants. We have to emphasize the term *transformation* as it is impossible to redesign and implement the system from scratch. It must be assumed that the cornerstones of the system described in the previous chapter and the coordinating processes and markets will be retained in principle and merely developed further in evolutionary terms, on the one hand due to the long operating life of the existing assets, on the other hand due to the anchoring of fundamental mechanisms and responsibilities in laws and ordinances.

The German think tank Agora Energiewende outlines the core challenges with the following three questions Energiewende [64]:

A. How do we synchronize supply and demand?

B. How do we minimize costs?

C. How do we implement the energy transition in a European Context?

The third question already focuses on the possible solution of a strongly developed European interconnected grid in combination with a European single market for electricity. This solution is mainly based on the effect that the feed-in from renewable energies becomes more even and reliable when the system expands geographically. The question which percentage of the load can be covered by VRE-based power plants has still not been conclusively clarified. The DWD [65] has analyzed weather data from Europe for the years 1995 to 2015 and formulated a capacity factor. This factor is defined as the quotient of the feed-in at a selected point in time and the nominal power of the installed wind and PV systems. Considering the area of Germany and assuming a uniform distribution of the installed onshore and offshore wind plants as well as PV plants, the DWD expects two events per year where the capacity factor falls below 0.1 for 48 h. If extending the considered area to Europe the number of expected events reduces to 0.2 per year. However, by defining the factor 0.1 as the critical threshold, it also follows that the installed power required for a purely VRE-based power system is approximately ten times the network load.

Without the availability of large scale storage systems, fossil fuel based CONVENTIONAL POWER PLANTS or power plants using biomass must therefore reserved in order to maintain supply security even in the event of large-area calms combined with low irradiation values.

While the DWD limited itself to the analysis of weather data, Schaber analyses the idea of an *European grid* in more detail. In her doctoral thesis, [66], she deals with the question to what extent a so-called supergrid can help with the integration or VRE-based plants and related questions such as: What percentage reduction in the capacity of conventional power plants can be achieved without reducing the reliability of supply? What is the effect to the overproduction? What is the effect of different divisions between PV systems and wind power plants? What is the optimal layout for the grid extension? Using a linear optimization model based on the the URBAN RESEARCH TOOLBOX: ENERGY SYSTEMS (URBS) framework she suggest to extend the European grid capacities heavily to enable an higher share of VRE without increasing overproduction. Overproduction is energy theoretically produced by VRE-based power plants but not usable due to grid limitations. The proposed cost-optimal network expansions are enormous and are in the order of magnitude of doubling the current capacity of the transport network. The solution proposed by Schaber [66] goes in a similar direction to the promising technical idea of supplying Europe via renewable power plants in North Africa. A solution as discussed e.g. by the Desertec consortium in mid-2000. Unfortunately, the designed systems not only supply people with energy but also require power lines that are built in other people's backyards, such as Corsica, and the construction of generation plants in states that have proved politically unstable or even completely failed in recent years, such as Libya. This human and political dimension is often deliberately ignored in technically oriented studies, since they have not yet been sufficiently understood and are very difficult to model; in the spectrum of modeling and simulation given by the Figure 2, it is at the left edge.

The experience gained in planning new transmission lines from the north of Germany to the south[21] in terms of acceptance is given here as an example. Continued criticism from the citizens concerned has led to a change in planning principles. The HIGH VOLTAGE DIRECT CURRENT (HCDC) line should now mainly be laid underground as cables. According to TSO estimates, the associated cost increases have a factor of three[22]. Without going into these problems further, we would like to summarize them with a quote:

> *No matter how brilliant the technology or perfect the scheme, changes in the energy system and energy usage can only come to fruition with the acceptance and participation of the public. (Cohen et al. [67])*

It is perhaps precisely this experience in the implementation of large-scale projects that fuels the idea of a decentralized organized POWER SYSTEM, in which generation and consumption are primarily balanced locally. Partially, as noted by [68], with a doctrinal point of view that puts the locality's characteristics above everything else.

21 See website of project suedlink: `https://www.tennet.eu/our-grid/onshore-projects-germany/suedlink/` (last accessed: 3 June 2021).

22 Unfortunately, the author is not aware of any study that investigates whether, under the changed framework conditions, the planned grid reinforcement is at all advantageous compared to alternative measures. The network development plan 2022 (See `https://www.netzentwicklungsplan.de/de/netzentwicklungsplaene/netzentwicklungsplan-2022` (last accessed: 3 June 2021).), which justifies the measure, does not provide any information on this either. Specifically, no alternatives are discussed at all in the publicly accessible documents, but only the demand for grid extension is derived from the expected grid load. However, the author expressly points out that he has not conducted an exhaustive literature search on this topic

A decentralized system only seems at first glance to contradict the previously discussed approaches of a strong European interconnected grid. Both ideas can complement each other in a meaningful way. For instance, Schaber [66] also showed in her investigations that a local coupling of the electricity system with the heating sector can reduce the required network expansion.

With regard to Germany, however, it is largely unclear how this mutual complementation should be structured. A consistent development plan prescribed by politics is missing, which is reflected in the regulations; to put it in the words of the the think tank Agora Energiewende: "The present regulation regarding decentralization is a big mess" ([68]). This is extremely critical, since a possible redistribution of costs, see e.g. Figure 12, and the possible incentives which the legislator can offer by reorganizing the respective guidelines gives the legislator enormous leeway. This scope is all the greater when one considers that costs, such as subsidies for renewable energies, which are passed on to the end customers of *electricity* via the EEG apportionments, could in principle also be passed on to other parties, such as to all parties utilizing any energy carrier.

A very striking example of policy failure is the retention of a market design that is known not to be suitable for systems with a high proportion of renewable generation. This problem is known in the literature as the *missing money problem* Joskow [69]. In summary, it is the problem that the yields achievable on the ENERGY-ONLY MARKET are not sufficient to stimulate investments in required generation or storage capacity.

To explain the problem, a simple comparison of the distribution of prices on the energy market, c.f. Figure 11, with the LEVELIZED COSTS OF ELECTRICITY (LCOES) is sufficient. Following [70] the LCOE for lignite-based power plants, which is the type of conventional power plant with the lowest LCOE, is between 45.9 €$MW^{-1}h^{-1}$ and 79.8 €$MW^{-1}h^{-1}$. As Figure 11 shows, even these prices are hardly reached on the market. Gas-fired power plants, which are located at the upper end of the MERIT ORDER and therefore have fewer full load hours, are particularly affected by this problem[23]

Accordingly, a more or less significant decline in the generation capacity of conventional power plants is expected in the next decades. The German "Netzentwicklungsplan"[24] expects a decrease in the generation of conventional power plants from 100 GW in 2019 to 57.7 , 61.2 or 62.0 GW in 2035 depending on the assumption for the further development of the POWER SYSTEM, especially the progressing of the DES [71]. The fore-

23 Fraunhofer ISE [70] also shows that the LCOE of VRE-based power plants are competitive compared to conventional power plants. This applies in particular to wind onshore systems, 39.9 – 82.3 €$MW^{-1}h^{-1}$, and ground-mounted PV systems, 37.1 – 67.7 €$MW^{-1}h^{-1}$.

24 German for grid development plan.

cast for future electricity consumption is also subject to a number of assumptions, particularly with regard to the expected electrification of the transport sector. The named "Netzentwicklungsplan" expect a relative small increase between 1 % and 8 % for the year 2030 compared with the year 2014.

The Market design has a second fundamental problem, which leads us back to the aspect of decentralization. Typically, no value is given to dencentrality, specifically the distance between producer and consumer. An exception, which again does not seem advantageous from an energy policy perspective, is the funding of own consumption through exemption from charges for self-consumed electricity. Individuals are therefore encouraged not to buy electricity from the market but to produce it themselves. In some cases, it is even worthwhile storing the electricity locally. The locally installed systems pursue their own goals and there are no mechanisms in place to ensure that the sum of all decisions taken by the individual systems leads to an efficient overall system. Especially with local storage systems, it is very easy to develop scenarios in which the pursuit of one's own goals does not lead to an overall optimum of the system under today's conditions.

The restructuring of a legal framework therefore appears to be absolutely necessary. A transformation that gives the decentral elements the value they possess. To describe this challenge in more detail, we draw on six theses formulated by Energiewende [68] given here in a summarized form.

DE-CENTRALTY is not a value on its own. Supra-regional POWER SYSTEMS have a fundamental efficiency advantage over local systems. This applies in particular to systems with a high proportion of VRE-based power plants under the assumption of sufficient grid capacity. If decentralisation is given a value, it must be justified, for example by avoiding the expansion of the grid, by economic preferences of consumers or on the basis of political will.

COMPLETELY NEW SYSTEM for fees, levies and network charges. Regionally different are mainly the network charges of the more than 800 distribution grid operators in Germany, ranging between 3.1 ct/kWh and 13.6 ct/kWh for residential customers in 2017. However, these differences do not provide any meaningful incentives, as network charges are paid solely by consumers. The high network charges in Schleswig-Holstein are an example of this. This is mainly due to the large number of wind turbines connected to the distribution grid. Absurdly, grid charges thus correlate positively with a high density of renewable energies and thus contradict the establishment of energy-intensive companies.

INTRODUCTION OF THREE CLEARLY DEFINED levels that structure the electricity market. The lowest level describes local systems in which electricity is exchanged without using the public grid. The second spans larger regions with a certain spatial context. The authors explicitly point out that the size of these regions is still to be determined, but assume a dimension of about 20 to 40 regions in Germany. The last level represents the supra-regional level that links the regions together. The transport network will be used for the exchange between the regions. Different prices are to be expected in the individual regions due to limited transport capacities.

THE REGIONAL POWER MARKETS are a key element. They enable regional preferences of customers to be served and allow the market to respect network restrictions. They are explicitly not isolated from each other but reflect different price zones as they already exist in other systems.

FEES AND APPORTIOMENTS should be related to the three levels. Fees and subsidies should be related to the three defined levels. For example, VAT could be eliminated for the lowest level and the electricity tax could be structured according to the levels.

CAUSER-RELATED grid frees are needed. This should be done in two aspects. Through their own generation, prosumer reduce the fees they pay, which are solely based on the amount of electricity they purchase. Likewise, they continue to use the insurance function of the public grid to the full extent at the expense of those who are rolled over to the constant costs of maintaining the grid. In addition to this problem, the principle of leveled electricity transport costs should also be reconsidered. In the previous design of the system, the transport costs are determined independently of the geographical relationship between producer and consumer and the actual transport costs incurred. In view of the objective of creating a nationwide energy market in Germany, the basically low transport costs and the existing structure of the system, in which load centers coincided with power plant locations, this regulation was logical at the time of its establishment at the beginning of the 2000s. However, this definition ignores the potentially high grid expansion costs caused by a drifting apart of the load and generation centers. A possible new regulation could create additional incentives for the construction of generation plants close to consumption.

2.3.2 *Further Challenges*

Several other concrete problems have motivated this thesis and the development of MOCES. Most of these aspects –and the problems associated with them– are details that arise from the concrete design of the system. The aspects presented in the following make no claim to completeness.

THE MISSING INTERFACE TO THE SMALL ENTITIES As shown in Figure 13, virtually all consumers are connected to the medium and low voltage grid. The vast majority of them without power measurement and without an interface to influence the behavior of the loads. The picture is similar to the feed-in systems. The energy quantities of the individual consumers or producers are small, but in their sum, they have a certain share in total consumption and generation. As explained in Section 2.2.8, although these loads are taken into account in the load and generation balancing processes, their behavior must be considered for granted due to a lack of ability to interact with the systems.

Technically speaking, the necessary interface for exerting influence would be relatively easy to implement. Under the given conditions, however, such systems often do not pay off for the end customer. Let us take the shift in electricity consumption for hot water production as an example. These are about 1000 kWh per person and year. Even if an BALANCE SUPPLIER halves the procurement given in Figure 12 due to a possible load shift, there is only a saving potential of roughly 30€ with which the interface would have to be financed. This potential is particularly low because all other elements of the end customer price are fixed.

If small loads and producers are to be more closely involved in the process of synchronizing demand and supply, then the necessary technical systems must be very inexpensive, and the other elements of the retail price must be more dynamic.

The lack of an interface for small consumer and their poor integration into the processes described in Paragraph 2.2.8 using STANDARD LOAD PROFILES is linked to further detail problems in the better integration and necessary dynamization of small loads. An example is the relationship between PCG and BALANCE SUPPLIER, which is limited to a 1:n relation as given in Figure 19. This means that the PCG has an exclusive supply contract with exactly one supplier. The contract stipulates, that the BALANCE SUPPLIER is financially responsible for the consumption and the differences between the forecasted and actual load. A problem arises if the PCG is contracted with a third party influencing its load behavior. In this case the BALANCE SUPPLIER is responsible for a deviation from the forecasted behavior caused by a third party.

In the German context, this problem already occurs with larger loads providing control power, whose call-off is not coordinated by the BAL-

ANCE SUPPLIER, but by a third party, which offers the FLEXIBILITY bundled as balancing power. Unfortunately, in the BK-17-046 [44] administrative procedure, that addresses this problem, a process was developed that is specially tailored to the specific use case without fundamentally rethinking the relationship between PCG, the BALANCE SUPPLIER and set of third parties. The basic idea of the process is that substitute measurement values are created for the supplier, which represent the load behavior of the PCG as if no intervention had taken place on the part of the third party.

Similar constellations are also emerging for household customers. These are made possible by existing interfaces to individual household appliances, which allow the individual load to be shifted. Typical examples are networked white goods and heat pumps. The question of how to deal with this influence and in what form the information about an influence by a third party must be communicated is largely unanswered to the author's knowledge.

TSO in particular are somewhat skeptical about this development, as they fear that the resulting, significantly more dynamic, system will be more difficult to manage. They attribute a certain disruptive character to these developments, which call into question the foundations of the electricity system. Ben Voorhorst president of ENTSO-E expressed this skepticism with the question "What will be the Uber moment for us?"[25].

HOMOGENEOUS PRODUCT AND IDEAL MARKET The previous section on the need for a local component in the power system shows the need to rethink some of the building blocks in organizing the power system. One of these specifications is the handling of electricity as a homogeneous product, which is mainly characterized within a market zone only by the TIME OF DELIVERY. Technically, this assumption is correct. Electricity supplied via the public grid does not differ from supplier to supplier. It is perfectly replaceable. The technical quality level, e.g. in terms of availability, depends on the network operator and cannot be influenced by the suppliers.

From the point of view of the customer buying electricity, however, this is no longer correct. He assesses the product electricity also on the basis of its origin, which has two dimensions: the primary energy source used to produce electricity and the geographical location of electricity production. For consumers within the DEMAND SECTOR household, this is shown in [72]. These two attributes can be supplemented by a third attribute for electricity, which describes the quality level of the power in terms of availability. This level would describe how likely it is that the energy purchased is actually available at the TIME OF DELIVERY. None of these attributes can be 'attached' directly to the electricity as electri-

25 See `https://www.euractiv.com/section/energy/news/power-grid-operators-expect-their-uber-moment/` (last accessed: 3 June 2021).

cal energy is not a packet routed through the grid, including the sender address, destination address and additional labels but if wanted the information can be processed supplementing IT-systems.

In principle, even under the current regulatory framework, the properties described can be rewarded by the customer and specific products already exist to meet the customer's wishes. There are green electricity tariffs and an increasing number of tariffs offering regionally produced electricity. However, the characteristics have no effect at all on the taxes and duties summarized in Figure 12.

The third possible attribute, the availability, additionally requires the possibility of actively intervening in electricity consumption, as the technical quality level is not affected. A technical system must therefore guarantee that the 'conditional energy' is actually consumed only when it is available. This new type of electricity product would be interesting for electricity storage and electric motor vehicles, which could easily deal with a possible interruption of energy supply.

As Bagemihl et al. [73] show, the usefulness of this product lies in the fact that it could use grid capacity, which has been kept free for safety reasons, but which could be used for conditional energy if it can be guaranteed that the load can be switched off without any problems immediately. In concrete terms, the authors propose not to include the 'conditional energy' in the N-1 SECURITY assessment, as N-1 SECURITY is by definition not essential for conditional loads. The author does not provide reliable data for the potentials of this approach, but in an example in the context of a DSO the grid capacity is almost doubled. The authors of [73] also provide a first approach for an system architecture and a possible process for the interaction of the different ACTORS involved. The study of is a good example that a modification of the processes shown in Section 2.2.8 can have a decisive influence on the efficiency of the overall system.

THE FAIL OF DIVIDING ACTORS INTO PRODUCERS AND CONSUMERS PCGS are subdivided into consumer and producers, cf. Figure 19. This classification was self-evident in the classic POWER SYSTEM and could be carried out without any problem. Numerous rights and obligations of the ACTORS are linked to this classifications assuming that a PCG is either one or the other and does not provide for a different type of PCG.

A third class of ACTORS, those main characteristic is to consume *and* produce electricity is typically not foreseen, especially not in the regulation. Storage systems in particular require special handling from the regulatory point of view.

2.4 CHALLENGES REGARDING MODELING AND SIMULATION

For decades, modeling and simulation has been one of the essential methods for making design decisions regarding the structure and operation of the POWER SYSTEM on a sound basis. A large number of different kind of models are used. The types of models range from those that examine specific aspects of the system in detail to those that attempt to model the system as a whole. Due to the focus of MOCES, we want to concentrate on the latter.

However, the evolution of the POWER SYSTEM described in the previous section not only changes the system to be modeled but also results in challenges concerning modeling and simulation itself as the models have to change themselves to reflect aspects not considered before.

We, therefore, describe these challenges in the following Section. Based on Heard et al. [74], we take a very critical review in Section 2.4.1 on the use of models and simulations in the context of POWER SYSTEMS. In the subsequent Section 2.4.2, we present the key challenges in more concrete terms. This Section focuses on characteristics of the future power system that needs to be considered more closely and is based on Pfenninger et al. [4]. In Section 2.4.3 we would like to discuss some so-called non-functional challenges such as open models and data availability. Finally, we end with a graphical illustration of possibly required model features in Section 2.5.

2.4.1 *Are all Models Wrong?*

The question about the correctness of models is answered by George E. P. Box in the mid 1970 with his famous quote[26]:

> *All models are wrong, some are useful.*

The challenges that arise from this for the creation of models are summarized by him as follows:

> *Since all models are wrong the scientist must be alert to what is importantly wrong. It is inappropriate to be concerned about mice when there are tigers abroad. (George E. P. Box [75])*

This quotation relates the concept of the EXPERIMENTAL FRAME introduced in Section 1.2 as it defines the set of experiments for which the model is useful. But what if there is no agreement about what EXPERIMENTAL FRAME is needed to answer certain questions? What if it is not clear which detail can and which cannot be abstracted in the modeling

26 George E.P. Box was a statistician. His statements therefore originally refer to statistical models. However, the statements can also be transferred to other types of models.

process? What if there is no consensus on which aspect is a mouse and which is a tiger?

It seems farcical, but there is widespread disagreement in the scientific community on these aspects for models addressing the POWER SYSTEM. Heard et al. [74] draw a very gloomy picture in the article "Burden of proof: A comprehensive review of the feasibility of 100% renewable-electricity system" published in the relatively high-ranked journal *Renewable and Sustainable Energy Reviews*. The authors investigate the feasibility of 24 studies which evaluate the question whether a POWER SYSTEM solely driven by VRE-based power plants is viable or not. The studies were selected according to several parameters, such as to consider larger geographic regions like nations. The summarized result of the study is alarming. On a scale from 0 up to 7, the best performing study has a rating of 4; four studies have a rating of 0.

To evaluate the studies, the authors define four criteria which are used to score the different studies.

A. The electricity demand must be projected realistically. According to the authors, a scenario assuming a reduction in demand of electricity is unrealistic.

B. The coverage of demand must be considered time-resolved. Seldom weather events such as long periods of calm spanning larger regions should be examined. The full score for this criterion requires a resolution of five minutes; a time resolution of half an hour gives two out of three points.

C. The used model must cover the transmission grid and its restriction concerning capacity.

D. Ancillary services required to ensure power quality have to be included in the consideration. The authors notably name frequency control and voltage control.

The last two criteria were met by the fewest models. Only one model considers a temporal resolution of 5 minutes.

The actually frightening fact is not even this destructive evaluation, it is the answer to the article in [76]. The authors claim that the criteria of [74] used to evaluate the feasibility are heavily overrated, and other criteria not named are much more critical regarding the feasibility of a model. To put it less bluntly, the authors of [76] say that the evaluation is wrong.

We are not in a position to evaluate the differences of opinion. Some of Brown et al. [76]'s objections seem reasonable. For example, the statement that frequency maintenance is a secondary problem that, if no smarter methods are found, can be solved at acceptable costs with conventional technology. Other statements are less convincing and remain on a narrative level. For example, the back of the napkin estimate that is used to refute the criterion regarding rare weather phenomena. According to this calculation, the peak load could be covered by gas turbines at reasonable costs in case of doubt. The estimation is based on the investment and operation costs for the power plant solely but neglect cost for the infrastructure.

Ultimately, the dispute between the two research groups shows a fundamental problem which is also the most significant challenge. There is no solid foundation, no established methodology that can be used to derive the EXPERIMENTAL FRAME needed to answer specific questions.

This lack of foundation means that every modeler has to decide for himself which aspect have to be modeled in detail and which ones can be abstracted to what extent. To sum it up with a quote that complements Box's statements from the beginning of that Chapter:

> *Modelers must also make sure to avoid the trap of modeling what is easily quantifiable rather than what are the essential driving variables of the system. (Pfenninger et al. [4])*

2.4.2 *Main Challenges*

The following overview of the basic challenges in modeling POWER SYSTEM is motivated by the listing of Pfenninger et al. [4] and extended by own investigations.

2.4.2.1 *Resolving Details in Space and Time*

The challenge *Resolving Details in Space and Time* is mainly driven by the increasing and high number of VRE, that are unevenly distributed and whose dynamic behavior depends on the actual boundary conditions at the locations of the VRE.

The feed-in of individual PV and wind power plants is highly dynamic. Primarily, the feed-in power of PV plants can drop to a fraction of the output within seconds due to the passing of clouds. Wind turbines have lower maximum load gradients compared to the dynamics of PV plants, but they are also relatively high. However, if considering the aggregated feed-in values of ensembles of VRE-based power plants the high ramp rates are smoothed out due to a low correlation factor for the changes of the power output. Analyzing feed-in values of wind power plants, Ernst [77] has shown that the correlation factor for the *changes* within five minutes is almost zero if the distance between the wind power plants is above several kilometers

This statistical smoothing effect allows to model and simulate nationwide POWER SYSTEMS with a relatively coarse temporal resolution of 15 min or even 1 h without losing accuracy if investigating the cost for the entire power system as shown by Deane et al. [78]. Their investigation considering the Irish POWER SYSTEM shows that the estimated total generation costs differ by only 1 % depending on either using a 5 min or 60 min resolution. However, the investigation also shows that other simulation results such as the estimated number of start-ups of power plants used to balance production and consumption heavily depend on the chosen temporal resolution.

To put the named temporal resolutions into relation, we have to mention, that classic energy models used to consider the long-term evolution of ENERGY SYSTEMS and POWER SYSTEMS as described in Section 3.2.1 are using much coarser resolution in time.

2.4.2.2 *Handling Uncertainties*

Pfenninger et al. follow Kiureghian and Ditlevsen [79] who distinguish between two different types of uncertainties: epistemic and aleatory uncertainty. The first one addresses uncertainties due to a lack of knowledge or consciously chosen simplifications, such as using a linear model for an aspect known to be non-linear. The second one addresses uncertainties that are an intrinsically characteristic of the modeled system, such as the diced number if the system to be modeled is a dice. The first type of uncertainty is a fundamental problem. As a result, findings from energy models are often fundamentally questioned as exemplarily discussed in 2.4.1. Significant uncertainties such as the behavior of the humans affected by the POWER SYSTEMS and in turn affecting the POWER SYSTEM have hardly been understood so far.

Aleatory uncertainties can be included into the models, but typically lead to an escalation of the model complexity. Abrell and Kunz [80], e.g., developed a stochastic model to analyze the effect of uncertain wind generation to the POWER SYSTEM with particular focus on the resulting dispatch of the power plants. Although the authors use methods to reduce the number of scenarios, they end up with a simulation time more than ten times as high as compared with a model that does not include forecast uncertainty.

2.4.2.3 *Complexity and Optimization across scales*

The challenge *Optimize Across Scales* is heavily interlinked with the challenge *Resolving Details in Space and Time*. It addresses the requirement to include phenomena of lower scale into the actual scale of the model [4].

A typical example in the context of energy system models is the integration of the (hourly) plant and storage dispatch into models addressing the optimal plant and grid investment planning. A similar challenge is posed by the integral modeling and simulation of the distribution network and the transport network against the background of a more actively operated distribution network. Palmintier et al. [81] publish a PROOF OF CONCEPT (POC) in which the Northern California grid with more than 1.4 million nodes was simulated for one day with the help of high-performance computers. However, the run time of the simulation was about half a day. This is a period of time for which the model is not suitable for many applications. Again, the correct choice of abstraction level depending on the application of the model is the primary challenge.

2.4.2.4 *Human Dimension*

The human behavior is the major source for uncertainties. In addition, as stated by Pfenninger et al. [4], the influence of the so-called human dimension is often neglected, especially when examining the long-term development of the energy system. This is a fundamental problem because political will and public acceptance have a large, if not the decisive, influence on the further development of the system.

The influence of the the human dimension is not limited to the long-term development of the energy system; it plays an important role, especially in concepts for the dynamization of consumption. How users react to dynamic prices, for example, is still an object of investigation in empirical studies. Klaassen et al. [82] stated, that the relatively few studies show no uniform picture. Following to the authors, among others things, due to different appliance available at the households.

Even if scientific research such as [83] examines the role of the user in future energy scenarios, the behavior of the users and their reaction to new technologies is hardly understood so far and leads, according to Pfenninger et al. [4], to the tendency to neglect these important aspects in modeling.

Systems that are characterized by a strong interaction between humans and technical systems are also known as SOCIO-TECHNICAL SYSTEM. Due to the characteristic of the system described so far it is therefore not surprising that the ENERGY SYSTEM is listed as a prime example of SOCIO-TECHNICAL SYSTEM, e.g. in Dam et al. [84] and Dam [85]. For the development of MOCES we will therefore use some of the concepts for modeling these systems.

2.4.3 *Non-Functional Challenges*

There are many other challenges in the development of models which we would like to outline briefly due to their importance and relevance for this work. Despite being essential aspects, we will not discuss them in detail, as this has already been done elsewhere. We would like to point the interested reader to Pfenninger et al. [86] and Pfenninger et al. [87], which discuss these topics comprehensively.

AVAILABILITY OF DATA The creation and validation of models require a wide variety of data. Power plant lists, historical feed-in time series, installed capacity of renewable energy plants, weather data, and market prices are just a few examples. There have only recently been increased efforts to make these kinds of data available openly. Examples of this are the *smard* data platform[27] hosted by the BNETZA, that is available since mid-2017 and provides market data for Germany, and the *SciGrid* project[28] that offers an open-source model of the European transmission grid[29].

OPEN MODELS For a long time, models that form the basis for statements and recommendations were understood as treasures to be hidden by the institutions that created them. Accordingly, they were usually not made available to the community, not even for research purposes. The fact that the corresponding projects and developments were often supported with public funds is also a questionable aspect of this story. The insight that such a treasure can multiply miraculously when given to the public has only recently come to light. A positive example in this context is the *oemof* modeling framework[30].

2.5 SUMMARY

In this chapter, we have provided a holistic description of the energy and power system, and elaborated the challenges for their MS. Some partial aspects, such as the BMP or the provision of control power, have been discussed in more detail whenever they were relevant for the understanding of the present work.

We will conclude and summarize the chapter with a graphical representation of the different aspects of the energy and power system, shown in Figure 24. This kind of representation is adopted from Schaber [66], Wiese [88] and Haller [89]. The individual aspects are called features here, as the representation was created against the background of the development of a model that represents or neglects certain aspects. A feature thus stands for a sufficiently exact modeling of an aspect.

27 See `https://www.smard.de/home` (last accessed: 3 June 2021).
28 See `http://scigrid.de`, (last accessed: 3 June 2021).
29 We list *SciGrid* as a positive example in the open data category rather than in the open model category, since the main contribution of *SciGrid* is the extensive data set of the network model.
30 See `https://oemof.org/` (last accessed: 3 June 2021).

The location of the features, based on the two chosen dimensions (temporal and spatial resolution), and their range thus illustrate once again the multitude of different but interlinked aspects and the different scales on which they occur. The graph also serves as a bridge to the next chapter, which will discuss the different approaches to 'implement' the features in models.

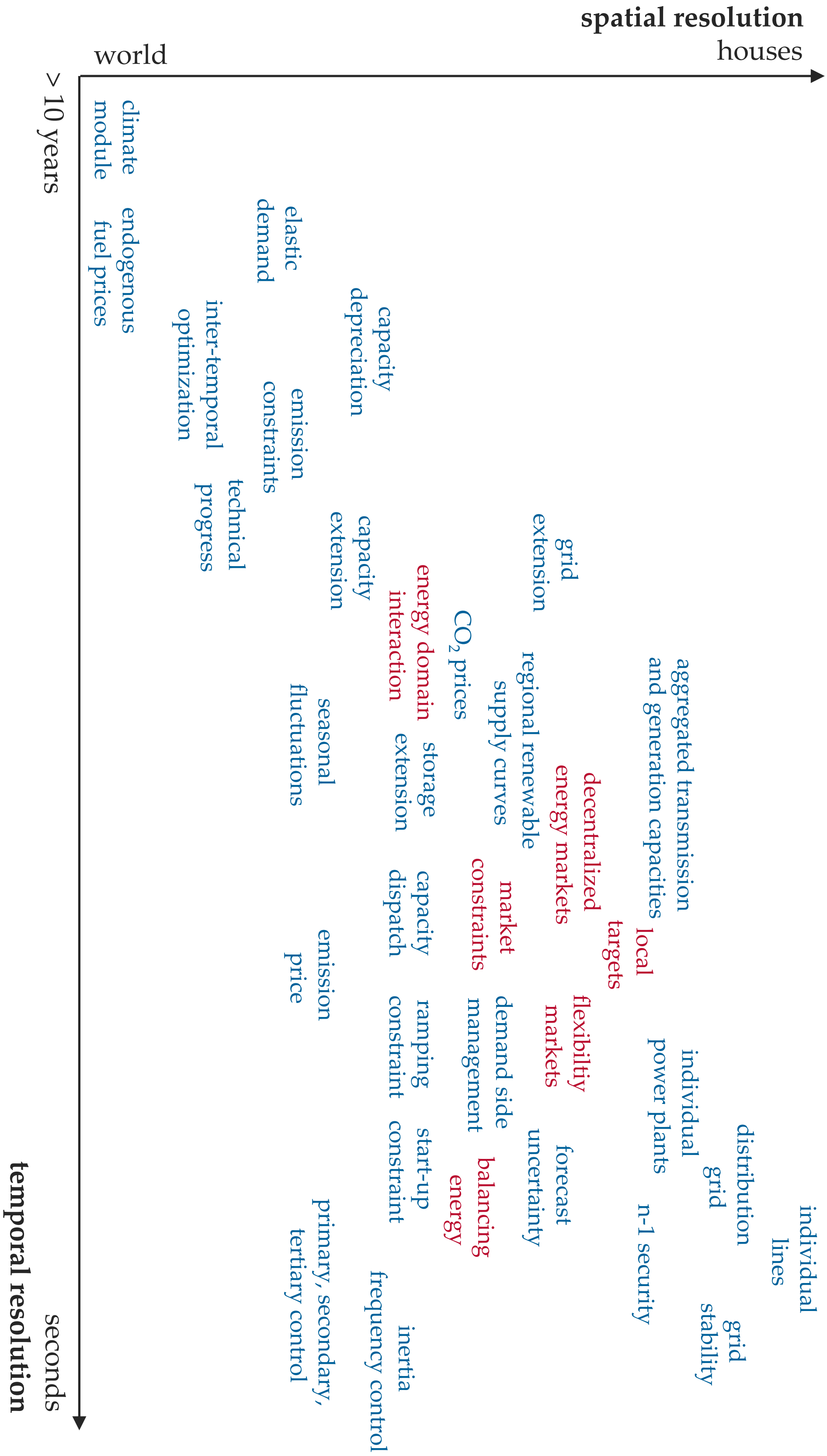


Figure 24: Model feature space adapted from [66]. Features colored in red are added by the author.

3 MODELING AND SIMULATION APPROACHES

In the previous chapter, we addressed the transformation of the ENERGY SYSTEM and the POWER SYSTEM and the resulting challenges to model and simulate the systems. In this chapter, we will give an overview of the different approaches, models and tools that strive to meet these challenges. In contrast to the rather abstract view so far, we will describe concrete questions and the models that are used to answer them in detail in this chapter.

The chapter is organized as follows. In Section 3.1 we elaborate the difference between two approaches that are often mixed-up, OPTIMIZATION and SIMULATION, and the difficulties to group existing models. In Chapter 3.2 we will introduce several models with a focus on their EXPERIMENTAL FRAME and the methodological framework used.

As MOCES uses MODELICA and its *equation- and component-based* modeling approach, Section 3.3 includes a short primer of MODELICA, and a critical examination of its suitability for the modeling and simulation of complex energy systems. Section 3.4 outlines the main characteristics of the *agent-based* modeling and simulation approach, which MOCES uses in addition to the equation-based modeling approach.

3.1 TYPES OF MODELS

3.1.1 *Types of Models by Method: Optimization vs. Simulation*

The different types of models used for the analysis of energy and power systems can be distinguished based on whether either the OPTIMIZATION MODEL APPROACH or the SIMULATIVE APPROACH is applied; they have been introduced in Section 1.5 (p. 15 ff.). Both approaches are widely used, but they differ substantially. We will outline this difference and the associated advantages and disadvantages in the following.

The OPTIMIZATION MODEL APPROACH requires that the entire system is described as an optimization problem which can be solved by a suitable algorithm. The main drawback of this approach is the rigidity of the mathematical formulation, as the expressivity of the model is limited to the chosen class of optimization problems, such as linear programming, mixed integer linear programming or non-linear programming. In general, there is a trade-off between good performance and high expressivity. The main benefit of the OPTIMIZATION MODEL APPROACH is that the formulation of the optimization problem includes the optimization goal and therefore gives a direct answer to the question of how the system should be operated or designed based on the boundaries given in the problem formulation.

The SIMULATIVE APPROACH does not include an optimization goal intrinsically, but it brings the freedom to use highly expressive models and even to mix different modeling approaches. Besides, it is possible to process individual parts of a model by different kinds of simulators that are linked in a co-simulation approach.

Both approaches are basically used for any kind of model feature, i.e., in any region spanned by the temporal and spatial dimensions of Figure 24. In the field of POWER SYSTEM models, however, models following the OPTIMIZATION MODEL APPROACH tend to address large regions with a relatively low spatial and temporal resolution.

A useful separation of the two approaches is given in Pfenninger et al. [4], where the distinction is based on the respective purpose. The purpose of the SIMULATIVE APPROACH is to *forecast* the behavior of the modeled systems. In contrast, the primary goal of the OPTIMIZATION MODEL APPROACH is to make *normative statements*. How the underlying modeling and simulation processes differ for the two approaches was already shown in Figure 3.

Lund et al. [90] discuss the two different approaches in detail. Concerning energy models, the authors argue that the advantages of the simulation approach outweigh the optimization approach. The main reason for this recommendation is that a simulation-based approach typically analyzes different scenarios against each other, whose fundamentals can by given by politicians. Compared to that, optimization-based approaches tend to determine one *optimal solution*, losing alternative solutions in the optimization process, and thus only provide that one solution to politicians without showing alternative pathways.

The two approaches do not necessarily exclude each other. Hybrid forms are possible and common. There are both simulation models that include optimization models and optimization models that include simulation models, cf. Lund et al. [90].

3.1.2 *Types of Models by Purpose: Model Families*

In order to provide an overview of the multitude of models, it seems suitable to classify them according to their application or usage, to thereby assign them to model families, and to describe these families.

However, this endeavor proves to be difficult for several reasons. One reason is the sheer number of very specific models due to the tendency of developing models for particular applications. Another reason is the questionable procedure of developing models behind closed doors, and to not disclose and share them even if they have been used to make political decisions. In the overview paper [4], the authors mention no less than ten articles which try to give an overview of the different approaches and models, some with a relatively broad focus, others with a focus on a specific application area such as energy market modeling.

The challenge of model classification is also mentioned within all overview contributions given in the following. As a result, the clusters chosen by the respective authors differ considerably and are mainly influenced by the selection of the considered models. Jebaraj and Iniyan [7], who focus on models and tools for national or supra-national energy systems, introduce the following grouping: "energy planning models", "energy supply-demand modes", "forecasting models", "optimization models", "models based on neural networks" and "emission reduction models".

Connolly et al. [6] use the following six tool types, which do not exclude each other, to classify the different tools: "simulation", "scenario tool", "equilibrium tool", "top down-tool", "bottom-up tool", "operation optimization tool". Their work provides a very comprehensive overview; the models and tools are not limited to a specific focus. The list thus includes tools for supra-national consideration as well as tools for modeling buildings and their HVAC[1] systems.

[91], who focus on tools for urban development, distinguish between "scenario models", "operational models", and "long-term planning models".

Schaber [66] classifies the models based on their temporal scope in analogy to the model features given in Figure 24. Therefore, she distinguishes between models that focus on long-term development, those used for operational decisions and those that lie in between, named HYBRID MODELS.

Connolly et al. [6] additionally point out that even a common language to unambiguously describe the functionalities of the models and tools is missing. Wiese et al. [92] argue that there is also a lack of methods to evaluate models against their suitability for a specific purpose, cf. Section 2.4.1 (p. 74 ff.). The authors therefore develop the first approach to an evaluation concept.

Despite these difficulties, we attempt a high-level grouping in the next section.

3.2 CLASSIC MODEL FAMILIES

Confronted with the high number of approaches and methods, and their non-uniform classification into model families, we have opted for the following procedure to provide an overview. Based on a high-level grouping performed by Pfenninger et al. [4], we will outline four different

1 HEATING, VENTILATION, AND AIR CONDITIONING

model families with a clear separation between OPTIMIZATION and SIMULATION. We entitle these families as *classic*, as prominent representatives have existed for years or even decades.

3.2.1 *Energy System Optimization Models*

Energy system optimization models are used since the 1970s to address macroeconomic and ecological questions regarding the entire ENERGY SYSTEM. Typically, they support national long-term strategic energy planning [4] and the design of the respective energy policy to control the development of the system. In this context, the energy system is understood as a process, which begins with the extraction of primary energy and ends with serving energy to fulfill the demands of different DEMAND SECTORS, such as building, transport or industry. Therefore, this model type intrinsically includes different ENERGY CARRIERS, focuses on a national, international or even global geographical level, and handles time spans up to several decades. To enable this global perspective, a high degree of abstraction is needed, and typically an optimization model, mostly linear, with appropriate solvers is used. The primary purpose of this model class is to answer the question of how to build the energy system.

3.2.1.1 *Long Term Planning, Example:* MARKAL

The initial MARKET ALLOCATION (MARKAL) model was the result of joint work of seventeen nations and two international agencies, namely the *Brookhaven National Laboratory* and the *Kernforschungsanlage*[2] *Jülich*. MARKAL is not a concrete model for a specific nation, it is a modeling paradigm that provides a standardized approach to model the ENERGY SYSTEM of a nation. Since the modeling process results in a linear optimization model, it is often referred to as a model generator. The development process started in 1976. In 1980, models for countries such as Belgium, Ireland, and Norway existed, and first results were published [3]. Under the backdrop of the oil crisis of the 1970s, one particular goal of the MARKAL model was to help to plan an ENERGY SYSTEM that depends less on oil imports. Therefore, one requirement was to be capable of evaluating the attractiveness of existing and new technologies to fulfill future demands. Besides, it should also show the sensitivity of the system regarding changing targets and imported energy. From a modeling perspective, MARKAL is a multi-period linear-programming model that uses an externally generated price-inelastic projection of the USEFUL ENERGY DEMAND of the different sectors to determine the best design of the system under the constraint that the demand is met in every single

2 German for Nuclear Research Center

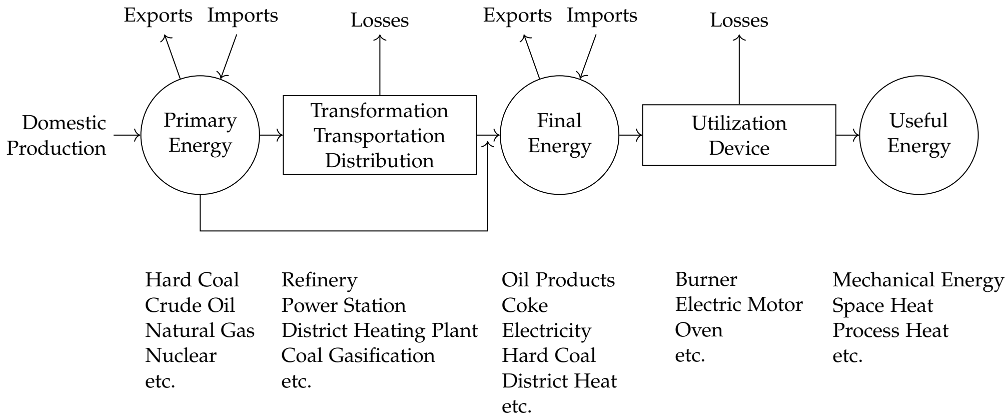


Figure 25: ENERGY SYSTEM MODEL as used by MARKAL, adapted from [93].

TIME SLICE. Time slices represent regular periods such as 08:00 – 20:00 in the winter of the first observation year. The use of such time slices leads to a very rough abstraction of the temporal behavior. Nevertheless, even this kind of time handling allows to take into account the dynamic load during the day and its seasonal dependence to a limited extend, and to model the daily shift of energy when utilizing storage systems.

As it is typical for ENERGY SYSTEM optimization models, MARKAL uses a process-based approach to model the energy system. Figure 25 shows an aggregated flow chart, as used by MARKAL to represent the entire energy system. Unfolding the aggregated flow chart for a concrete system results in different paths that satisfy the demand, cf. Figure 7. It is important to note that in such a model the amount of USEFUL ENERGY needed by a specific sector is fixed, while the way through the system is not. For example, the USEFUL ENERGY required for residential space heat is fixed, but it can be provided by different *utilization devices* which use various forms of final energy, such as electricity, oil, natural gas or a mixture of several kinds.

To use the MARKAL model for a specific nation, the modeler has to define the *reference energy system*, that includes all possible paths from the sources to the demands. Also, he or she has to set the nation-specific technical and economic parameters of the individual processes and energy carriers, such as the efficiency of a process, the different transport costs and the investment costs. Based on this information, the optimization model provides the best energy system.

The initial version of MARKAL had a very coarse resolution in space and time. It did not consider regional aspects at all and used not more than 54 TIME SLICES to model a typical time horizon of 45 years. MARKAL has been continuously developed and expanded in the last decades in the context of the ENERGY TECHNOLOGY SYSTEMS ANALYSIS PROGRAM (ETSAP) under the patronage of the INTERNATIONAL ENERGY AGENCY (IEA).

Following information available on the project website[3], the model is still supported and has a user group of more than 70 institutions in 37 countries. Seebregts et al. [94] give a comprehensive overview of the developments over the years. The first extensions were related to an inclusion of a macroeconomic growth model that supersedes the price-inelastic exogenously given USEFUL ENERGY DEMAND. Later, it was extended to handle several interconnected regions.

THE INTEGRATED MARKAL-EFOM SYSTEM (TIMES)[4] is the official successor of MARKAL. TIMES uses the same modeling paradigm as MARKAL but enables more flexibility regarding the definition of TIME SLICES. They are no longer fixed in length for the complete modeling duration, e.g., the first few years can be described in more detail than the rest of the overall period. TIMES further allows modeling TIME SLICE-dependent parameters and expands the capability to model storage systems. In MARKAL, only daily storage of electricity could be modeled [95].

The OPEN SOURCE ENERGY MODELLING SYSTEM (OSEMOSYS)[5] should also be mentioned in the context of MARKAL and TIMES. It shares the main principles and can be used for similar purposes. However, it is designed as an open source tool enabling a fast learning curve and can therefore be used easily for educational purposes as well.

The main advantage of the named models is their holistic perspective on the ENERGY SYSTEM including the emissions of the considered system. As the output of the models is the least-cost energy system under given restrictions, the models can be used to identify cost-effective responses to limits on emissions and to evaluate the effect of regulations, taxes, and subsidies[6].

The main drawback of both approaches is their coarse resolution in space and time, which makes a correct assessment of the influence of local and time-dependent renewable energy plants on the overall system difficult to impossible. Overcoming this main drawback is one of the primary challenges named by Pfenninger et al. [4] and described in Section 2.4.2.1. However, this problem is intrinsic, as the scale of the resulting optimization problems inevitably increases with higher resolutions.

3.2.2 *Energy System Simulation Models*

Energy system simulation models are highly related to energy system optimization models (cf. Section 3.2.1). Following [4], energy system sim-

3 See `http://iea-etsap.org/index.php/etsap-tools/model-generators/markal` (last accessed: 3 June 2021).

4 See `http://iea-etsap.org/index.php/etsap-tools/model-generators/times` (last accessed: 3 June 2021).

5 See `http://www.osemosys.org/` (last accessed: 3 June 2021).

6 See `https://iea-etsap.org/index.php/etsap-tools/model-generators/markal` (last accessed: 3 June 2021).

ulation models are somewhat younger; first models were developed in the late 1980s. They also focus on the long-term evolution of national or international ENERGY SYSTEMS and therefore intrinsically include the different ENERGY CARRIERS as well as the various DEMAND SECTORS. Furthermore, they also use similar spatial and temporal resolutions and are also mainly used in the political decision-making process, but, compared to energy system optimization models, they answer a different question. The primary purpose of the class of energy system simulation models is to answer the question of how the system will evolve given different scenarios. For this purpose, the model describes the system's behavior by the interaction between various subsystems that represent separate supply and demand sectors, sometimes also referred to as agents or modules. The interaction between the agents is mostly modeled by their participation in markets. This approach brings great liberties in describing the behavior of individual agents, that can also include decisions on investment.

3.2.2.1 *Long-Term Planning, Example:* PRIMES

The PRICE-INDUCED MARKET EQUILIBRIUM SYSTEM (PRIMES) model was developed by the *National Technical University of Athens* in the mid-1990s and is still maintained by the same university, respectively its *E3MLab*. The PRIMES model simulates the European ENERGY SYSTEM and is used to forecast its evolution; e.g., it was heavily used in the EU Reference Scenario 2016 that provides a possible pathway for the ENERGY SYSTEM up to the year 2050 under the assumption that the action targets for 2020 regarding greenhouse gases and renewable energy sources will be reached. Despite this central role and ongoing criticism, e.g., by [87], the model code is closed source. However, the lab provides a model description [96], which we refer to in the following.

PRIMES uses a modular structure in which each module, called agent, represents the behavior of a type of energy consumer or producer, such as the households of a specific nation. The different modules are linked to each other via markets, for which an integration algorithm determines a market equilibrium or a state in which all markets are in equilibrium simultaneously. Concretely, it determines an equilibrium by finding a price for each ENERGY CARRIER such that the demand meets the supply; the equilibrium is static for each model time step that covers 5 years. The spatial resolution of PRIMES is limited to nations.

The core element of PRIMES are individual agents that try to minimize their costs or maximize their revenues. Agents who consume energy are modeled as price takers who perform a minimization based on a sophisticated projection of the energy demand. The demand for USEFUL ENERGY, such as low-temperature heat in the industrial sector, is modeled by an individual consideration of nine sectors, such as building material production, and a total of 24 sub-sectors, like cement, ceramics or glass. To meet this demand as cost-effectively as possible, the respective agent can not only use various technologies with different needs of final energy; it can also invest in energy efficiency measures. This agent-based modeling leads to a characterization of PRIMES as a model with microeconomic foundations.

For the model part that reflects the POWER SYSTEM, the modular structure allows for a higher temporal resolution than the otherwise used five years. According to the authors, PRIMES uses a "very detailed model for electricity generation, trade and supply and for steam generation and distribution" (E3MLab [96], p. 113). However, this statement of detailed modeling must be put into proportion. The electrical grid and its restrictions are mapped, but only with one node per country. The simulation of the system is not carried out for every single hour, but on the basis of selected type days; the demand for electricity is modeled as price-independent. Following the documentation of PRIMES, the system is simulated by formulating a unit commitment problem; we will further explain this type of modeling in Section 3.2.3.1.

The typical use case of the model can be described best with the input and output variables of a model run. We list the most typical ones in the following:

Inputs

- Development of gross domestic product and economic growth per sector
- Development of world prices of fossil fuels
- Taxes and subsidies
- Climate action targets
- Characteristics of future technologies
- Potentials for RENEWABLE ENERGY SOURCE (RES)

Outputs

- Detailed energy balances for different ENERGY CARRIERS
- Demand projections for USEFUL ENERGY
- Investment in different sectors
- CO_2 and further greenhouse gas emissions
- Energy flows between European countries and between Europe and other states

When considering the input variables, it becomes clear that they themselves are in reality dependent on the energy system, even though they are assumed to be fixed. Assumptions made for the inputs and thus also for the results of the models are therefore subject to uncertainties. The model developers therefore describe the ability of their model with the term *projection* rather than with the term *forecasting* to underline the purpose of the model, which is to project the long-term development of the energy sector based on assumptions for the economic evolution. To be able to model interactions with the economy, PRIMES can be used in a closed-loop simulation along with the GEM-E3 model[7].

7 See `https://ec.europa.eu/jrc/en/gem-e3/model` (last accessed: 3 June 2021) for an overview.

Like its US counterpart NEMS[8], PRIMES is one of the most complex models in the field of energy system analysis. However, this complexity also means that the models, even if they are under the public domain like NEMS, are hardly used outside the institutes that develop them; a broad discussion of the assumptions underlying the models is therefore difficult up to impossible.

3.2.3 *Power System Optimization Models*

Power system optimization models focus on the electricity system and largely ignore other ENERGY CARRIERS. Typically, their modeling granularity regarding spatial resolution, temporal resolution, and modeling depth is significantly higher compared with ENERGY SYSTEM models.

Based on the optimization goal, the model family can be further divided into two sub-families: models used to optimize the operation of a system under the assumption that the structure of the system is fixed, and models used to find the optimal structure or an optimal development of an existing structure. In the last decade, a third group of models came into the focus of the research community: the so-called *hybrid models* are placed between the two aforementioned families. These models have the same purpose as those models that determine an optimal structure, but they also describe the daily optimization to correctly reflect the effect of VRE-based power plants on the optimal structure of the POWER SYSTEM.

3.2.3.1 *Operation Models, Example: Unit Commitment Problem*

In the period before the liberalization of the energy market, operation models were used to operate the system optimally. Optimal in this context typically meant to serve the required load as cost-effectively as possible. The result of the optimization was the decision which conventional power plants produce electricity at which times.

Operation models are among the oldest models in the field of POWER SYSTEM modeling. Already in 1943, Steinberg and Smith [97] published a summary of related work. Some of the named contributions go back to the 1920s, e.g.,Davison [98]. Even if the task to solve remained the same for several decades, the methods used changed over the decades. Linear and non-linear optimization problems have formed the methodological foundation since the 60s at the latest. According to Wood et al. [99], great progress was made in the area during the decade between 1985 and 1995; improved methods and more powerful computers made it possible to integrate line losses and capacity limits into the formulation of optimization problems.

8 NATIONAL ENERGY MODELING SYSTEM

Following Wood et al. [99], the family of operation models can be further divided into the following subgroups: *Unit Commitment Models*, *Economic Dispatch Models* and *Optimal Power Flow Models*.[9] *Unit Commitment Models* address the so-called UCP:

> *Given that there are a number of subsets of the complete set of* $N_{\text{gen.}}$ *generating units that would satisfy the expected demand, which of these subsets should be used in order to provide the minimum operating cost? (Wood et al. [99])*

Economic Dispatch Models differ from *Unit Commitment Models* in that the subset of active generating units is fixed, but the optimal operating points of the individual power plants still have to be determined. From a traditional perspective, both model types do not include a model of the electric grid. In contrast, *Optimal Power Flow Models* typically link an *Economic Dispatch Model* with a model of the grid that also includes losses. They can also be regarded as a generalization and extension of the *Economic Dispatch Models*.

In a fully liberalized POWER SYSTEM, cf. Table 2, the classic UCP no longer exists due to the lack of a central planning authority, but it is still a central issue in a slightly modified form. E.g., in the US 'standard' market design, it is used to determine cost-optimal schedules from bids taking into account grid restrictions. In this context, it is known as SECURITY-CONSTRAINED UNIT COMMITMENT (SCUC). In addition, the UCP is used by generation companies to determine an optimal use of their power plant fleet.

Nevertheless, we formally introduce the UCP, as it is often used to model the electricity market and, therefore, the operation frequency of different types of power plants and the resulting electricity costs. An example for this usage is the PRIMES model, cf. Section 3.2.2.1; another case is the URBS-EU model, cf. Section 3.2.3.2.

To give an insight into the depth of modeling, we present a short primer on the UCP, as given by Carrion and Arroyo [100]. In a general form, UCP is described by:

$$\min \sum_{t\in\mathcal{T}} \sum_{j\in\mathcal{J}} \underbrace{c_j^{\mathrm{p}}[t]}_{\text{prod. costs}} + \underbrace{c_j^{\mathrm{u}}[t]}_{\text{startup costs}} + \underbrace{c_j^{\mathrm{d}}[t]}_{\text{shutdown costs}} \tag{3.1}$$

9 Unfortunately, Wood et al.'s nomenclature is not universal; in particular, the term *Unit Commitment Problem* is often used as an umbrella term for different types of operation models.

subject to

$$\sum_{j\in\mathcal{J}} p_j[t] \quad = \quad D[t],\ \forall t \in \mathcal{T} \tag{3.2a}$$

$$\sum_{j\in\mathcal{J}} \overline{p}_j[t] \quad \geq \quad D[t] + R[t],\ \forall t \in \mathcal{T} \tag{3.2b}$$

$$p_j[t] \quad \in \quad \Pi_j[t],\ \forall j \in \mathcal{J},\ \forall t \in \mathcal{T} \tag{3.2c}$$

with the following variables (constants respectively input variables are marked with 'fixed'):

$\mathcal{J}$	set of all conventional power plants,
$\mathcal{T}$	considered time span,
$c_j^{\mathrm{p}}[t]$	production costs of unit j in time period t,
$c_j^{\mathrm{u}}[t]$	startup costs of unit j in time period t,
$c_j^{\mathrm{d}}[t]$	shutdown costs of unit j in time period t,
$p_j[t]$	power output of unit j in time period t,
$D[t]$	load demand in time period t (fixed),
$\overline{p}_j[t]$	maximum available power of unit j in time period t,
$R[t]$	spinning reserve required in time period t (fixed),
$\Pi_j[t]$	region of feasible production of unit j in time period t.

The constraints given by the Equations 3.2 represent the condition of balancing production with consumption (Eqn. 3.2a), to provide spinning reserve (Eqn. 3.2b), and to operate the individual plants in a feasible region (Eqn. 3.2c); Equation 3.2c is a shortened representation of operation constraints of the individual plants, such as generation limits, ramping limits, and limited up and down times leading to an inclusion of binary decision variables into the UCP. The TIME PERIOD covered by $\mathcal{T}$ typically spans one day; a standard value for the time period t is 1 h or 15 min.

In this general representation, the UCP is a hard to solve non-linear mixed integer problem. Carrion and Arroyo [100] therefore reformulate it as a linear objective function, which is more efficient to solve. A piecewise linear function is used for the production costs c_j^{p}, which are generally assumed to be quadratic as a function of the power output; a mixed-integer linear formulation is used for the startup costs c_j^{u} to approximate the off-time dependent startup cost; and, the shutdown costs c_j^{d} are described with a constant value C_j for each shutdown event, formulated by means of a binary variable $v_j[t]$ representing the state (on/off) of the power plant:

$$c_j^{\mathrm{d}}[t] \quad \geq \quad C_j\left(v_j[t-1] - v_j[t]\right),\ \forall j \in \mathcal{J},\ \forall t \in \mathcal{T} \tag{3.3a}$$

$$c_j^{\mathrm{d}}[t] \quad \geq \quad 0, \forall j \in \mathcal{J}, \forall\ t \in \mathcal{T} \tag{3.3b}$$

In their work, Carrion and Arroyo use a formulation for the constraints (Eqn. 3.2) which, apart from the binary variables v_j, does not require any additional binary variables, and thus leads to a MIXED INTEGER LINEAR PROGRAMMING (MILP) formulation that can be solved efficiently without neglecting the basic properties of conventional power plants.

The formulation of the UCP as MILP is sufficiently efficient in terms of computation time for the primary use-case of the UCP; the calculation times are in the range of minutes. However, if a UCP is used to model the daily power plant operation within a holistic model of the POWER SYSTEM considering TIME INTERVALS of years, a linear form of the UCP without binary decision variables has to be used to keep the complexity of the holistic model at an acceptable level. However, this limitation inevitably requires neglecting state-dependent startup and shutdown costs as well as simplifying technical restrictions such as state-dependent load gradients or minimal downtimes and uptimes.

An example of this is the formulation used by the URBS-EU model, which is presented in the next section. The model neglects the terms $c_j^{\mathrm{u}}[t]$ and $c_j^{\mathrm{d}}[t]$ of Eqn. 3.1, and the operation constraints 3.2c are simplified to a constant ramping limit and a constant lower and upper operation limit.

3.2.3.2 *Hybrid Models, Example:* URBS-EU

Hybrid models of the POWER SYSTEM combine the characteristics of operation models as presented in the previous section with long-term planning models. Their primary application is similar to that of the long-term ENERGY SYSTEM optimization models, cf. Section 3.2.1, but the focus is on the electricity sector solely. Hybrid models are an answer to the challenge of *resolving details in space and time*, cf. Section 2.4.2.1, as they include the daily operation patterns influenced by the increasing share of VRE-based power plants into the long-term consideration to improve the quality of the latter.

The main advantage of the hybrid models is their level of detail regarding temporal and spatial resolution, and their modeling depth, allowing an in-depth insight into the optimized system, such as into the electricity price for a particular time step. The term *high resolution* must be seen in relation here. The models typically use time steps of one hour and the number of individually modeled *regions* is often below 100. If technical constraints of individual plants are modeled, they are modeled in a simplified way.

The main problem is the size of the optimization problem that scales with the number of time steps and regions. In combination with the long time periods of typically 50 years, the size of the models quickly reach a size hard to be solved even with high-performance systems. Several different approaches, cf. Cao et al. [101], to improve the performance of solvers or to reduce the size of the system without loosing accuracy exist, but no approach is capable of addressing the intrinsic problem.

The URBAN RESEARCH TOOLBOX: ENERGY SYSTEMS (URBS)-EU model was developed by Schaber [66] in her doctoral thesis to investigate the impact of VRE-based power plants to the grid on a European level and to make reliable statements about optimal pathways to ingrate them. The primary focus is on the effect of a European OVERLAY GRID and how it should be designed. For her work, Schaber uses the URBS model generator. Following the author, the *method* URBS was introduced in 2004 for the analysis of urban energy systems and later on applied to several research questions, especially in the area of power systems; Dorfner [102] further developed the method in his doctoral thesis. The URBS model generators and associated tools are available as open source implementations[10], whereas the models themselves, such as URBS-EU, are typically closed source.

URBS-EU is a linear optimization model with the optimization goal to minimize the total cost, which is defined as the sum of investment costs, fixed operation costs, and variable costs. For an overview, we list the inputs and outputs based on Schaber [66]:

Inputs

- Techno-economic parameters (costs, fuel, prices, efficiencies)
- Load and renewable supply profiles (per model region and time step)
- Existing infrastructure (per model region)
- Limits to infrastructure extensions (potential political circumstances)

Outputs

- Power plant dispatch and additions
- Grid capacity usage and extension
- Storage dispatch and extension
- Energy price and total costs
- Emission per region and source

To give an insight into the modeling depth of URBS-EU, we outline the characteristics of the model. It models the European POWER SYSTEM by means of 83 nodes, interconnected by transmission lines. Each node represents a region, and its generation and load. 50 of the nodes are onshore regions, 33 are offshore regions. The individual power plants are assigned to the respective next node and aggregated per technology and fuel. The hourly load values of the individual nodes are inelastic regarding the energy prices and are given as time series; a similar approach is used for the feed-in characteristic of VRE-based power plants. Hydro storage is included into the model. For each node, an energy balance guarantees that the consumption of the node is covered at all times by the generation at the node, or by energy flows through the lines connected to it. The capacity of the edges interconnecting the nodes represents the

10 See `https://github.com/tum-ens/urbs` (last accessed: 3 June 2021).

individual transmission lines. If the edges connect different countries, the data for transmission capacity is taken from ENTSO-E that publishes the NET TRANSFER CAPACITYS (NTCS). For edges representing lines within one country, data is obtained from publicly available grid data.

The modeling depth is typical for linear optimization models. Conventional power plants are described with the following set of equation:

$$p_j[t] = \eta_j \cdot p_j^{\text{fuel}}[t] \tag{3.4a}$$

$$p_j[t] \leq a_j \cdot c_j[t] \tag{3.4b}$$

$$|p_j[t] - p_j[t-1]| \leq \tau_j \cdot c_j[t] \tag{3.4c}$$

$$C_j^{\text{min.}} \leq c_j[t] \leq C_j^{\text{max.}} \tag{3.4d}$$

with the following variables (constants respectively input variables are marked with 'fixed'):

$\mathcal{J}$ set of all power plants,
$p_j[t]$ power output of unit j in time period t,
n_j efficiency factor of unit j (fixed),
$p_j^{\text{fuel}}[t]$ fuel consumption of unit j in time period t,
a_j availability factor of unit j (fixed),
$c_j[t]$ utilized capacity of unit j it time period t,
τ_j ramping restriction of unit j (fixed),
$C_j^{\text{min./max.}}$ operation limits of unit j (fixed).

$p_j[t]$ therefore represents the electrical energy produced by a specific type of power plant, such as a COMBINED CYCLE GAS TURBINE (CCGT), at a specific node within a given time interval t spanning one hour.

If a VRE-based power plant is represented by the given set of equations, Eqn. 3.4c is substituted by:

$$p_j[t] = cf_j[t] \cdot a_j[t] \cdot c_j[t] \tag{3.5}$$

with $cf_j[t]$ being the time-dependent capacity factor for VRE obtained from weather data.

Due to the reduced modeling depth of the technical restrictions, cf. UCP in Section 3.2.3.1, the model tends to overestimate the flexibility of conventional power plants –a problem that is also mentioned by the authors themselves. For example, the model allows a sluggish power plant, such as a nuclear power station, to be shut down for one hour, which is not feasible in reality.

The power flow between the nodes is "modeled as a transport problem, neglecting the nature of the AC load flow" (Schaber [66]). Specifically, Kirchhoff's first law is used to formulate that all power flows into one node sum up to zero. Transmission losses are assumed to be linear with the utilization of the line. This choice is based on findings from Ahlhaus and Stursberg [103], according to which the model errors due to this simplification can be accepted for the purpose of the model.

As typical for linear optimization problems, the energy market is not modeled explicitly. The dispatch of the power plants strictly follows the optimization function that determines the cost-optimal power production for each individual hour. Possible forecast errors with regard to the load and feed-in of VRES are not taken into account.

Although the URBS-EU model is used to make statements on an optimal pathway for the power system and includes investment decisions regarding grid extension and conventional power plants, only a temporal scope of one year is chosen. The investment costs are therefore integrated into the problem via their annuity. The respective annuity is determined using the WEIGHTED AVERAGE COST OF CAPITAL (WACC) approach; a technology-related depreciation period and a discount rate are assumed. In order to further reduce the number of time steps, the entire year is simulated with TIME SLICES to reduce the number of considered 'steps' to 1008.

Models of the type of hybrid models are in particular focus of further development, and the boundaries between power system optimization models and energy system optimization models are becoming increasingly blurred. For example, Schaber [66] develops a URBS-based model specifically for Germany that also includes the hydrogen as well as the heat sector to allow the investigation of research questions regarding sector coupling. Gils et al. [104], who use a RENEWABLE ENERGY MIX (REMIX)-based[11] model, called Remix-OptiMo, put a special focus on the extension of the power system with storage capacities.

3.2.4 *Power System Simulation Models*

The group of power system simulation models subsumes a large number of models with different purposes. Some of the models focus on the POWER SYSTEM in a holistic manner, e.g. the commercial tool PLEXOS[12], bringing them close to energy system simulation models (cf. Section 3.2.2). However, most of the models address specific aspects of the POWER SYSTEM. From this group, we would like to outline two model types: agent-based energy market models, cf. Section 3.2.4.1, and ordinary grid simulation models, cf. Section 3.2.4.2.

11 REMIX is a closed-source modeling framework maintained by the DEUTSCHES ZENTRUM FÜR LUFT- UND RAUMFAHRT (GERMAN AEROSPACE CENTER) (DLR), see `https://www.dlr.de/tt/en/desktopdefault.aspx/tabid-2885/4422_read-12423/` (last accessed: 3 June 2021) and Scholz [105]

12 See `https://energyexemplar.com/software/plexos-desktop-edition/` (last accessed: 3 June 2021).

3.2.4.1 *Energy Market Model, Example: Agent-based Modeling*

The liberalization of the energy markets in the late 1990s and 2000s led to the question of how the energy markets should be designed [106]. This question was and is mainly addressed with a method that was in focus at the same time, namely *agent-based modeling and simulation*. As MOCES itself also uses an agent-based modeling and simulation approach, we will explain this method in more detail in Section 3.4. Overviews of the research activities in the context of POWER SYSTEMS are given by Weidlich and Veit [107], Guerci et al. [106], and Ringler et al. [108]. While Weidlich and Veit and Guerci et al. focus on agent-based modeling for the interaction at the wholesale electricity markets, Ringler et al. have a somewhat broader focus, also addressing agent-based modeling and simulation for *prosumer behavior* and *decentralized structures*.

AGENT-BASED MODELING OF ELECTRICITY SYSTEMS (AMES) is an open-source implementation of a U.S. style wholesale market. The primary focus of the tool is to allow the investigation of the "complex interplay among structural conditions, market protocols, and learning behaviors in relation to short-term and longer-term market performance" (Sun and Tesfatsion [109]). The initial version of AMES was published in 2007 [109], and the tool is still maintained by the *Iowa State University*.[13] Regarding the modeling of the wholesale market, in its core, AMES solves a UCP. Generators and loads act as agents on the market, and the modeler can define a static or self-learning strategy for the agents.

Based on AMES, Veit et al. [110] investigate the German wholesale market. Their focus is on the effect of the increasing share of wind power plants and the corresponding limitations regarding the grid, especially regarding the capacity of the transmission lines connecting the individual zones of the four German TSOs. A total of six different zones are considered, and the connections to foreign markets are included in a simplified way assuming a fixed exogenous price.

Conventional power plants, hydroelectric power plants, wind power plants, and loads are modeled as agents interacting at the wholesale market. Only the agents representing the conventional power plants implement a self-learning bidding strategy, all other agents are using a predetermined and static bidding strategy. The loads, for example, are assumed to be price-insensitive and thus follow a "buy at any price" strategy.

For the learning strategy of the conventional power plants, Veit et al. use a common approach based on the reinforcement learning model developed by Erev and Roth [111]. The basic idea is that the individual agents learn an optimal strategy based on their experience in the past.

13 See `http://www2.econ.iastate.edu/tesfatsi/AMESMarketHome.htm` (last accessed: 3 June 2021).

In this specific case, the individual agent j learns to select the best offer curve n from a set of $|\mathcal{N}| = 16$ predefined offer curves. The elements within the set of offer curves range from a curve that represents the set-point dependent MARGINAL COSTS in a linear way to a curve four times as high. This offer curves resemble the production costs $c_j^p[t]$ of the UCP, cf. Section 3.2.3.1.

In the initial round $t = 0$, the propensities $q_{j,n}[t]$ to select one specific offer curve k out of the set $\mathcal{N}$ are equal for all possible offer curves ($q_{j,n}[0] = q_{j,n+1}[0] = \ldots = q_{j,|\mathcal{N}|}[0]$). In each round t, the propensities of the offer curves are updated depending on the profit $\text{prf}_{j,k}[t]$ earned with the selected offer curve in this round:

$$q_{j,n}[t+1] = (1-\phi) \cdot q_{j,n}[t] + r_{j,n}, \tag{3.6a}$$

$$r_{j,n} = \begin{cases} (1-\varepsilon) \cdot \text{prf}_{j,k}[t] & \text{if } k = n \\ \frac{\varepsilon}{|J|-1} \cdot q_{j,n}[t] & \text{if } k \neq n \end{cases}. \tag{3.6b}$$

These propensities are used to determine the probabilities to choose an offer curve:

$$p_{j,n}[t] = \frac{e^{q_{j,n}[t]/c}}{\sum_{n \in \mathcal{N}} e^{q_{j,n}[t]/c}}. \tag{3.7}$$

The individual parameters $\phi([0,1])$, $\varepsilon([0,1])$, and $c((0,\infty))$ represent specific aspects of the learning strategy. ϕ reduces the importance of experience in the past. ε was introduced by Erev and Roth [111] as experimentation parameter to update the selected as well as similar strategies. In the present modification (Eqn. 3.6), all strategies are assumed to be equally dissimilar; the parameter therefore loses its meaning in this case. c effects the speed of convergence.

The authors use the developed and parameterized model to investigate a total of eight different scenarios. They combine the following three factors: grid capacity limitations vs. no limitation, high vs. low wind power feed-in, and strategic vs. no strategic behavior. No strategic behavior means that the agents representing the conventional power plants place an offer with their MARGINAL COSTS. The results are hardly surprising and qualitatively consistent with expectations: a higher share of wind power reduces wholesale market prices, and the restrictions on transmission capacity lead to different prices in the respective zones.

The third finding is that if agents representing conventional power plants follow a strategic behavior, the market prices for energy increase and social welfare decreases. Ultimately, this last insight shows that the agent-based approach is suitable for describing the actual behavior of actors in the energy market. They learn to exploit their market power, which in this case is based in particular on the fact that the demand is inflexible in terms of price. Absurdly, this modeling of market power, which is actually correct, would lead to a false statement about the actual real-world development of electricity prices. This is because such a behavior learned by the agents is classified as abusive due to their market power by the antitrust authorities and is, therefore, not accepted.[14]

Agent-based modeling and simulation is repeatedly mentioned as a key idea for correctly mapping not only the complexity of the electricity market but also that of the entire electricity system, e.g. by Pfenninger et al. [4] or Dam [85]. EMCAS[15] and GridLab-D[16] are examples of tools using an agent-based approach.

3.2.4.2 *Grid Simulation, Example: Power Flow Calculation*

All modeling approaches presented so far have largely simplified the grid itself and the machines connected to it; dynamic phenomena, such as machine dynamics, have not been taken into account. Moreover, a single variable, representing the active power, has usually been used to model the behavior of a machine as a function of time. Grid simulation tools, also referred to as power system analysis tools, focus on these topics at the expense of neglecting other characteristics of the POWER SYSTEM, such as the energy market or the user behavior. In a grid simulation tool, such characteristics are external and are represented in the parameters of the boundary nodes, such as a PQ bus used to model a simple load. With reference to the spectrum of modeling and simulation, cf. Figure 2, it can be noted that grid simulation models are placed on the right edge of the spectrum; the systems are very well understood and the models are suitable for design purposes.

An overview of the different dynamic processes in the network is given in Figure 26. It shows a multitude of different phenomena on different time scales. Transient processes in the grid itself, shown in red in the figure, have the smallest time constants. These are electromagnetic transients and traveling waves which are triggered, e.g., by switching on an electric motor. The time constants of the dynamics of the machines connected to the network, shown in blue in the illustration, are substantially larger. Significant for the dynamics of the machines are in particular the electromechanical transients of the turbine-generator group and the dynamics caused by the used controllers.

14 The German Federal Cartel Office classifies the financial withholding of capacity as abusive behavior, cf. [112, p.22].

15 EMCAS, short for Electricity Market Complex Adaptive System, see `https://ceeesa.es.anl.gov/projects/emcas.html` (last accessed: 3 June 2021) and Conzelmann et al. [113], was developed by the *Argonne National Laboratory* but it is not supported anymore.

16 GridLab-D is maintained by the *U.S. Department of Energy*, see `https://www.gridlabd.org/` (last accessed: 3 June 2021) and Chassin et al. [114].

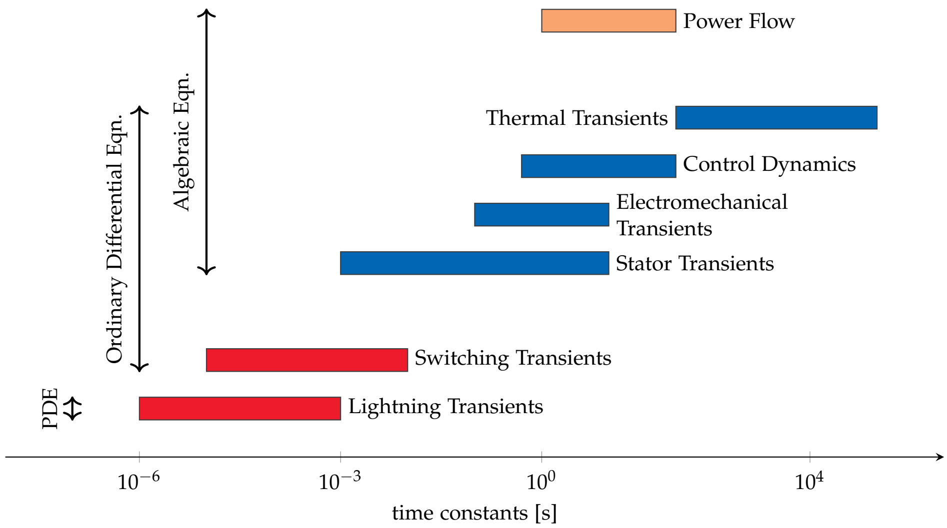


Figure 26: Dynamic phenomena in grids and controlled machines, Figure adapted from Andersson [115, p. 81] and Marenbach et al. [116, p. 386].

For most investigations, it is not necessary to consider the electromagnetic processes and, therefore, a static model of the grid using algebraic equations solely is used. This static analysis of the grid is known under the term *power flow analysis* or *load flow study* and is a basic functionality of grid calculation tools such as as PANDAPOWER[17]. Since MOCES allows for the modeling of the grid by a static model, we will briefly introduce the power flow model, based on the formulation by Andersson [115]. The power flow model considers the grid as a set of nodes $n \in \mathcal{N}$, also called buses, connected by branches $b_{n_1,n_2} \in \mathcal{B}$; these branches represent the lines. Each node n is associated with four variables:

U_n voltage magnitude at node n,
θ_n voltage angle of node n,
P_n active power feed into node n,
Q_n reactive power feed into node n.

Depending on the type of node, two of the four variables are known. If P and Q are given, we speak of a PQ bus, if P and U are given, of a PU bus. Which type is used, depends on the technical system described by the bus: PQ buses are used to model loads, PU buses for generators.

17 PANDAPOWER is an open-source power system analysis tool maintained by *Fraunhofer IEE* and the *University of Kassel*, see `https://pandapower.readthedocs.io` (last accessed: 3 June 2021).

The load flow model is obtained by formulating the node equations, according to which all power flows into the node n through connected branches ($P_{n,m}, Q_{n,m}$) and injected at the node (P_n, Q_n) sum up to zero:

$$P_n = \sum_{m \in \mathcal{M}_n} P_{n,m}\left(U_n, U_m, \theta_n, \theta_m\right), \tag{3.8a}$$

$$Q_n = \sum_{m \in \mathcal{M}_n} Q_{n,m}\left(U_n, U_m, \theta_n, \theta_m\right). \tag{3.8b}$$

with $\mathcal{M}_n$ being the set of all nodes connected to the n-th node by a branch.

Unfortunately, the functions $P_{n,m}(\cdot)$, $Q_{n,m}(\cdot)$ have a non-linear form; even for simple network elements like transmission lines modeled with a π-model and neglecting power losses, we get [115, p. 21]:

$$P_{n,m} = P_{m,n} = -b_{n,m} U_n U_m \sin\left(\theta_n - \theta_m\right), \tag{3.9a}$$

$$Q_{n,m} = -U_n^2 (b_{n,m} + b_{n,m}^{sh}) + U_n U_m b_{km} \cos\left(\theta_n - \theta_m\right), \tag{3.9b}$$

$$Q_{m,n} = -U_m^2 (b_{n,m} + b_{n,m}^{sh}) + U_n U_m b_{km} \cos\left(\theta_n - \theta_m\right) \tag{3.9c}$$

with $b_{n,m}$ and $b_{n,m}^{sh}$ representing the series respectively the shunt susceptance of the transmission line. Like the UCP, this non-linearity has brought generations of scientists into wage and bread by developing methods to formulate and solve the systems of equations as efficiently as possible. A further discussion should be dispensed with, but we would like to summarize that ultimately there is a non-linear system of algebraic equations of the following form:

$$\mathbf{0} = \mathbf{f_1}\left(\mathbf{y_1}, \mathbf{m_1}, \mathbf{p_1}\right), \tag{3.10}$$

with:

- $\mathbf{y_1}$ vector of unknown variables,
- $\mathbf{m_1}$ vector of given variables,
- $\mathbf{p_1}$ vector of parameter.

If a synchronous generator is part of the model and is described as a *PU* bus, its active power P as well as its voltage magnitude U is part of the known variables $\mathbf{m}$.

When the stability of the network is to be investigated, static modeling of the system is no longer sufficient. Following a definition of the IEEE, the term *stability* is to be understood as follows:

> [...] stability is the ability of an electric power system, for a given initial operating condition, to regain a state of operating equilibrium after being subjected to a physical disturbance, [...] so that practically the entire system remains intact. (Andersson [115])

It is, thus, an umbrella term for several different effects; Andersson [115] distinguishes between *rotor-angular*, *frequency*, and *voltage* stability. Frequency stability addresses the task to maintain a grid-wide balance between active power fed into the grid and consumed power. We have already discussed this task from the perspective of the responsible ROLE (cf. Section 2.2.7.1, p. 50), the market processes, (cf. Section 2.2.8.2, p. 57), and the related markets (cf. Section 2.2.9.2, p. 63) in Chapter 2. Since MOCES supports the modeling of frequency stability, we will introduce its mathematical formulation.

Modeling the frequency dynamic requires the inclusion of the dynamic electromechanical phenomena taking place in the synchronous machine of the CONVENTIONAL POWER PLANTS and the turbine. Their behavior can be described using a set of differential algebraic equations with the swing equation in its core, [115, p. 104]:

$$\omega_m J \frac{d^2\theta_m}{dt^2} = P_m - P_e, \tag{3.11}$$

with:

- ω_m mechanical angular velocity,
- J inertia of the machine,
- θ_m mechanical angle of rotor,
- P_m mechanical power on the rotor of the machine,
- P_e electrical power on the rotor of the machine.

Together with a set of equations describing the generator machine, the equations finally form a DIFFERENTIAL ALGEBRAIC EQUATION (DAE) . Combined with the equations describing the grid, Eqn. 3.10, both sets of equations form a system of equations that needs to be solved in order to describe the dynamic behavior of the grid. Although general approaches exist to directly process the resulting system of equations, cf. Section 3.3.1, corresponding tools use optimized simulation techniques. In principle, the two systems of equations are considered separately. In a cyclical approach, first, a solution for 3.10 is determined, and then used as a fixed solution for the equations describing the dynamic of the grid, e.g. Equation 3.11, and so forth.[18]

3.3 MODELICA, A CANDIDATE FOR A MODELING LANGUAGE FOR ENERGY AND POWER SYSTEMS

The term *modeling language* is not precisely defined and describes any type of artificial language that allows information about a system to be

18 See `https://www.slideshare.net/luigivanfretti/wanted-open-ms-standards-and-technologies-for-the-smart-grid-introducing-rapid-and-ipsl-oss-tools-for-power-system-model-simulation-and-model-validation-from-the-fp7-itesla-project` (last accessed: 3 June 2021) for a short introduction.

expressed in a structured manner [117]. We would like to narrow down the term considerably:

> *A modeling language is an artificial, human- and machine-readable language with which the dynamic behavior of a* SYSTEM *can be unambiguously described. It is only a modeling language if at least one interpreter (simulator) exists that can determine the behavior solely from the –potentially automatically transformed–* MODEL*.*

Quite annoyingly, there is no such thing as *the one* holistic MODELING LANGUAGE that is widely used by the community for the modeling of energy and power systems. All of the tools and frameworks mentioned in the previous Section 3.2 use different modeling languages.

MODELICA is a frequently discussed candidate for such a holistic modeling language [e.g. 118, 119]. We use it for MOCES to enable a straightforward modeling and simulation of power systems. In this section, we therefore outline its main principles (Sections 3.3.1 and 3.3.2), discuss challenges (Section 3.3.3) but finally justify the choice (Section 3.3.4) and provide an overview over related work (Section 3.3.5).

3.3.1 *Modelica Language Elements*

MODELICA is a declarative modeling language for complex systems, maintained by the *Modelica Association*.[19] It follows an *acausal modeling* approach, cf. Section 1.6 for further information. The MODELICA LANGUAGE SPECIFICATION (MLS) [120] defines its syntax, that can be used to describe MODELS in a component-oriented manner. To briefly and compactly describe MODELICA, we refer to the abstract of the MLS:[20]

> *Modelica is a freely available, object-oriented language for modeling of large, complex, and heterogeneous systems. It is suited for multidomain modeling, for example, mechatronic models in robotics, automotive and aerospace applications involving mechanical, electrical, hydraulic control and state machine subsystems, process oriented applications and generation and distribution of electric power. Models in Modelica are mathematically described by differential, algebraic and discrete equations. No particular variable needs to be solved for manually. A Modelica tool will have enough information to decide that automatically. Modelica is designed such that available, specialized algorithms can be utilized to enable efficient handling of large models having more than one hundred thousand equations. Modelica is suited and used for hardware-in-the-loop simulations and for embedded control systems. (Modelica Association [120])*

19 See `https://modelica.org` (last accessed: 3 June 2021) for further information.
20 Doing so is inspired by Liu [121].

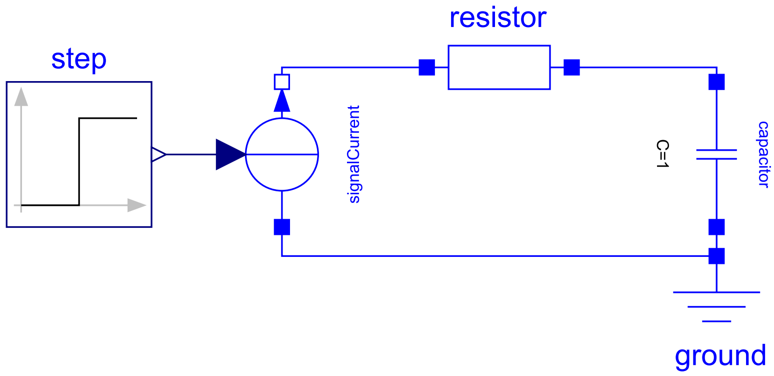


Figure 27: MODELICA diagram representing an RC circuit.

For a detailed description of MODELICA, please refer to the language specification [120], or to Fritzson [14]. Here, we dispense with a detailed description of MODELICA, and instead illustrate the basic language constructs with an example. For this, we refer back to the RC circuit with ideal current source already introduced in Section 1.6, Figure 6b, which is extended by a signal source controlling the ideal current source. The corresponding MODELICA diagram is shown in Figure 27, and listing 3.1 provides the corresponding MODELICA code.

As can be seen from the code lines 27–32, the `Capacitor` is described in a straightforward way by a differential equation, $i = C\frac{du}{dt}$[21], as a function of the voltage drop u over the capacitor, the current of the element i and the time-invariant value for C, declared as known a priori by the keyword `parameter`. The variables as well as basic algebraic equations are inherited from the base class `OnePort`, cf. lines 8-17, by the keyword `extends`. This base class, from which the model `Resistor` also inherits, is marked with the keyword `partial`. It indicates that the model is not balanced and, therefore, cannot be instantiated. In MODELICA, a `model` has to be balanced, meaning that the number of equations has to be equal to the number of unknowns.

The partial model `OnePort` includes two instances of an electrical `Pin` which represent the interface of the model. The definition of the `Pin`, cf. lines 3–6, uses the keyword `connector` which refers to a fundamental concept in MODELICA. `connectors` describe interfaces of models and do *not* impose a CAUSAL DIRECTION. They thus allow a linkage between models introduced as *powerbonds* in Section 1.6. Accordingly, a `connector` is typically described by an *effort* variable (cf. line 4) and a *flow* variable (cf. line 5). To declare a variable as `flow` variable, the corresponding keyword is used.

---

21 In the code, we use `v` for the electric potential because the MODELICA STANDARD LIBRARY (MSL), from which the used base models were taken, does this.

```
package ModelicaExample
  import SI = Modelica.SIunits;
  connector Pin "Pin of an electrical component"
    SI.Voltage v "Potential at the pin";
    flow SI.Current i "Current flowing into the pin";
  end Pin;

  partial model OnePort "Component with two el. pins p and n"
    SI.Voltage v "Voltage drop between the two pins (= p.v - n.v)";
    SI.Current i "Current flowing from pin p to pin n";
    Pin p;
    Pin n;
  equation
    v = p.v - n.v;
    0 = p.i + n.i;
    i = p.i;
  end OnePort;

  model Resistor "Electrical resistor"
    extends OnePort;
    SI.Resistance R_actual "Actual resistance";
  equation
    R_actual = R_func(time);
    v = R_actual*i;
  end Resistor;

  model Capacitor "Ideal linear electrical capacitor"
    extends OnePort(v(start=0));
    parameter SI.Capacitance C=1 "Capacitance";
  equation
    i = C*der(v);
  end Capacitor;

  model RC "Model of RC circuit"
    Ground ground;
    Resistor resistor;
    SignalCurrent signalCurrent;
    Capacitor capacitor;
    Modelica.Blocks.Sources.Step step;
  equation
    connect(ground.p, capacitor.n);
    connect(resistor.n, capacitor.p);
    connect(resistor.p, signalCurrent.n);
    connect(signalCurrent.p, ground.p);
    connect(step.y, signalCurrent.i);
  end RC;

  function R_func "Algorithm to determine R"
    input Real x;
    output Real y;
  algorithm
    y := 0;
    if (x > 0.5) then
```

```
54        y:= y+1;
55      else
56        y:=x;
57      end if;
58    end R_func;
59
60    block Step "Generate step signal of type Real"
61      extends Modelica.Blocks.Interfaces.SignalSource;
62    equation
63      when time > 0.5 then
64        y = 0.5;
65      end when;
66    end Step;
67  end ModelicaExample;
```

Listing 3.1: Modelica code of the RC circuit. Model of `ground` and `signalCurrent` are not shown.

The lines 34–46 put everything together and describe the overall model of the RC circuit. In its first part (lines 35-39), the models that make up the RC circuit are instantiated; in the second part, the `equation` part, the model instances are linked by `connect` statements referring to their connectors. A `connect` statement is syntactic sugar and is later expanded to a set of equations based on the information given by the `connectors`. In general: potentials are equated, flow quantities sum up to zero.

As indicated by the keyword `equation`, the single statements of the models are equations and no assignments as known from ordinary programming languages. The ordering of the equations is thus irrelevant. In contrast to that stand algorithms that can also be used to describe the behavior of models. For this purpose, the corresponding code block must be marked with the keyword `algorithm`, see lines 51–57 for an example. In the present typical application case, the algorithm is part of a `function` with defined input and output variables. If the function is part of a model, cf. line 23, it must be ensured that it is a mathematical function, also referred to as a pure function. impure functions can only be used under special restrictions, cf. discussion in Section 4.4.7.

The last language element we want to point out is the `when` element, cf. line 63. It is one of the language elements that enable the modeling of hybrid systems. The `when`-statement triggers an event when the condition associated with it, `time > 0.5` in this case, becomes true. The equations associated with the `when`-statement, $y = 0.5$ in this case, apply only at the point in time associated with the event, cf. Section 4.4.7 for a more detailed discussion. `time` is a built-in variable which is treated as an input variable. It represents the simulation time.

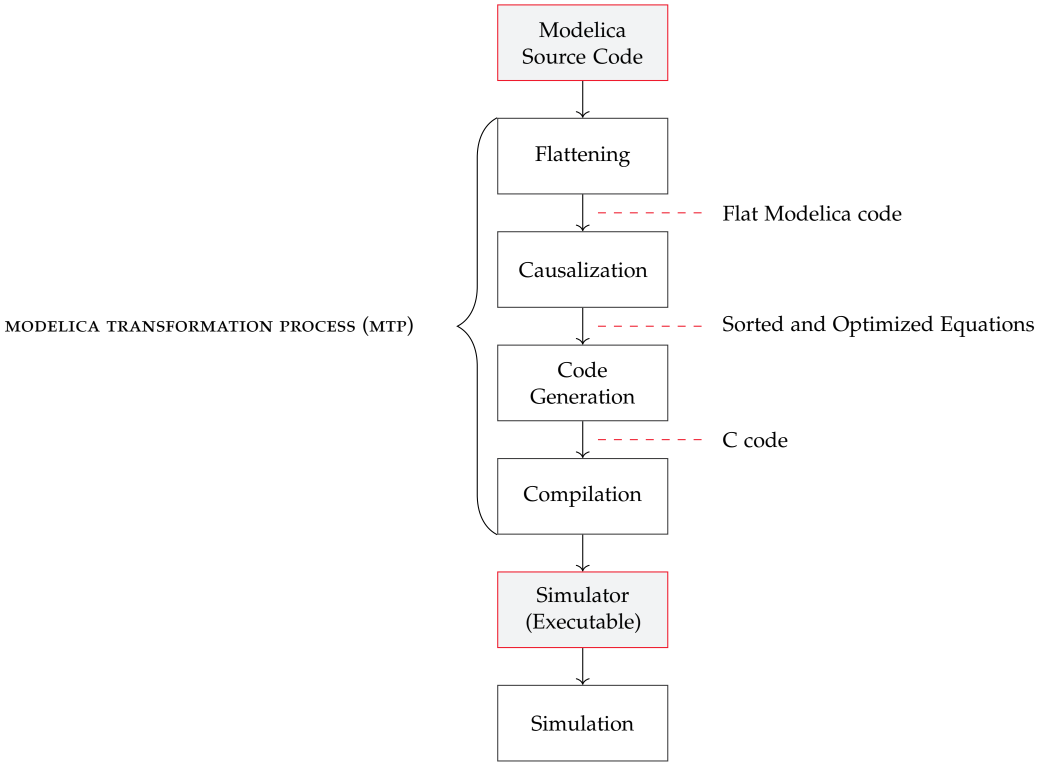


Figure 28: Overview of the standard MODELICA MODELING AND SIMULATION PROCESS (MMSP) and its core, the MTP, used to transform MODELICA source code into an executable that is capable of performing a simulation of the model. Figure is adapted from Fritzson [14].

3.3.2 *Classic Modelica Modeling and Simulation Process*

In the previous chapters, we have described MODELICA as a possibility to formulate MODELS of complex systems. As MODELICA is a declarative language, the specification does not have to specify how these MODELS should be handled by a SIMULATOR to determine the time-dependent behavior of the MODEL rather to translate the MODEL into a SIMULATOR. However, MODELICA has been developed with the aim of fully automating this step. The sub-steps of the classic procedure, with its core named MTP, are given in Figure 28 which fits into the general modeling and simulation chain given in Figure 3. The ability to perform a translation of a MODEL into a SIMULATOR is the main functionality of MODELICA IDES; a dissimilar performance in the translation of large complex models is the main difference between different MODELICA IDES.

The following description of the steps of the MTP is based on Frenkel [122], Cellier and Kofman [17], and Braun et al. [123].

1. FLATTENING: The first step of the MTP is the translation of the highly hierarchical Modelica model into the so-called flat Modelica model. This step includes expanding the inheritances, replacing the connect equations, and mapping when-clauses into equations. Further on, some simple optimization steps such as removing of alias variables may be performed. An excerpt of the flat Modelica model of the RC circuit, cf. Figure 27, is given in the following listing 3.2:

```
model ModelicaExample.RC
parameter Modelica.SIunits.Capacitance capacitor.C = 1 "Capacitance";
[..]
Modelica.SIunits.Current capacitor.n.i "Current flowing into the pin";
[..]
  // Component step
  // class Modelica.Blocks.Sources.Step
  equation
    step.y = step.offset+(if time < step.startTime then 0 else step.height);
[..]
  equation
    capacitor.n.i+ground.p.i+signalCurrent.p.i = 0.0;
    ground.p.v = capacitor.n.v;
[..]
end ModelicaExample.RC;
```

Listing 3.2: Excerpt of a flat model Modelica model.

Flattening does not involve any critical step concerning scalability. It should be noted that information on the hierarchical structure of the model, which may be useful for the translation process, is partially lost after this step.

2. CAUSALIZATION: The output of the flattening step is a HYBRID DIFFERENTIAL ALGEBRAIC EQUATION (HDAE) system, cf. Appendix A.1 for the full Modelica HDAE formulation. An HDAE system can be formulated as a set of DAE systems, each representing a mode of the HYBRID SYSTEM, and conditions that trigger switching between the modes (Mao and Petzold [124]). This characteristic allows us to neglect the hybrid part in the following, and the result of the flattening step is, therefore, a DAE in implicit form:

$$\mathbf{f}\left(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t\right) = \mathbf{0}, \tag{3.12a}$$

$$\mathbf{x}(t_0) = \mathbf{x_0} \tag{3.12b}$$

with the vector of dynamic variables $\mathbf{x}$ to which the differential operator is applied and the vector of algebraic variables $\mathbf{y}$. The final goal of the causalization step is to transform the DAE in implicit form into an ORDINARY DIFFERENTIAL EQUATION (ODE) in explicit form:

$$\dot{\mathbf{x}}^* = \hat{\mathbf{f}}\left(\mathbf{x}^*, t\right), \tag{3.13a}$$

$$\mathbf{x}^*(t_0) = \mathbf{x_0}^* \tag{3.13b}$$

and to find a way to calculate $\hat{\mathbf{f}}(\cdot)$ efficiently. Please notice that $\mathbf{x}^*$ represents the state variable and may differ from $\mathbf{x}$. A model formulated in state-space form, cf. Eqn. 3.13, allows using ODE solvers, also known as integration algorithms, to perform a simulation of the model.

2A. SORTING To formulate the ODE system, the equations of the DAE must be sorted. The aim is to transform the system into the BLOCK LOWER TRIANGULAR (BLT) form. Ideally, $\dot{\mathbf{x}}^*$ can then be determined by a sequence of assignments without involving any numerical solver.

To outline the BLT sorting, we introduce the *structure incidence matrix* $\mathbf{S} = s_{i,j} \in \{0,1\}^{m \times m}$, with m being the number of unknowns or equations. The matrix represents the structure of the equation system by listing the equations as rows and the unknown variables $\mathbf{v} := [\dot{\mathbf{x}}, \mathbf{y}]$ as columns. If the j-th variable v_j appears in the i-th equation f_i, $s_{i,j} = 1$, otherwise $s_{i,j} = 0$:

$$\mathbf{S} = \begin{matrix} & v_1 & v_2 & \dots & v_m \\ f_1 & 0 \veebar 1 & 0 \veebar 1 & \dots & 0 \veebar 1 \\ f_2 & 0 \veebar 1 & 0 \veebar 1 & \dots & 0 \veebar 1 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ f_m & 0 \veebar 1 & 0 \veebar 1 & \dots & 0 \veebar 1 \end{matrix} . \tag{3.14}$$

Ideally, by permuting rows and columns, an incidence matrix $\mathbf{S}^*$ is obtained which has the form of a lower triangular matrix:

$$\mathbf{S}^* = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 \veebar 1 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 \veebar 1 & \dots & 0 \veebar 1 & 1 \end{pmatrix} . \tag{3.15}$$

Calculating the unknown variables $\mathbf{v}$ is straightforward in this case, as we have made the equations causal by sorting them horizontally and have brought them into an executable sequence by sorting them vertically [17, p. 259].

A common approach to perform the permutation is to apply Tarjan's algorithm [125] which uses a bipartite graph G to represent the structure incidence matrix:

$$G = (\mathcal{F}, \mathcal{V}, \mathcal{E})\,, \tag{3.16a}$$
$$\mathcal{F} = \{f_1, f_2, \ldots, f_m\}\,, \tag{3.16b}$$
$$\mathcal{V} = \{v_1, v_2, \ldots, v_m\}\,, \tag{3.16c}$$
$$\mathcal{E} \subseteq (\mathcal{F} \times \mathcal{V})\,. \tag{3.16d}$$

$\mathcal{E}$ is the set of edges representing the connections between the functions and the variables. Fortunately, the algorithm developed by Tarjan is efficient, and its complexity grows in the worst case linearly with the product of edges and vertices: $\mathcal{O}(|\mathcal{V}| \cdot |\mathcal{E}|)$, [17, p. 259]. Following Frenkel [122, p. 33], heuristics reduce the complexity mostly to $\mathcal{O}(|\mathcal{V}| + |\mathcal{E}|)$. The memory requirement is quoted by Frenkel [122, p. 33] with $2|\mathcal{V}| + |\mathcal{E}|$.

Unfortunately, not all DAES can be transformed into the shape of a lower triangular matrix. In this case, further modifications are either required for performance reasons or even mandatory to perform a simulation at all. *Tearing* and *relaxation* are examples for the first group, *index reduction* for the latter.

2B. TEARING AND RELAXATION Depending on the structure of the modeled system and the used mathematical formulation, the BLT sorting, respectively Tarjan's algorithm, might end with an incidence matrix with a block lower triangular form:

$$\mathbf{S}^{\dagger} = \begin{array}{c} \\ f_1 \\ \mathbf{f}_{2\ldots} \\ \vdots \\ \mathbf{f}_{\ldots m-1} \\ f_m \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & \mathbf{v}_{2\ldots} & \ldots & \mathbf{v}_{\ldots m-1} & v_m \end{array} \\ \begin{pmatrix} 1 & \mathbf{0} & \ldots & \mathbf{0} & 0 \\ \mathbf{0} \veebar \mathbf{1} & \mathbf{A}^* & \ldots & \mathbf{0} & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{0} \veebar \mathbf{1} & \mathbf{0} \veebar \mathbf{1} & \ldots & \mathbf{A}^{\dagger} & \mathbf{0} \\ 0 \veebar 1 & \mathbf{0} \veebar \mathbf{1} & \ldots & \mathbf{0} \veebar \mathbf{1} & 1 \end{pmatrix} \end{array}, \tag{3.17}$$

with

$$\mathbf{A}^*, \mathbf{A}^{\dagger} = \begin{pmatrix} 0 \veebar 1 & \ldots & 0 \veebar 1 \\ \vdots & \ddots & \vdots \\ 0 \veebar 1 & \ldots & 0 \veebar 1 \end{pmatrix}. \tag{3.18}$$

Consequently, the unknown variables can no longer be determined easily, as the blocks $\mathbf{A}^*$ and $\mathbf{A}^{\dagger}$ represent strongly linked components known as algebraic loops. Without further modification, $\mathbf{A}^*$ and $\mathbf{A}^{\dagger}$ represent linear or non-linear subsystems of equations which can in principle be solved numerically using appropriate methods, such as Newton's method; this must then be done in each integration step. It is therefore desirable to symbolically optimize the subsystems to improve the simulation performance. Methods that can be used for this are known as tearing and relaxation.

The basic idea of *tearing* is to reduce the number of equations which have to be handled by Newton's method by making an assumption about one or several variables, named tearing variables. The *relaxation* algorithm can be applied to linear systems solely but is capable of solving the problem with an elimination process. For a description of both methods, please refer to Cellier and Kofman [17]. The crucial point here is that tearing and relaxation are hard problems. According to Cellier and Kofman [17], the computational effort to find an optimal tearing variable grows exponentially with the number of equations forming the algebraic loop.

2C. INDEX REDUCTION Index reduction is mandatory if the DAE under consideration is a higher-INDEX DAE that, therefore, contains structural singularities. In this case, Tarjan's algorithm fails and indicates the existence of a higher-index system. The algorithm mainly used for index reduction is that of Pantelides [126]. Again, for an introduction see Cellier and Kofman [17].

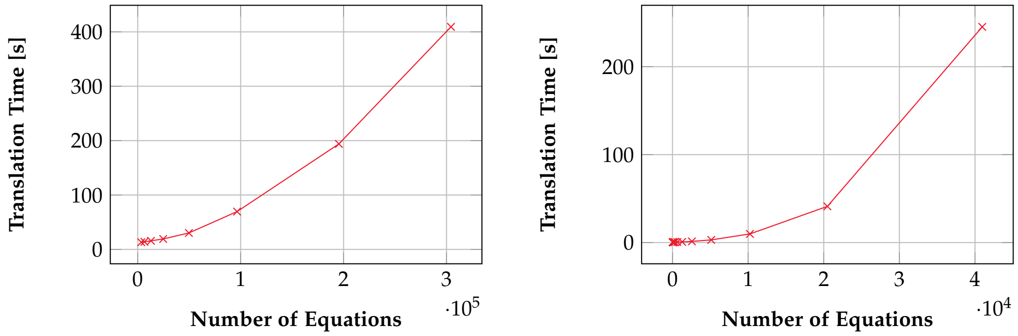
3. CODE GENERATION In the code generation process, the model is converted into source code, that can be compiled by a standard compiler and linked with an available ODE or DAE SOLVER, usually with an implementation of a DIFFERENTIAL ALGEBRAIC SYSTEM SOLVER (DASSL). All MODELICA IDES known to the author use C as output language.

4. COMPILATION Standard C compilers are used to generate the SIMULATOR.

5. SIMULATION The exact sequence of the SIMULATION depends on the SIMULATOR as well as the selected SOLVER and its features. A high-level representation of the sequence is given later in Figure 51, which also includes the not yet discussed treatment of EVENTS.

For the simplest case, a model in state-space form and a SOLVER, respectively a SIMULATOR using an explicit integration algorithm, we would like to sketch the interaction between SOLVER and model based on Cellier and Kofman [17, p. 320]. This interaction can be found in the steps marked in blue in Figure 51.

(1) Use the state-space model, cf. Eqn. 3.13, to compute $\dot{\mathbf{x}}^*(t_1)$ out of the known state variables $\mathbf{x}^*(t_1)$ for the point in time t_1,



(a) Translation time for different sizes of the `DistributionSystemModelica-ActiveLoads` model that is part of the `ScaleableTestSuite` [128], DYMOLA 2018, standard PC of 2017.

(b) Translation time for a flat model, data adapted from Frenkel et al. [127], DYMOLA 7.4, standard PC of 2011.

Figure 29: MODELICA translation time for models of different sizes.

(2) use an integration algorithm to compute the state variables for $t_2 = t_1 + \Delta t$ out of the known state variables and their known derivatives for t_1. If using a simple Euler Forward, this is just: $\mathbf{x}^*(t_2) = \mathbf{x}^*(t_1) + \Delta t \dot{\mathbf{x}}^*(t_1)$,

(3) proceed with (1) until end time is reached.

3.3.3 *Performance Problems and Alternative Processes*

The classic MTP (Section 3.3.2) is sufficiently efficient for many purposes. However, studies by Frenkel et al. [127] have also shown that the translation time increases significantly more than linearly with the number of equations. Figure 29b gives exemplary values determined by Frenkel et al. [127] for a flat model; Figure 29a shows own results based on a test suite for large models provided by Casella [128]. The values are to be understood as a rough guideline since the required translation time depends heavily on the structure of the model.

Frenkel et al.'s investigations have also shown that the translation fails above a certain model size in the various MODELICA IDES which implement the classic MTP. The commercial MODELICA IDE DYMOLA turned out to be the best performing IDE among the options compared: DYMOLA[22], OPENMODELICA[23], SIMULATIONX[24] and JMODELICA[25]. For DYMOLA (ver-

22 Commercial IDE by DASSAULT SYSTEMS, see `https://www.3ds.com/de/produkte-und-services/catia/produkte/dymola/` (last accessed: 3 June 2021).
23 Open Source IDE, see `https://openmodelica.org/` (last accessed: 3 June 2021).
24 Commercial IDE by ESI, see `https://www.simulationx.com/` (last accessed: 3 June 2021).
25 Commercial IDE by MODELON, which used to be open source, see `https://jmodelica.org` (last accessed: 3 June 2021).

sion 7.4), the empirically determined limit is about 160,000 equations. Our own investigations with newer versions of DYMOLA show a slightly higher limit.

In summary, it must be noted that current MODELICA IDES have significant performance issues when facing large-scale systems. Those issues were also recognized by Palensky et al. [129]. In a benchmark between MODELICA and GRIDLAB-D using a certain kind of large-scale ENERGY SYSTEM model, the GRIDLAB-D implementation outperforms the MODELICA implementation by far.

For performance, two different aspects can be considered, which are ultimately interlinked. Besides the already discussed aspect of the translation of the model, there is also the performance of the SOLVER itself, or rather its suitability for the respective problem. The two aspects are linked, amongst others, via the different capabilities of SOLVERS to handle DAES in implicit formulation directly, making transformation steps unnecessary.

In general, it is doubtful whether the classic MTP in conjunction with a multipurpose SOLVER such as DASSL is optimal for large systems and their characteristics. One characteristic of large systems is their sparse structure, as individual elements of the system are only linked to neighboring elements directly. The structure incidence matrix **S**, cf. Eqn. 3.14, has only relatively few entries set to 1.

However, at the current point in time and to the best of the author's knowledge, none of the major MODELICA IDES natively offer alternatives to the standard process, and surprisingly little research can be found that is dealing with alternative strategies. Even Frenkel [122] that deals with a compiler back-end for large MODELICA models does not call into question the general structure of the MTP.

There seems to be only one commercial tool in the broader context of the MODELICA language that uses a different approach. The concept of the IDA SIMULATION ENVIRONMENT[26], presented in a very early stage of development in Sahlin and Grozman [130], is to pre-compile the individual sub-models rather than to optimize the complete model before compilation. The authors name several benefits like reduced compilation time, simpler usage of tailored methods for special components, and easier integration of external tools for the handling of specialized components. As the purpose of this simulation environment is mainly building simulation, finite element analysis tools are mentioned specifically for the latter. The main drawback of the approach is a poor handling of high-INDEX systems or to handle such systems at all.[27] Although the IDA SIMULATION ENVIRONMENT has been expanded into a powerful tool

26 See `https://www.equa.se/en/ida-ice` (last accessed: 3 June 2021)

27 Unfortunately, Sahlin and Grozman [130] remain somewhat vague at this point.

for building simulation, the proposed concept was rather ignored by the MODELICA community.

Casella [128] discusses several challenges and promising methods to tackle large-scale systems, such as Zimmer [131] who suggests using the hierarchic structure of the MODELICA models rather than destroying it to speed up the translation. Reference is also made to work on *multi-rate* SOLVERS. This type of SOLVER strives to efficiently solve systems where parts have a faster dynamics than other parts by using multiple integration step sizes which adapt to the particular dynamics.

Against the background of large electrical power supply networks, Braun et al. [123] discuss possible alternatives to the established MMSP. A decisive role in their considerations is played by SOLVERS that can handle implicit DAE directly and provide optimized methods for models having a sparse structure. When using this type of solver, the complete causalization step or sub-steps such as tearing can be dispensed with. Braun et al. [123] show that the use of sparse solvers is advantageous for most of the systems considered in their investigation. In some cases, the required simulation time is reduced by a factor of 60. The results of the study were inconsistent concerning the advantage of the causalization step which, however, was always performed without the tearing step. For some models, the causalization step is advantageous regarding simulation time, for others, it is not. Casella et al. [132] and Braun et al. [123] also show that an alternative MMSP can handle models that could not be processed by the established process, as the classic MTP fails. The models under investigation include a nonlinear model of the European grid describing its electromechanical behavior with up to 600,000 equations representing hundreds of generators and thousands of transmission lines – simulation task that could previously only be performed by specialized simulation tools, cf. Section 3.2.4.2. Casella et al. [132] and Braun et al. [123] however make no statement about how much the changed processing influences the translation time.

For the presented approaches of [132] and [123], OPENMODELICA is used since only an open-source implementation allows the necessary changes to the MMSP without the help of the vendor of the IDE. Unfortunately, the presented novel approaches are work in progress and far from productive, as the whole OPENMODELICA implementation is basically not optimized for the use of sparse solvers –which is typical for MODELICA implementations. This applies in particular to the flattening part, the code generation and how HYBRID MODELS are treated by DAE sparse solvers. Especially the last point is problematic for the use in the context of this work, because our models will be of this type.

For the present work, three findings can be derived from the problems described and the latest research results:

- Current MODELICA IDES are unsuitable for the handling of large models. Therefore, when creating models for MOCES that are to be used for modeling large systems, care must be taken to ensure that the modeling depth is as simple as possible, but sufficient for the purpose.
- The performance problem is not an intrinsic problem of the MODELICA language itself, and there are promising approaches being developed to process large models efficiently.
- MOCES builds on the conviction that these alternative process chains will be available soon, making MODELICA and MOCES a suitable way to model *and* simulate large systems.

3.3.4 *Reasons for Using Modelica*

The decision to choose MODELICA as the base for the development of MOCES was made at an early stage of the project based on the following considerations:

- In the field of ENERGY SYSTEM modeling, there is a lack of alternative modeling languages.
- MODELICA is designed as multi-purpose MODELING LANGUAGE that is not limited to a specific domain. The capability to handle HYBRID MODELS is proven.
- The object-oriented approach of MODELICA enables the modeling of systems out of existing models available in model libraries.
- MODELICA allows to focus on the modeling, as MODELICA IDES such as DYMOLA or OPENMODELICA translate the models into a simulator automatically.
- MODELICA IDES additionally provide non-functional simulation functionalities, such as a graphical user interface, model management, and simulation result management.
- In recent years, MODELICA has moved into the focus of several research activities in the context of the ENERGY SYSTEM and the POWER SYSTEM. We will discuss them in the next Section 3.3.5.

3.3.5 *Modelica-based Modeling of Energy and Power Systems*

MODELICA's ability to be used across domains means that it can be applied to many different aspects of the energy and power system. In the following we would like to present a few exemplary projects and research areas.

LARGE-SCALE ENERGY SYSTEMS To the author's best knowledge, there are no MODELICA-based models for a large-scale ENERGY SYSTEM. An exception in the broader sense is the MODELICA implementation of the Club of Rome's WORLD3 model [133]. WORLD3 models the ecosystem of the earth including non-renewable resources and air pollution, and was the foundation for the famous report *The limits to growth* [134] published in 1972. This work is worth mentioning here because the WORLD3 model was originally written in DYNAMO, which can be considered a precursor of MODELICA [133].

DISTRICT ENERGY SYSTEMS A large number of libraries exist for modeling of district energy systems and dynamic building simulation. In both areas, the IBPSA PROJECT 1[28] has succeeded in developing standard models in cooperation with several research institutions and making them publicly available.[29] The project offers mathematical models formulated in MODELICA for individual components for which there is a consensus on the physical effects which must be described and which can be neglected. These models are also used to automatically generate models of larger districts on the basis of data models and corresponding toolchains, see Thorade et al. [135].

TRANSIENT-EE In the area of *regional* energy systems, the TRANSIENT-EE project of the *Technical University of Hamburg-Harburg* and its successor project RESILIENTEE should be mentioned.[30] As part of the project, the ENERGY SYSTEM of the city of Hamburg was modeled as a MODELICA model. The focus of the model is on the joint modeling and simulation of CONVENTIONAL POWER PLANTS, RES-based power plants, the electricity grid, the gas grid, central and local hot water distribution networks, as well as large and small consumers of energy. The research question addressed by the project is summarized as follows:

> *The aim of the project was to identify innovative and reliable ways of efficiently integrating renewable energies into an existing energy supply structure in order to maximize the energy self-sufficiency of the energy system. (Hamburg-Harburg [136])*

The granularity of the model developed for Hamburg varies. Individual and relatively detailed models represent the conventional power plants; the models take into account, e.g., electromechanical transient phenomena. The dependency between the electricity sector and the heating sector is described by models of COMBINED HEAT AND POWER (CHP) plants

28 See `https://ibpsa.github.io/project1/index.html` (last accessed: 3 June 2021).
29 See `https://github.com/ibpsa/modelica-ibpsa` (last accessed: 3 June 2021).
30 See `https://www.tuhh.de/transient-ee/` (last accessed: 3 June 2021) for a project description and links to further information.

linking the two sectors. The demand side, on the other hand, is only modeled with a single time series with 15 min resolution. The electricity grid is not modeled at all. The modeling granularity of the gas network is similarly diverse.

In comparison to MOCES, taking a look at the integration of the processes to maintain the balance between production and consumption, the BMP described in Section 2.2.8, should be interesting. These processes are mostly not modeled in TransientEE. Against the background of a preferably self-sufficient ENERGY SYSTEM, a classic UNIT COMMITMENT PROBLEM, cf. Section 3.2.3.1, is solved to cover the expected load as cost-optimal as possible with the Hamburg power plants and to derive a schedule for the individual power plants.

The model of Hamburg also includes an *Economic Dispatch Model* and a model for the control power. However, as described in Dubucq and Ackermann [137], the UCP is not part of the Modelica model but is solved offline before the Modelica-based model is simulated; the results of the UCP are, therefore, provided to the simulation as fixed, table-based schedules.

ITESLA The iTesla Power System Library[31] is a part of the iTesla project[32]. The project description summarizes the goal of the project with "improving network operations with a new security assessment tool". The library enables the dynamic simulation of transmission grids and the connected machines; this kind of model was outlined in Section 3.2.4.2. Modelica was chosen in this context as it allows for an unambiguous and component-oriented description of individual components, and the models can therefore be used to generate models of large systems automatically.

COMMON INFORMATION MODEL Modelica is also discussed as a possible extension of the IEC 61970, known as COMMON INFORMATION MODEL (CIM). The CIM is intended for the information exchange between TSO and the respective networks but misses the aspect to unambiguously describe the dynamic behavior of the network elements. Gómez et al. [138] provide a POC that shows how MODELICA can be used to fill this gap.

3.4 AGENT-BASED MODELING AND SIMULATION

Before providing a primer on *agent-based* MODELING AND SIMULATION, we would like to clarify the perspective of the research community re-

31 See `https://github.com/itesla/ipsl` (last accessed: 3 June 2021).

32 Project website is not yet available, but `https://github.com/itesla/ipsl/wiki/Publications` (last accessed: 3 June 2021) gives a list of publications.

garding the characteristics of a SYSTEM and its MODEL, which in turn is also called a SYSTEM[33]. Unlike equation-based modeling, described in the previous Section 3.3, agent-based modeling does not attempt to create a white box model with a high *quality level*. Instead, it is assumed that the complexity of the system to be described and in particular the interaction between its elements is complex and partially unknown.

[139] give a detailed discussion on this aspect and use a definition of a system by W. Ross Ashby as starting point:

> *A system is a set of variables sufficiently isolated to stay discussable while we discuss it.*

We want to restrict ourselves to the aspect that this definition reflects a very critical perspective regarding the validity of a self-made MODEL of a SYSTEM for an EXPERIMENT. In fact, this perspective is much more critical than what is common in the field of POWER SYSTEM modeling. For many aspects in the POWER SYSTEM, reliance on models is certainly justified, such as for grid simulation, cf. Section 3.2.4.2. For other aspects, a more critical perspective seems appropriate and should be applied, cf. the discussion in Section 2.4.1.

3.4.1 *Modeling*

Agent-based modeling is based on the simple idea of modeling the behavior of individual entities –the AGENTS– solely in relation to their environment and to observe the consequences if a multitude of agents, collectively representing the overall system, interact with each other. This bottom-up approach is based on the assumption that, in contrast to the complex behavior of the entire system, the behavior and the goals of the individual entities can be described very well. Figure 30 gives an overview of the concept and the building blocks described in the following.

AGENT The agent as a concept has a set of characteristics; following Nikolic and Kasmire [139], agents are:

1. encapsulated, as they are uniquely identifiable and have a well-defined boundary and interface;
2. situated in a given environment, as they are sensing the environment and can stimulate the environment through actuators;
3. flexible regarding the target achievement, as they respond to changes and act with anticipation;

33 Cf. the discussion of MODEL vs. SYSTEM within the community in Section 1.5.

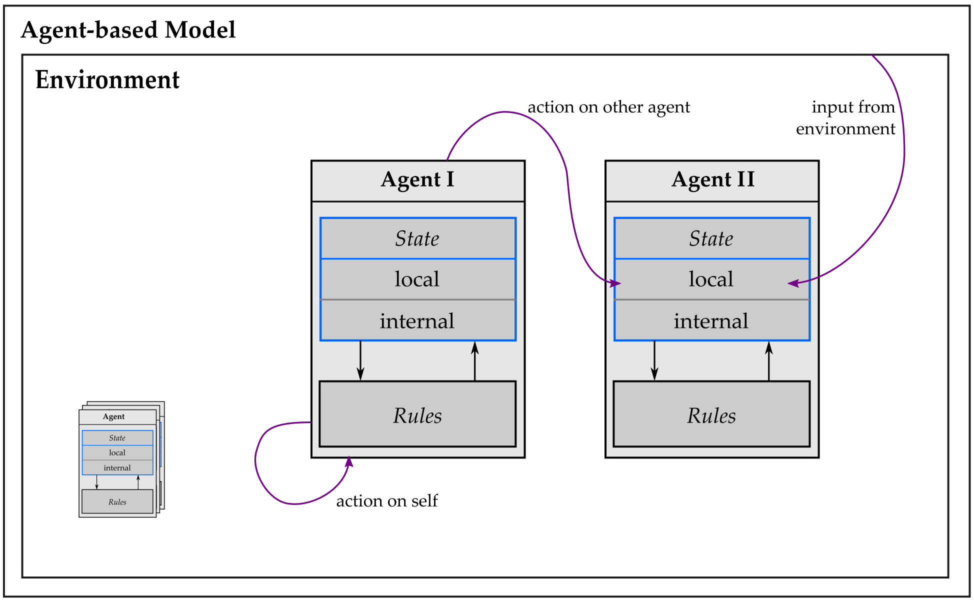


Figure 30: Overview of an agent-based model, inspired by Nikolic and Kasmire [139] and modified. Typically a large number of agents are used in an agent-based model.

4. autonomous, as they have full control over their INTERNAL STATE and their behavior;
5. goal-driven, as they strive to meet an objective.

STATE The state of the agent is represented by a set of variables. They hold information about the agent at one point in time. It is essential to distinguish between the LOCAL STATE and the INTERNAL STATE of an agent. As already mentioned, the INTERNAL STATE of an agent can only be changed by the agent itself. Contrary to that, the LOCAL STATE is influenced by other agents and the environment. It describes how an agent 'sees' the environment, respectively which part of the system it perceives. This view of the world is the basis on which the agents' decisions are made.

GOAL Agents pursue goals. A goal can be a simple task such as to monitor the environment or a complex one such as to maximize the profit of a power plant.

ACTION An action is an atomic activity of an agent that cannot be interrupted. An action *may* change the internal and LOCAL STATE of the agent acting and the LOCAL STATE of other agents. Examples for actions in the context of POWER SYSTEMS are "sending an offer to the day-ahead market" or "adapting the schedule for the next day" or anything else depending on the context of the model. An action can also be generic, such as "deciding what to do given the current state." According to [139, p. 57], actions can also act on the INTERNAL STATE of other agents. We expressly object to this possibility, as in our opinion it contradicts the autonomy of an agent.

RULES Rules "describe how states are translated to actions or new states" (Nikolic and Kasmire [139]). The term rule is to be considered abstractly here; a rule can be a set of simple decision rules in the form of if-then-else structures. However, rules can be much more complex and do not have to be deterministic. Rules can also be dynamic, meaning they can be changed or adjusted by the agent during its lifetime by learning from previous decisions.

Following the basic idea of agent-based modeling, there are no restrictions in formulating the rules. The following types of rules are therefore to be understood as options that can be combined. The list is inspired by Nikolic and Kasmire [139] and extended.

A. Rules based on decision rules can have the form of simple if-then-else structures. However, the decisions can be based on complex calculations of the agent, such as solving an optimization problem. Solving an optimization problem to make a decision is intensively used within MOCES, cf. Section 4.4.9.1.

B. Machine learning methods, such as ARTIFICIAL NEURAL NETWORKS (ANNS), can be used to deduce a decision without explicitly encoding rules. However, such methods typically come at the cost of a learning phase requiring data of the fitness of the decisions made in the past. This data is generated only during the lifetime of the agent, which in turn means that learning must take place during the simulation, and usually agents do not make advantageous decisions at the beginning of the simulation. In MOCES, this option is also used, but to a limited extent. In Section 5.4.5, we will furthermore show that a slightly modified formulation leads to advantageous decisions at the beginning of the simulation, respectively the agent's lifetime.

C. Rules based on Petri nets can be interpreted as a variant of the decision rules. Contrary to them, Petri nets come with a methodology for straightforward modeling of cyclic and parallel processes which are typical in the area of POWER SYSTEMS, cf. the BMP in Section 2.2.8. As we will present in Section 4.4.9, MOCES primarily uses this type of rule. If other types of rules are used, they are included in the rules based on Petri nets.

ADAPTATION Adaptation is the ability to change decisions by evaluating the decisions made and thus to make better decisions in the future; adaptation can be an important characteristic of an agent. A widely used method to describe this property is the approach presented by Erev and Roth [111], which is briefly described in the context of energy market simulation in Section 3.2.4.1.

3.4.2 *Simulation*

So far, we have not yet made any statement about the order of the agents' actions, i.e. how they are scheduled, which is required to perform a SIMULATION. For that purpose, we introduce the notion of an EVENT. In this thesis, an EVENT indicates that something happens at a particular point in time. Under certain circumstances, this point in time may be in the future, meaning that the occurrence of the EVENT is known in advance. Typically, however, an EVENT occurs *spontaneously* but nevertheless due to a reason or by chance. An EVENT triggers one or more actions. A specific EVENT occurs exactly once, but multiple EVENTS can, and typically do, share the same reason as well as the resulting actions.

Agent-based modeling and simulation frameworks do not share a common methodology for translating a MODEL into a SIMULATOR. In detail, they differ in how to handle the SIMULATION TIME and how to schedule the actions of the individual AGENTS, or rather the EVENTS triggering corresponding actions (Grimm and Railsback [140]). Therefore, if not using an existing framework, decisions have to be made regarding the design of the SIMULATOR and its core, named scheduler. Those design decisions for the SIMULATOR are typically coordinated with the design decisions regarding modeling since the various options are differently well suited for different types of models.

Grimm and Railsback [140] mention three interlinked design decisions to be made. We will provide them in an adapted version: how to represent time; how to map actions to EVENTS and generate EVENTS; and how to schedule EVENTS and their associated actions that take place at the same time.

HANDLING OF TIME A simple and widespread approach to model time is the assumption of discrete time steps, named ticks, with a fixed length. Accordingly, time proceeds in chunks and temporal dependencies within a single step are lost.

Another approach is to model time to be continuous, and the actions of the AGENTS are performed at any point in time as triggered by the related event. With this approach, time proceeds from EVENT to EVENT. As the length of the steps between events can be infinitely small, it is considered to be continuous.

MAPPING ACTIONS TO EVENTS AND GENERATING EVENTS It is the modeler's task to determine the different actions that can be performed by an agent, and it is a design decision how granular these actions are supposed to be. It is also up to the modeler to map these actions to events. One approach is to merge all possible operations of an agent into exactly one generic activity. This approach can be combined with stipulating that an EVENT can trigger exactly one action. Basically, all other conceivable decisions are possible; and, it depends on the modeled system and the chosen handling of time which design decision is the most preferable, see Grimm and Railsback [140] for a discussion.

Up to now we have not yet discussed how to process EVENTS and where the EVENTS originate from. To outline the processing of EVENTS, we can metaphorically arrange them at a timeline, as given in Figure 31. Events arranged orthogonally to the timeline represent events at the same point in time but with a defined order of execution. For simulation, if we use a continuous time model, we *slide* through the timeline with the executor and process the EVENTS in the given order; or, we jump from time step to time step if we use a discrete time model.

Adding the EVENTS to the timeline and handling the processing is the "dirty little secret of agent-based modelling" according to Dam et al. [84], as typically a central controller is used for that purpose. In the simplest case, the scheduler just adds one EVENT for every AGENT to each tick, cf. Figure 31a.

The DYNAMIC SCHEDULING approach is an alternative where EVENTS can be placed at any point in time on the timeline, cf. Figure 31b, if a continuous model of time is used.[34] According to Grimm and Railsback [140], this kind of scheduling works particularly well when the agents themselves, rather than a central scheduler, place their individual events onto the timeline. The approach is notably suitable if agents idle most of the time but sometimes perform several consecutive actions within a short time span.

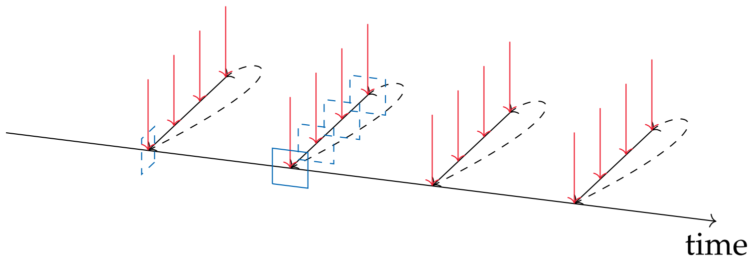
An exemplary implementation of the DYNAMIC SCHEDULING approach is provided by Colin Sheppard[35] as an extension for NETLOGO[36].

HANDLING OF CONCURRENT EVENTS Depending on how time is handled, EVENTS that occur *at the same time* may be a bigger or smaller challenge. The is not only a theoretical problem; as shown, e.g., by Ruxton [141], the ordering can have a significant impact on the simulation result.

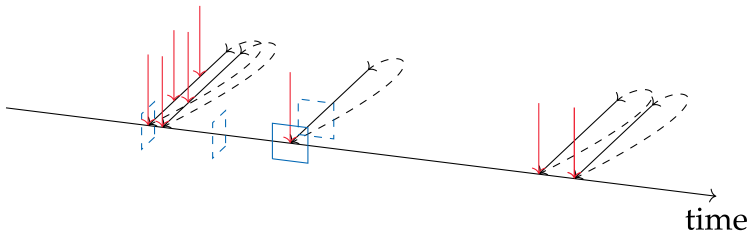
34 Dynamic scheduling can also be used in combination with fixed time steps. We will not discuss this further, as MOCES uses a continuous representation of time.

35 See `https://github.com/colinsheppard/Dynamic-Scheduler-Extension` (last accessed: 3 June 2021) for the implementation and a short description.

36 NETLOGO is a multi-agent modeling framework, see `https://ccl.northwestern.edu/netlogo/` (last accessed: 3 June 2021) for details.



(a) Discrete time model.



(b) Continuous time model in combination with DYNAMIC SCHEDULING.

Figure 31: Different possibilities for the handling of events (red arrows). The blue box shows the executor (solid marks the current position) sliding through the time (black line). Inspired by Grimm and Railsback [140] but significantly modified.

If a discrete representation of time is used, it is common that a large number of EVENTS have to be processed at the same time step, as a time span is squeezed to a single time step. One possible approach is to define a sequence, respectively a method how to order the EVENTS. For example, the execution can be random, follow a predefined hierarchy or any other ordering method.

If a continuous representation of time is used, concurrent events are far less frequent. With this kind of modeling of time, it is also feasible to argue away a simultaneity and to assume that two EVENTS basically cannot occur at the same time. However, from a practical viewpoint, we will have to tackle EVENTS occurring at the same point in time, as many entities typically have the same cyclic behavior, e.g., an action that reoccurs every 15 minutes. In the end, we face a similar problem as with the discrete representation of time and have to define a method how to order concurrent events.

Alternatively to such *asynchronous* approaches that give a sequence to the events, events can be processed in parallel. In simplified terms, this *synchronous* procedure fixes the LOCAL STATE of the individual agents, and all agents execute their actions on the basis of this state as if it were not influenced by the other agents. If using a synchronous approach, the question arises as to how conflicts can be resolved with shared resources, such as two agents requesting the same *packet* of energy.

3.4.3 *Modelica and Agent-Based Simulation*

There are already a few approaches to perform agent-based simulation with MODELICA. However, none of them have been widely adopted, have been developed beyond a POC status, or have been seriously discussed as an extension of the MLS. For the sake of completeness, some of the work should be mentioned here nevertheless.

The basic idea in Sanz et al. [142] is to represent the individual agents as messages that are processed via a flowchart. The behavior can, therefore, be described by the design of this flowchart. For implementation, the authors propose the introduction of new MODELICA language elements allowing interaction between messages and models. Unfortunately, the description here remains on a narrative level. It is therefore not entirely clear to the reader how the introduced language elements were implemented using existing MODELICA language constructs, such as external function interfaces, and whether these are ultimately syntactic sugar or profound changes. As an example of application, Sanz et al. [142] use the LOTKA–VOLTERRA EQUATIONS, in which the prey (sheep) is described as an agent while the predator (wolf) is described by continuous equations.

The work in Constantin et al. [143] follows a co-simulation approach using the Java-based JADE Framework[37]. For the coupling of the two simulators, the JADE functionality allowing communication via TCP/IP sockets is used. The interaction between the simulators is performed at discrete time steps defined by MODELICA and is set to 5 min in the application example. Unfortunately, a detailed discussion of the coupling of JADE and DYMOLA does not take place. In the application example, a temperature control strategy in an office room is determined by the agent. The temperature control itself as well as the thermal modeling of the room is done in MODELICA.

Although entitled *A Modelica library for the agent-based control of building energy systems*, Bünning et al. [144] describe only very vaguely how the presented library performs agent-based MS in MODELICA. The authors only provide references to the MODELICA `StateGraph` library[38], which is used to describe the behavior of the agents as state machines, and to the FOUNDATION OF INTELLIGENT PHYSICAL AGENTS (FIPA)[39], which is used as a standard, along with the UDP/IP stack provided by the `ModelicaDeviceDriver`[40] library.

37 See `https://jade.tilab.com/` (last accessed: 3 June 2021) for further information.
38 This library is part of the MSL.
39 See `http://www.fipa.org/` (last accessed: 3 June 2021) for an introduction.
40 A free library maintained by the DLR for hardware-in-the-loop simulation, see `https://github.com/modelica-3rdparty/Modelica_DeviceDrivers` (last accessed: 3 June 2021) for details.

3.5 SUMMARY

Chapter 3 set the foundation for the design of the MOCES MS framework that follows in the next chapter. For this purpose, the state of research was elaborated from two perspectives.

Sections 3.1 and 3.2 have focused on typical questions in the context of energy system analysis and showed which kind of models are used in each case. This broad overview has shown that for many tasks specialized but proprietary models and tools have been developed. However, a holistic modeling language to *bridge* different models, as required to investigate the interaction between different aspects of the energy system such as the grid and the market, is missing.

Sections 3.3 and 3.4 have described the two modeling approaches used by MOCES respectively: the equation- and component-based modeling based on MODELICA and agent-based modeling. The MODELING LANGUAGE MODELICA plays a special role here, since it is a candidate to fill the described gap of a holistic modeling language for energy and POWER SYSTEMS.

The next chapter will show how linking the two modeling approaches can be achieved in a meaningful way, and within the limitations of the MODELICA LANGUAGE SPECIFICATION of MODELICA.

4 MOCES

This chapter is dedicated to MOCES, a novel framework for the MODELING OF COMPLEX ENERGY SYSTEMS developed in this thesis.

We call MOCES a *framework* as it is *not* a ready-to-use model for a concrete scenario, but a skeleton including base models that can be used to investigate a concrete scenario, meaning to model a SYSTEM, run the SIMULATION and interpret the results. The framework additionally involves a modeling *concept* that is also developed in this thesis, which is basically independent of the MODELING LANGUAGE used; therefore, this work also refers to the *concept* MOCES in the relevant sections.

Accordingly, this chapter develops the concept, describes the design guidelines of MOCES and outlines the implementation of the framework, which uses the MODELING LANGUAGE MODELICA and an extension developed in C++. The chapter is an extended version of the short descriptions of MOCES already given in Exel et al. [145] and Exel and Frey [146].

We will start with the presentation of the main ideas of MOCES in Section 4.1. It provides a high-level overview that serves as an orientation for the following sections and the more detailed descriptions therein. Section 4.2 presents a non-formal ontology of the concepts of MOCES. It defines the main entities and their relation, and thus serves as a cornerstone for the modeling of complex systems and to describe the modeling approach.

Section 4.3 describes how the main ideas of MOCES can be implemented in MODELICA using a simple example. For this purpose, a prototype of that example is developed in pure MODELICA. The experience made with the prototype, in particular a series of performance tests, lead to the decision to develop parts of MOCES in C++. However, it will be done in a way that is consistent with the MLS.

Section 4.4 is concerned with the final implementation of MOCES. We focus primarily on the justification of design decisions, and on implementation details ensuring that the developed extension in C++ conforms to the MLS and performs well. In Section 4.5 we introduce a procedure model describing the modeling and simulation process that should be applied when MOCES is used to analyze a concrete scenario –such as done in the subsequent Chapter 5.

4.1 GUIDING IDEA

The detailed description of the POWER SYSTEM in Chapter 2 has revealed two main characteristics of the entire system. First, the POWER SYSTEM is a complex technical system with a vast number of technical units; second, the technical units are owned by different PARTIES pursuing their goals by interacting with each other and performing actions on their assets. Two additional aspects are essential: the behavior of many technical systems heavily depends on the environment, especially the weather; and, the interaction between the individual PARTIES is restricted and partially prescribed by the MARKET RULES.

In Chapter 3 we have presented two modeling approaches, each of which is suitable for representing one of the two characteristics: the component- and equation-based modeling approach allowing a structure-preserving modeling of technical systems, and the agent-based modeling approach enabling to model the interaction between AGENTS pursuing their goals with a high degree of freedom concerning the modeling of the decision making process. Both modeling approaches share the characteristics that the individual elements have a well-defined interface and allow structure-preserving modeling.

The core idea of MOCES is two combine both worlds for an adequate modeling of the POWER SYSTEM. A MODEL formulated as a HDAE in MODELICA, cf. Section 3.3.2 and Appendix A.1, is composed with an AGENT, as described in Section 3.4. We call this combination a MOCES ENTITY (MENT), with the parts called *Physical Node* and *Business Node* respectively. A visual representation of this core idea is given in Figure 32, that also shows a third crucial element, the *Adapter* interfacing both parts.

The figure additionally shows the element *Environment* which can affect the *Physical Node* and the *Business Node*. Crucially, unlike the connections between MENTS, this is necessarily a unidirectional influence, as the MENTS do no influence the *Environment*. A typical example for an aspect that is represented in this part of MOCES is the weather.

MOCES follows the paradigm of component-based modeling, cf. Section 3.3.1, and thus allows modeling of a complex system by instantiating models and connecting them.

Figure 33 shows an additional representation of MOCES. It is obtained by representing the two main aspects of the POWER SYSTEM as different layers, the PHYSICAL LAYER and the BUSINESS PROCESS LAYER, and adding a third level, the INFORMATION LAYER, which represents the environment. This illustration also shows the role of the adapter, which enables a connection between the two lower levels by linking the two parts of a MENT. As indicated by the axes x_1, x_2 and t in the figure, the PHYSICAL LAYER and the INFORMATION LAYER span a geographical space in which the individual nodes are placed. In the BUSINESS PROCESS LAYER, this

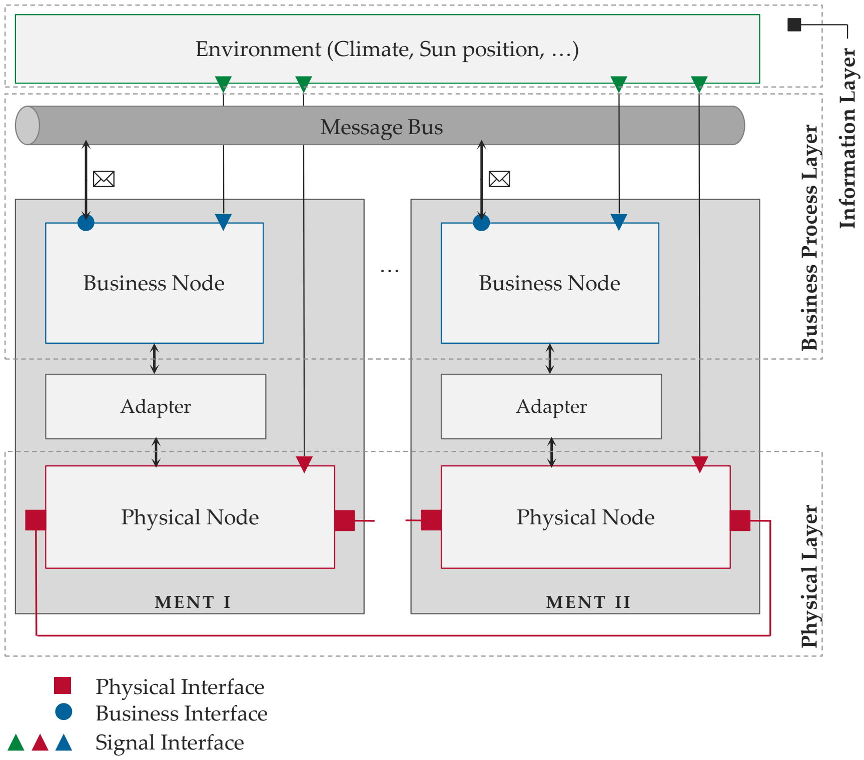


Figure 32: Basic structure of a MENT and its relation to other elements of MOCES.

positioning is only represented implicitly by the link to the *Physical Node*. Only a temporal axis exists in this layer.

Note that the Figures 32 and 33 introduce a color concept which will be used in the following whenever possible and helpful. Blue refers to all elements in the BUSINESS PROCESS LAYER, red to the PHYSICAL LAYER, and green to the INFORMATION LAYER.

4.2 ONTOLOGY

The ontology presented in this section is used as a cornerstone for the modeling framework MOCES. The approach to use an ontology as the basis for the definition of a framework is adapted from Dam [85] who developed an ontology for SOCIO-TECHNICAL SYSTEMS. We share the central idea that the overall system can be viewed as a set of *Nodes* interconnected by a set of *Edges*, and we adopt certain parts of that ontology. However, the ontology developed here differs fundamentally from Dam's. We will discuss the differences later in this section (cf. p. 131).

Figure 34 provides an overview of the entities of the ontology and their relationships to each other. A MOCES MODEL consists of precisely one *Environment*, one *System Node* and a set of predefined nodes, called *Ments*, connected by a set of *Edges*. *Edges* are either of type *Physical Edge*

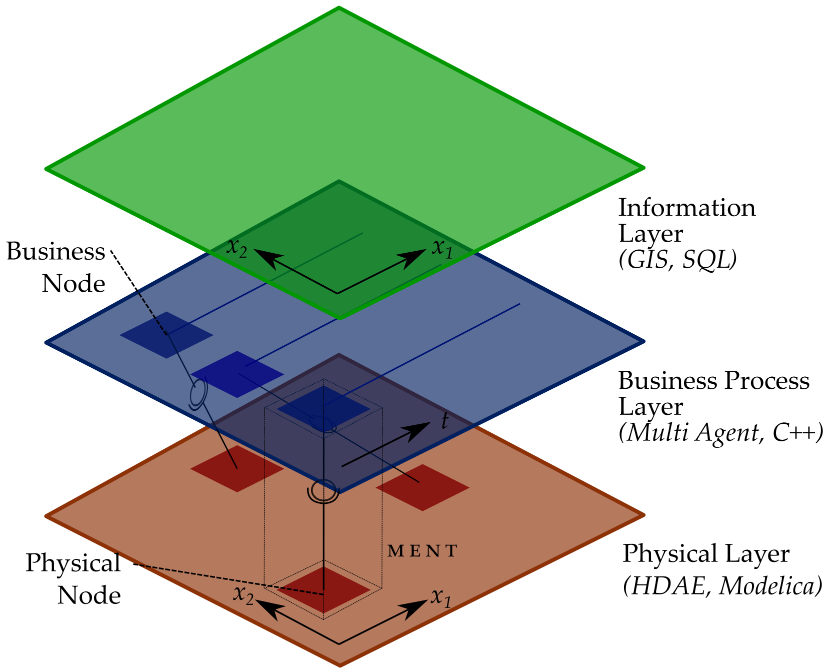


Figure 33: Layer architecture of MOCES

or *Business Edge*. *Physical Edges* are fixed for a given model, as are the *Ments*; *Business Edges* can change.

The *Ment* is the main building block of MOCES. A *Ment* consists of a *Physical Node*, a *Business Node*, and a distinctive element, called *Adapter*, interfacing both parts. The *Physical Node* and the *Business Node* of one *Ment* share a set of static parameters, but do *not* share a common state. However, the *Business Node* can define rules when to exchange which information. These rules are given to the *Adapter* and can be modified during the SIMULATION by the *Business Node*. The *Physical Node* reports its complete state or a subset thereof. A *Ment* only has two mandatory attributes: a unique identifier and an account balance. However, it typically has other optional attributes such as a GPS position. To align with the terms PARTY and ACTOR as given in Section 2.2.6, one can consider a *Ment* as a PARTY with a small number of associated ACTORS.

The *Physical Node* represents a physical element. It has a state represented by a set of continuous and/or discrete state variables. It is therefore *not* restricted to represent CONTINUOUS SYSTEMS; it can also represent HYBRID SYSTEMS. A *Physical Node*, respectively the linked *Business Node*, is capable of influencing its state, but cannot fully control it. Its state is also influenced by the nodes directly or indirectly connected to it. A *Physical Node* is only allowed to have *Physical Interfaces*.

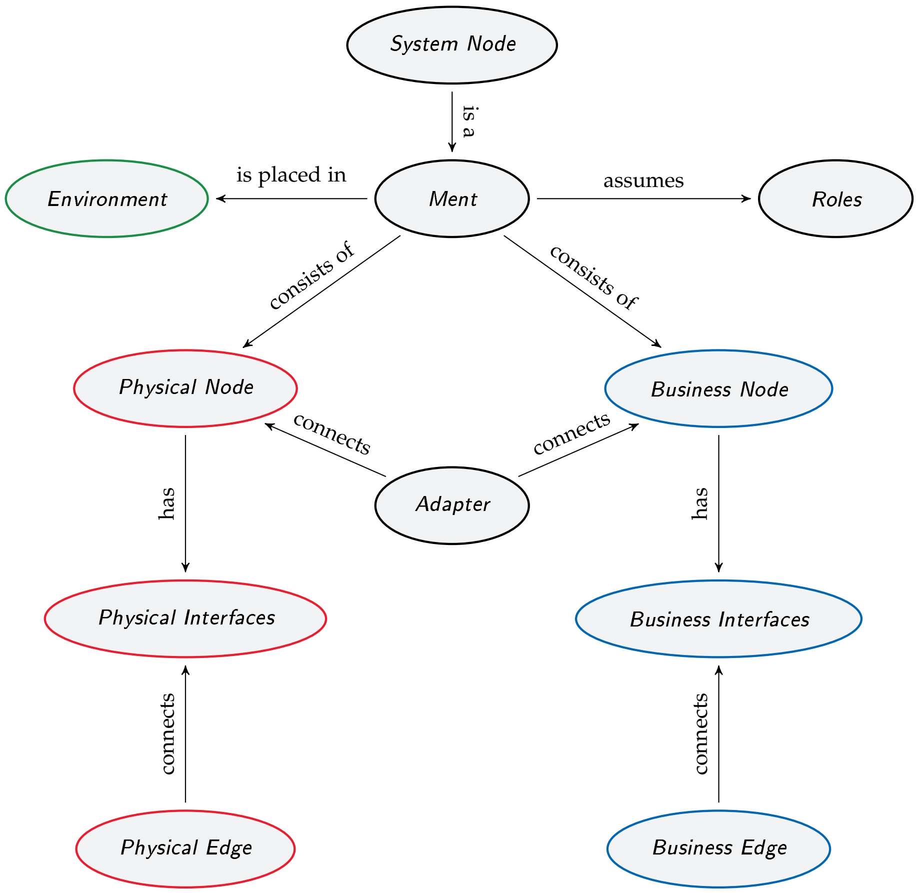


Figure 34: Fragment of the ontology used by MOCES

The *Business Node* represents "a [...] Node [...] capable of making decisions about [exactly one] *Physical Node*" (Dam [85]). A *Business Node* is an agent as defined in Section 3.4. Therefore, it also has a state, but, in contrast to the state of the *Physical Node*, this state is only influenced by the *Business Node* itself. This state is called INTERNAL STATE. The *Business Node* additionally has a LOCAL STATE, which consists of the reported state of the linked *Physical Node* and fragmentary information about other *Business Nodes*.

The *System Node* is a particular type of *Ment* that exists exactly once per model. All other *Ments* a linked to the *System Node* by *Business Edges* and *Physical Edges* and have to provide information to it.

One or several *Roles* are assumed and performed by a *Ment*.

The *Adapter* is the only element allowed to link the *Physical Node* of a *Ment* with its *Business Node* and thus enables an information exchange between the two nodes. The design of the *Adapter* is inspired by the general structure of a HYBRID SYSTEM as given by

Lunze, cited after Janschek [22, p. 140]. The *Adapter* holds a set of *discrete-time* variables, subdivided into a set of variables used to observe the state variables or any other variables of the *Physical Node* and a set of variables used to *inject* values into the *Physical Node*. These injected values should be limited to control variables that affect the physical system and its state variables. Besides, the *Adapter* holds a set of parameterizable EVENT CONDITIONS that trigger synchronization of the variables of the *Physical Node* with variables of the *Business Node*. The *Adapter* can, therefore, be interpreted as a kind of sensor-actuator system whose sampling-rate is dynamic and driven by conditions.

A *Meter* is a special type of *Adapter* dedicated to meter a *Physical Interface* and to report the values to the *Business Node* at predefined points in time. The action of reporting is not allowed to trigger an action by the *Business Node*.

A *Business Node Observer* is the equivalent to the *Meter* but reports values from the *Business Node* to the *Physical Node*.

The *Environment* represents the world outside the *Ments* that is not influenced by the *Ments*; however, the *Environment* heavily influences the *Ments*. The relation to the *Ments* is therefore restricted to *Signal Edges*, whose respective tail is the *Environment*. The *Environment* includes a WALL CLOCK that is accessible by all *Ments*, and it has a kind of GLOBAL INFORMATION SYSTEM (GIS) offering information on weather data based on time and geographic position.

An *Interface* is a point for interaction. It is either of type *Physical Interface* or of type *Business Interface*.

The *Physical Interface* defines a point for physical interaction. A *Physical Interface* has a specified type which defines the characteristics of the interface. A primitive interface is a pair consisting of a *flow* and an *effort*, cf. Section 1.6 (p. 17 ff.). More complex interfaces can be defined by a set of primitive interfaces.

The *Signal Interface* is a particular type of a *Physical Interface* that either represents the source or sink of a SIGNAL.

The *Business Interface* defines a point for interaction between *Business Nodes*. As *Business Nodes* interact by means of *Messages*, it has the characteristic of a message-consuming endpoint. *Messages* are used to enable information exchange between *Business Nodes*. A *Message* consists of a set of meta information, such as a sender identifier, a receiver identifier, and type of message, as well as, the payload of the message whose structure is specified by the type of message. It is up to the recipient to respond to a message or not.

An *Edge* indicates a possible interaction between two or more *Ments*.

A *Physical Edge* represents a direct physical relationship between two or more *Physical Nodes*, respectively their *Physical Interfaces*. A *Physical Edge* has no orientation and is only allowed between *Interfaces* of the same type. A *Physical Edge* has no further properties. The only information provided by a *Physical Edge* is the linking of two or more *Physical Nodes*. *Physical Edges* are static. However, please note that this limitation does not restrict the model to structure-invariant systems, as a *Physical Node* can represent a hybrid system, such as a switch, which can result in a structure-variant system.

A *Signal Edge* is a special type of *Physical Edge*. In contrast to the *Physical Edge*, it has a defined CAUSAL DIRECTION.

A *Business Edge* represents a relationship between *Business Nodes*. A *Business Edge* is an abstract concept indicating that two *Business Nodes* know each other and interact in some way. The relationships represented by *Business Edges* are dynamic during the simulation. According to the concept of an AGENT, the direct influence of a *Business Node* on another *Business Node* through a *Business Edge* is limited to the modification of the other node's LOCAL STATE, or rather to *adding* information to that LOCAL STATE.

The described elements can be mapped to the three layers of MOCES (cf. Figure 33) in a straightforward way. The *Physical Nodes* with the *Physical Interfaces* connected by *Physical Edges* are placed in the plane of the PHYSICAL LAYER. Accordingly, the *Business Nodes* and the *Business Edges* lie in the plane of the BUSINESS PROCESS LAYER. The *Environment* is located in the INFORMATION LAYER. A *Ment* spans the BUSINESS PROCESS LAYER and the PHYSICAL LAYER by using *Adapters*.

RELATION TO DAM'S ONTOLOGY As stated before, even though the given ontology is inspired by Dam [85], it differs in several crucial aspects. According to Dam [85], a *Node* is either of type *Physical Node* or *Business Node*. As nodes of different types cannot be directly linked with the defined edges, they are linked by a third class of edges, *Ownership Edges*, that represent the ownership relations of the network. This distinction is made to allow "that social and technical aspects of the system can be modeled independently of each other" (Dam [85]). We disagree with this decision. In general, both aspects should be modeled together. Consequently, we postulate that a *Node*, in our case a *Ment*, is always both, a *Physical Node and* a *Business Node*.

In Dam [85], *Physical Edges* represent the infrastructure needed to connect nodes, such as power cables, or they represent the physical flow, such as the power flow. Dam therefore models *Physical Edges* as non-symmetric, which means they have a fixed source and sink, and thus a specified CAUSAL DIRECTION[1]; and, they connect exactly two *Nodes*.

We expressly disagree with this kind of modeling in two aspects. We postulate that an *Edge* represents nothing more than the connection of *Nodes* and has no predefined CAUSAL DIRECTION. As discussed in Section 1.6, if two or more *Physical Systems* are connected, the causal structure is not yet known at the modeling stage if we use a structure-preserving modeling approach. Furthermore, in our opinion, a *Physical Edge* should not represent any parts of the infrastructure, as there is nothing that discerns a pipe from a *Physical Node*.

4.3 MOTIVATING EXAMPLE AND PERFORMANCE ASSESSMENT

As noted earlier, we have chosen not to implement the concept described in Figure 32 in MODELICA solely, but to describe the BUSINESS PROCESS LAYER and the INFORMATION LAYER as extensions in C++, and to link them to the MODELICA models via a suitable synchronization mechanism. This decision was based on the author's experience in modeling with MODELICA, cf. [147, 148, 149, 150], and on the knowledge gained from a POC implemented in pure MODELICA. This POC is presented in the following as a motivating example and outlines the developed synchronization mechanism as well as its advantages compared to an alternative synchronization mechanism.

The example is intentionally not located in the domain of the ENERGY SYSTEM to underline the generality of the approach and to keep the system as simple as possible.

4.3.1 *System Description*

The considered system, see Figure 35, consists of a ball placed in a gravity field and a set of movable boards trying to prevent the ball from falling below the ground level.

The ball moves at a constant speed in the direction that is perpendicular to the falling direction. There is no air friction, and a possible collision of the ball with a board is assumed to be an ideal elastic collision. The individual boards have a respective radius of action, in which they are responsible for preventing the falling of the ball below the level

1 Dam [85] does not discuss this aspect in detail. It is not definitive from his explanations whether his ontology defines a fixed causal direction or only a convention for the sign of the flow direction. From formulations such as "the Node the flow is originating from," however, we deduce that Dam assumes a defined causal direction.

of the boards. More precisely, it is not the boards alone that prevent the ball from falling, but rather an external entity for each board that determines a set point for its position, which the board follows with a certain dynamic.

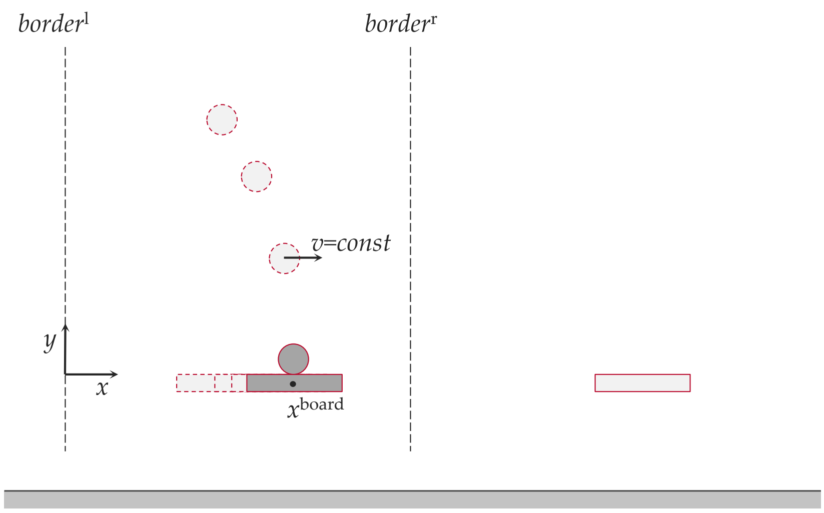


Figure 35: Overview of the example system.

We will use the following notation. x, y is the horizontal, respectively vertical position of the ball. x_i^{board} is the horizontal position of the i-th board with the width w^{board}. The radius of action of the i-th board is bounded by $border_i^{l}$ on the left side and by $border_i^{r}$ on the right side. The dynamics of the ball in the horizontal direction are given by $\dot{x} = v$, where v is constant. The dynamics in the vertical direction are described as follows: if $y \in \{y \neq 0\}$, then $\ddot{y} = -g$, where g is constant, if $y \in \{y = 0\}$, then $y^+ = -y^-$. $x(t = 0) = x_{\text{start}}$ and $y(t = 0) = y_{\text{start}}$ are positive numbers. We postulate that each board follows a board specific set point $x_s^{board}(t)$ with the following dynamics[2]:

$$\dot{x}^{\text{board}} = \frac{\left(x_s^{\text{board}} - x^{\text{board}}\right)}{k}, \text{with } \ k \ll 1. \tag{4.1}$$

In terms of MOCES' notions, we have described the *Physical Nodes* – the ball and the boards– and the *Environment* –the gravity field– so far. Besides, the *Physical Nodes* are each linked to a respective *Business Node* that perform actions on the *Physical Node*; together they form a *Ment*. If wanted, we can formulate the *Role* of the *Ment* as: "A PARTY responsible for keeping the ball above the ground."

The *Business Node* of a board can decide where and how to move the board by setting x_s^{board}. To do so, it has to observe the state of the ball and

2 For the sake of simplicity we neglect the index i from now on.

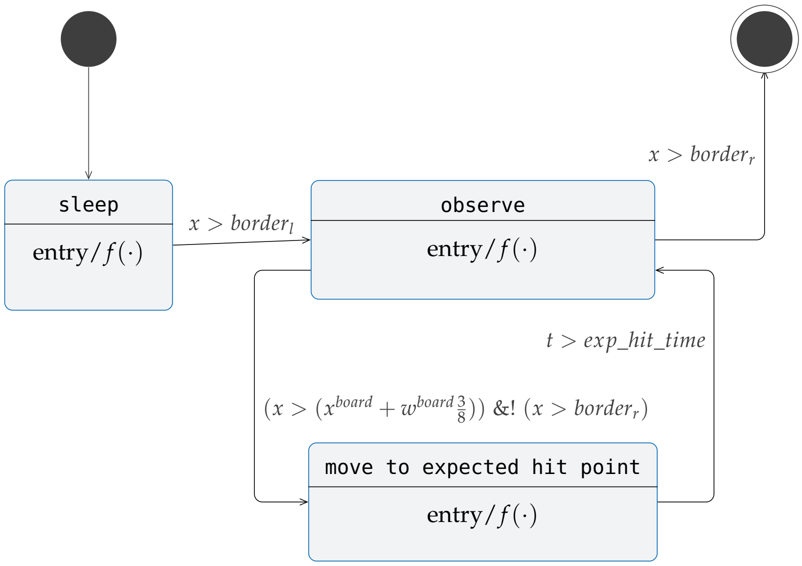


Figure 36: State machine of the *Business Node* performing actions on the linked *Physical Node* representing the board.

the board; and, the *Adapter* has to handle the synchronization of the information. As described in the previous section, it is the *Business Node*'s task to define the rules specifying when an exchange of information takes place. A simple procedure would be a time-discrete observation of the variables x and x^{board}, and an injection of x_s^{board}. In this case, the necessary sampling rate must be adjusted to the dynamics of the ball to prevent it from falling. Such rules may be formulated in MOCES, but are not recommended as we will later see in Section 4.3.4 when investigating the performance.

Instead, consider the following preferred strategy: When the ball enters the region of responsibility of the i-th board, the board should observe the state of the ball *once* and inject parameters of a trajectory for x_s^{board} that leads the board to the expected point where the ball hits the ground at the expected hit time. We use a trajectory with the following polynomial form and the parameters a and b:

$$x_s^{\text{board}} = a \cdot (t - t_1)^3 + b \cdot (t - t_1)^2 + x^{\text{board}}(t_1) \tag{4.2}$$

allowing us to set the speed of the board to 0 at the beginning and end of the movement; t_1 is the point in time of observation. Note, that the parameters a and b are not constant, but change their value at discrete time steps only. With reference to the MLS, we call these variables DISCRETE-TIME VARIABLES.

We will discuss this idea in more detail by using the state machine given in Figure 36 describing the behavior of the *Business Node* of the i-th board. Each state involves an entry function $f(\cdot)$ that calculates:

- the values for a and b as set points of the trajectory, and,

- DISCRETE-TIME VARIABLES modifying the rules when to perform the next action on the *Physical Node*.

As given by the state machine, the node is in the `sleep` state until the ball enters the radius of action of the board. If the ball enters the radius, the state changes to the `observe` state. If the position of the ball in the x-direction is larger than $x^{\text{board}} + w^{\text{board}}\frac{3}{8}$, the board should move to the expected point where the ball crosses $y = 0$. If the ball leaves the radius of action, the *Business Node* enters the final state. As the entry functions of the states `sleep` and `observe` stop the movement of the board, both functions set $a = 0, b = 0$. The entry function of the `move to expected hit point` state is determined in such a way that the following applies:

$$x_s^{\text{board}}(t_1) = x^{\text{board}}(t_1), \tag{4.3a}$$
$$\dot{x}_s^{\text{board}}(t_1) = 0, \tag{4.3b}$$
$$x_s^{\text{board}}(t_2) = \hat{x}(t_2), \tag{4.3c}$$
$$\dot{x}_s^{\text{board}}(t_2) = 0. \tag{4.3d}$$

So far, we did not discuss how the *Business Node* determines the hit time t_2 and the expected position of the ball in the x-direction $\hat{x}(t_2)$. For this purpose, the *Business Node* uses the observed state of the ball and certain knowledge of the system. Precisely it observes the variables $x, y, \dot{x}, \dot{y}$; and, it knows how to calculate the trajectory of a falling ball.

To define the behavior of the *Business Node* and its actions on the *Physical Node* in a more formal way, we introduce the following notation:

$$X.B \left\langle \left[\mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{info},+}, \mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{cond},+}\right] := \mathbf{f}\left(\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}\right)\right\rangle \underset{\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}^{*} \circledcirc \mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{cond},-}}{\overset{\text{cur. state}\rightarrow\text{next state}}{\circlearrowright}} X.P \left\langle \mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}} \right\rangle \tag{4.4}$$

which has to be read as follows: $X.P$ is the *Physical Node* represented by a set a variables $\mathbf{v} := [\dot{\mathbf{x}}\ \mathbf{x}\ \mathbf{y}\ \mathbf{m}^{-}\ \mathbf{m}^{+}\ \mathbf{p}]^{\mathsf{T}}$[3]. $\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}$ is the corresponding subset of variables which are observed by the *Business Node* $X.B$. If any of the EVENT CONDITIONS $\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}^{*} \circledcirc \mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{cond},-}$ becomes true, $X.B.$ is performing an action; $\circledcirc$ is defined as an element-wise relational operator. The action, notated with $\mathbf{f}(\cdot)$, is performed on a dedicated subset of the variables of $X.P$, namely $\mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{info},+}$, and the DISCRETE-TIME VARIABLES used by the event conditions $\mathbf{m}_{\mathsf{BN}\rightarrow\mathsf{PN}}^{\text{cond},+}$. The plus or minus sign at the variables refers to the right or left limit value of the respective variable, which typically changes if the *Business Node* is performing an action. We emphasize that this action may, and typically does, evolve depending on the state of the *Business Node*.

3 The naming of the variables follows the notation of MODELICA given in Appendix A.1.

Applying the notation to the bouncing ball example leads to the following for each board:

$$X.B \left\langle [a^+, b^+, x_c^+, t_c^+] := \mathbf{f}\left(t, x, \dot{x}, y, \dot{y}, x^{\text{board}}\right) \right\rangle \underset{\begin{bmatrix} t > t_c^- \\ x > x_c^- \end{bmatrix}}{\overset{\text{cur. state}\to\text{next state}}{\circlearrowleft}} X.P \left\langle \cdot \right\rangle . \tag{4.5}$$

We can also use the notation to describe the transition from one state into the subsequent state together with the action performed by the entry function. For the transition from the `initial` to the `sleep` state we obtain:

$$X.B \left\langle \left[a^+, b^+, x_c^+, t_c^+\right] := [0, 0, border_l, \infty] \right\rangle \underset{\begin{bmatrix} t > 0 \\ x > -\infty \end{bmatrix}}{\overset{\texttt{initial}\to\texttt{sleep}}{\circlearrowleft}} X.P \left\langle \cdot \right\rangle . \tag{4.6}$$

For the transition from the `sleep` state to `observe` state the following applies[4]:

$$X.B \left\langle \left[a^+, b^+, x_c^+, t_c^+\right] := \begin{cases} [0, 0, x^{\text{b.}} + w^{\text{b.}} \frac{3}{8}, \infty] & \text{if } x < border_r \\ [0, 0, border_r, \infty] & \text{else} \end{cases} \right\rangle$$

$$\underset{\begin{bmatrix} t > \infty \\ x > border_l \end{bmatrix}}{\overset{\texttt{sleep}\to\texttt{observe}}{\circlearrowleft}} X.P \left\langle \cdot \right\rangle . \tag{4.7}$$

The values for x_c^+ and t_c^+ determined by Eqn. 4.6 now appear as the new EVENT CONDITIONS in Eqn. 4.7. If entering the `observe` state, the action on $X.P$ depends on the relation between x and the right border ($border_r$). Thus, the *Business Node* determines by the triggered action which transition fires next; in the example, the transition to the `move` state ($x < border_r$) or to the `final` state ($x \leq border_r$). If the `move` state is selected as the next state, we get:

$$X.B \left\langle \left[a^+, b^+, x_c^+, t_c^+\right] := [a_1, b_1, \infty, t] \right\rangle$$

$$\underset{\begin{bmatrix} t > \infty \\ x > x^{\text{board}} + w^{\text{board}} \frac{3}{8} \end{bmatrix}}{\overset{\texttt{observe}\to\texttt{move}}{\circlearrowleft}} X.P \left\langle \cdot \right\rangle . \tag{4.8}$$

4 $x^{\text{b.}}$ is just a short form of x^{board}.

If the `final` state is the next state, the following applies:

$$X.B \left\langle \left[a^+, b^+, x_c^+, t_c^+\right] := [0, 0, \infty, \infty] \right\rangle \underset{\begin{bmatrix} t > \infty \\ x > border_r \end{bmatrix}}{\overset{\texttt{observe}\rightarrow\texttt{final}}{\circlearrowleft}} X.P \langle \cdot \rangle \,. \quad (4.9)$$

By setting the EVENT CONDITIONS to infinite, which thus never become true, the *Business Node* deactivates itself.

If the `move` state is selected as next state instead by Eqn. 4.7, the board starts moving if the ball's x-position is greater than $x^{\text{board}} + w^{\text{board}}\frac{3}{8}$.[5] The transition from the `move` to the `observe` state is similar to Eqn. 4.7 but differs in the EVENT CONDITIONS:

$$X.B \left\langle \left[a^+, b^+, x_c^+, t_c^+\right] := \begin{cases} [0, 0, x^{\text{b.}} + w^{\text{b.}}\frac{3}{8}, \infty] & \text{if } x < border_r \\ [0, 0, border_r, \infty] & \text{else} \end{cases} \right\rangle \underset{\begin{bmatrix} t > t_c \\ x > \infty \end{bmatrix}}{\overset{\texttt{move}\rightarrow\texttt{observe}}{\circlearrowleft}} X.P \langle \cdot \rangle \,. \quad (4.10)$$

Contrary to the transition from the `sleep` to the `observe` state, the input function of the `observe` state is not triggered by the ball entering the area of responsibility, but by the point in time t_c being exceeded. This time is the expected moment of collision of the board with the ball.

4.3.2 *Modelica Implementation*

Implementing the boards and ball dynamics in MODELICA is straightforward. The core of the implementation is described by the code fragment given in Listing 4.1[6].

Lines 5–7 implement Equation 4.2. Lines 9–15 show the implementation for Equation 4.5. The action of the *Business Node* on the *Physical Node* is represented by the function `ActionOnBoard` that is included in a `when` block and determines the DISCRETE-TIME VARIABLES a and b for

5 The decision not to use exactly $x^{\text{board}} + w^{\text{board}}\frac{3}{8}$, i.e. the right border of the board, is due to having a safety margin. However, this is not relevant for the example.

6 See Appendix A.2 for the full vectorized source code. The naming of the variables differs slightly, as the code snippets in this section have been adapted for better readability.

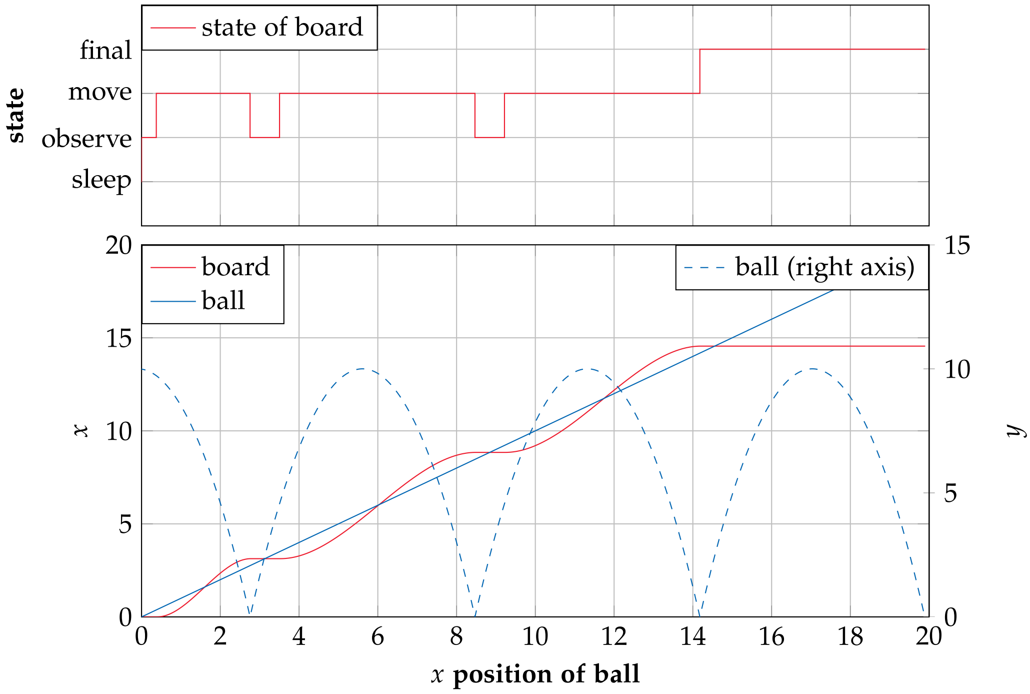


Figure 37: Behavior of the first board and the ball.

the trajectory. The conditions for activating the `when` block are also returned by the function call; the EVENT CONDITIONS align with Equation 4.5. The usage of the `when` language element shown in lines 9 to 15 thus implements an *Adapter* as it describes how and which information is exchanged between both nodes.

The `pre` operator returns the *left limit* of a variable at a time instant. It thus allows to model the change of the *Business Node*'s state and the event conditions as given by the when block in Listing 4.1.

Lines 19 to 35 of Listing 4.1 outline the `ActionOnBoard` function and thus the *Business Node* and its behavior. In principle, the function implements the Equations 4.6 to 4.10 which are selected depending on the state.[7]

A SIMULATION of the MODEL `MotivatingExample` results in the behavior that is depicted in Figure 37. This diagram shows the behavior of the first board only which moves a total of three times before the ball leaves its area of responsibility.[8] Figure 38 shows in detail how the parameters a and b of the trajectory change over time. It also depicts the resulting trajectory for the board x_s^{board}. Figure 39 illustrates the behavior of the

7 As can be seen in the code fragment, the function has more inputs and outputs than shown in Equation 4.5. We will discuss the reason for this in the next section.

8 As can be seen in the diagram , the board partially leaves its area of responsibility that ends at $x = 10$. This is just a technicality which could be prevented by limiting x_s^{board} to $boarder_r$ which has been omitted in this example.

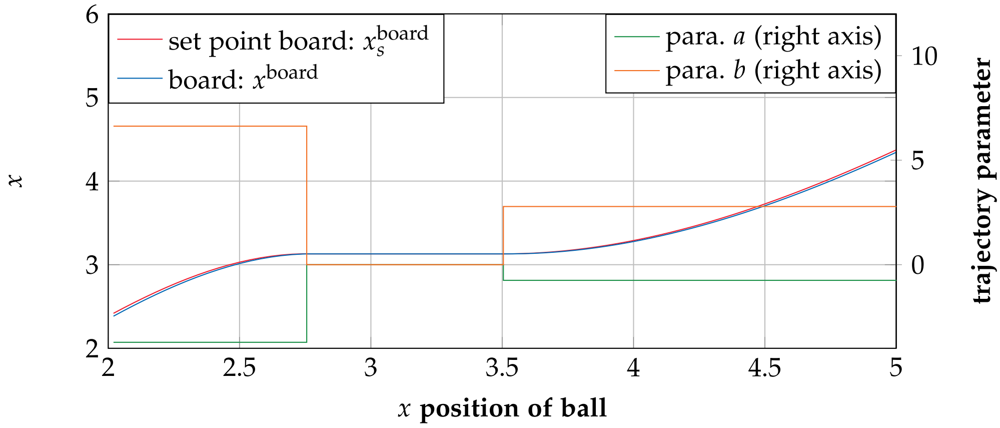


Figure 38: Behavior of the board (x^{board}) in relation to the given trajectory (x_s^{board}) and its parameters a, b.

first four boards. It demonstrates how the individual boards switch to the `observe` mode when the ball enters their area of responsibility and how they switch to the `final` state when it leaves it again. The graph thus shows how the boards, one by one, ensure that the ball does not fall below $y = 0$.

4.3.3 *Discussion*

The implementation of the motivating example illustrates important properties of how MOCES is capable to link equation-based and agent-based modeling, and thus allows for a holistic MS of the POWER SYSTEM.

With the `ActionOnBoard` function giving the conditions for triggering the next action, we have found a way to implement a DYNAMIC SCHEDULING approach in MODELICA where the individual *Ments*, or rather the *Business Nodes*, specify the EVENT CONDITIONS. Those are either of type TIME EVENT or STATE EVENT. In reference to the different layers of MOCES, this means that the *Business Node* part of a *Ment* can react to changes in the physical layer. However, it has to specify the expected change; this also means that the *Business Node* cannot perform any further action if the expected event does not occur. It is therefore in the responsibility of the *Business Node* to avoid its deadlocking due to EVENT CONDITIONS that never become true. For this, one can use the possibility to formulate several EVENT CONDITIONS; each one triggers an action of the *Business Node*.

When comparing Equation 4.5 with the function call in lines 11-12 of Listing 4.1, there are some differences concerning the input and output parameters. This is because the behavior of the *Business Node* is implemented using a PURE FUNCTION, `ActionOnBoard`, due to the limitations for algorithms in pure MODELICA. As a workaround, the state of the *Busi-*

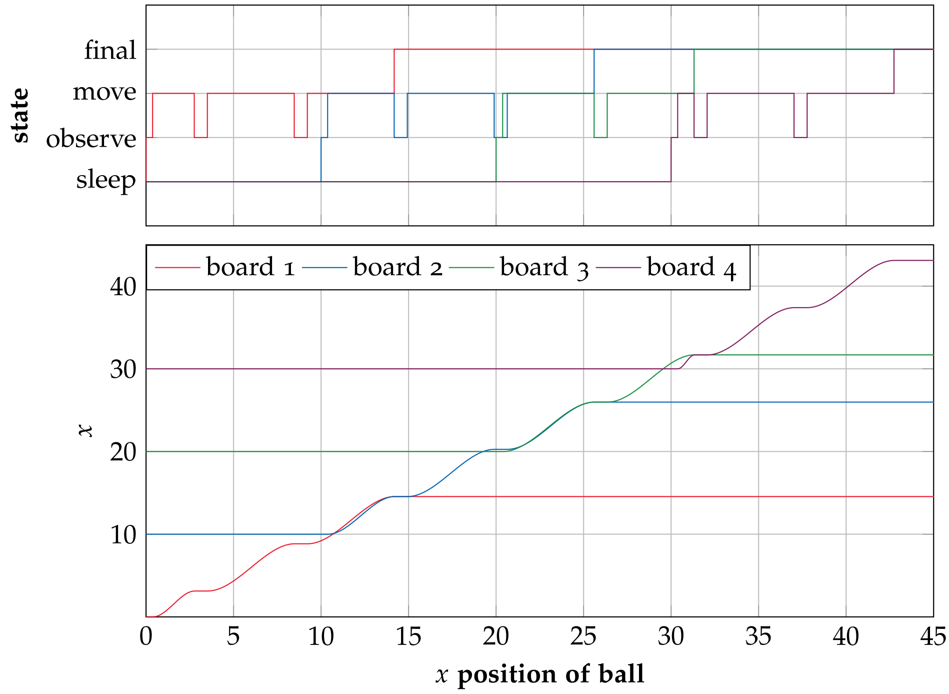


Figure 39: Behavior of the first four boards.

ness Node `agent_state_next` and static parameters such as $border_l$ and $border_r$ are added to the *Physical Node* and given to the function as input at each call. This limitation in modeling the behavior of *Business Nodes* is a decisive reason not to describe their behavior in MODELICA but using C++ for MOCES.

MOCES makes intensive use of the possibility to describe time-continuous trajectories by DISCRETE-TIME VARIABLES as shown with x_s, the set point for the board determined by the *Business Node*, cf. Equation 4.2 and lines 5-7 in Listing 4.1.

Note that the motivating example does not reflect *all* elements of MOCES introduced in Section 4.2; e.g., no interaction between *Business Nodes* takes place.

4.3.4 *Alternative Approach and Performance Assessment*

The modeling approach outlined with the motivating example in this Section 4.3 and in particular the design of the *Adapter*, cf. Listing 4.1, was developed against the background of creating a high-performance solution. Performance here refers to the WALL TIME required to simulate large systems.

To adumbrate the performance gain of the presented approach (*dynamic scheduler*), we compare it with an implementation that uses an *Adapter* implementing a synchronization at discrete time steps with fixed step size. We will refer to this as *fixed time step scheduler*. The code frag-

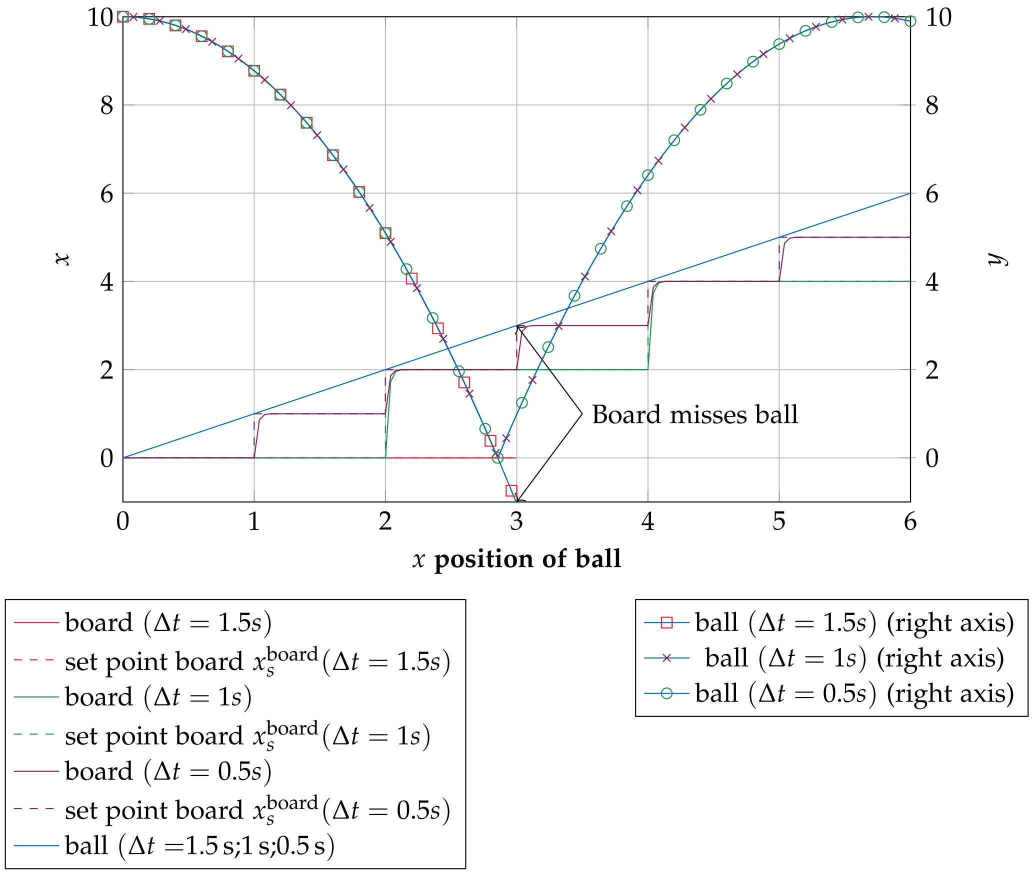


Figure 40: Behavior of board and ball for the *fixed time step scheduling* approach. Behavior is given for different values of Δt. If Δt is set to 1.5 s, the board misses the ball as the gap between the board's x position and the ball's x position is not covered by the board's width, see the position marked with "Board misses ball".

ment implementing this type of *Adapter* with a fixed step size `delta_t` is given in Listing 4.2.

The strategy used in Listing 4.2 differs from the previous one in Listing 4.1 in that the strategy of the *Business Node* is to set the board's set point x_s^{board} to the current value of the ball, cf. line 7. The resulting behavior of the board is shown in Figure 40. In this approach, the behavior of the system depends on the fixed step size of the scheduler. Parametric settings of the step size for which the *Business Node* fails to prevent the ball from falling are easily conceivable as Figure 40 shows.

Table 5 provides the results of a performance test comparing both approaches in combination with different SOLVERS. To investigate different sizes of systems, we scale the number of boards; the parameter Δt is set to 0.1 s for all experiments. For details see the source code of the models given in the Appendix A.2, cf. Listing A.2 and Listing A.3.

The key metric for the performance of the solver is the ratio between the duration of the experiment (expected simulation time), which increases linearly with the number of boards, and the run time the simulation (WALL TIME). This metric thus indicates how many seconds of SIMULATION TIME per second of 'real time' are simulated by the simulator.

The absolute numbers given in the table are not of interest here, as they heavily depend on the parameter settings of the model, e.g. for k or x_{start}, the performance of the machine running the simulation and further configurations such as which variables are stored. We therefore want to focus on the relation of the obtained values. The absolute values of the WALL TIME should be understood as a rough estimate for a standard machine. For the performance test, the solvers DASSL, LIVERMORE SOLVER FOR ORDINARY DIFFERENTIAL EQUATIONS (LSODAR) and CVODE[9] were compared. DASSL and LSODAR were selected because they are the default solvers of DYMOLA and performed well in Liu et al.'s studies [151]. The solver CVODE is included as the implementation comes with a version optimized for sparse systems, which is automatically selected by DYMOLA based on the structure of the system. We derive the following findings from the results of the performance test.

- The dynamic scheduling approach outperforms the fixed time step scheduling approach for all investigated solvers and all numbers of boards.
- The sparse version of CVODE is selected by DYMOLA for systems with 100 or more boards. However, the solver fails to simulate the model if the number of boards is above 100. The reason for this cannot be determined with certainty, but the error messages suggest that there is a problem with the event handling.
- LSODAR seems to be the best-performing solver for large-scale systems including high numbers of events; however, it also fails to simulate the system with 500 boards. Again, it seems to be a problem of the event handling.
- DASSL is the only solver capable of simulating the system with 500 boards. Besides, it is the only solver that can handle the fixed time step scheduling approach for larger systems; however, its performance is lower compared to LSODAR.
- The conditions $time > t_c$ are processed by the solvers not as a TIME EVENT but as a STATE EVENT.

9 Implementation of VARIABLE-COEFFICIENT ODE SOLVER (VODE) in C.

Table 5: Performance test comparing the dynamic scheduling approach (DS) with the fixed time step scheduling (FS) approach. All simulations are performed in Dymola 2018 64bit using Visual C++ 2015 Express Edition (14.0) and standard configurations for solvers.

Number of Boards [-]	***Exp. Sim. Time*** [s]	***Solver***	***Scheduler***	***Wall Time*** [s]	***Ratio** (Sim. T. / Wall T.)* [s/s]	***State Events*** [-]	***Time Events*** [-]
10.00	200.00	dassl	DS	0.14	1408.45	315.00	0.00
			FS	1.04	192.31	146.00	2053.00
		lsodar	DS	0.19	1025.64	315.00	0.00
			FS	0.72	277.01	146.00	2066.00
		cvode	DS	0.08	2500.00	315.00	0.00
			FS	0.30	664.45	146.00	2050.00
100.00	2000.00	dassl	DS	6.23	321.03	3031.00	0.00
			FS	21.40	93.46	1526.00	20 046.00
		lsodar	DS	6.48	308.64	3073.00	0.00
			FS	14.90	134.23	1578.00	20 047.00
		cvode	DS	7.28	274.73	3966.00	0.00
			FS	fails	fails	fails	fails
200.00	4000.00	dassl	DS	26.70	149.81	6099.00	0.00
			FS	115.00	34.78	3084.00	40 059.00
		lsodar	DS	21.40	186.92	6153.00	0.00
			FS	fails	fails	fails	fails
		cvode	DS	fails	fails	fails	fails
			FS	fails	fails	fails	fails
300.00	6000.00	dassl	DS	70.50	85.11	9160.00	0.00
			FS	581.00	10.33	4618.00	60 065.00
		lsodar	DS	49.70	120.72	9241.00	0.00
			FS	fails	fails	fails	fails
		cvode	DS	fails	fails	fails	fails
			FS	fails	fails	fails	fails
400.00	8000.00	dassl	DS	157.00	50.96	12 227.00	0.00
			FS	923.00	8.67	6166.00	80 058.00
		lsodar	DS	95.80	83.51	12 343.00	0.00
			FS	fails	fails	fails	fails
		cvode	DS	fails	fails	fails	fails
			FS	fails	fails	fails	fails
500.00	10 000.00	dassl	DS	296.00	33.78	15 313.00	0.00
			FS	1840.00	5.43	7706.00	100 059.00
		lsodar	DS	fails	fails	fails	fails
			FS	fails	fails	fails	fails
		cvode	DS	fails	fails	fails	fails
			FS	fails	fails	fails	fails

```
model MotivatingExample
[...] // nonrelevant code is excluded to enhance readability
equation
// board trajectory
x_s_board = a*(time-t_last_event)^3
            + b*(time-t_last_event)^2
            + x_board_last_event;
[...]
when {(time > pre(t_c)), (x > pre(x_c))} then
  agent_state = pre(agent_state_next);
  (t_c, agent_state_next, x_c, a, b) = ActionOnBoard(
     x, y, v_x, v_y, x_board, time, pre(agent_state_next), b_l, b_r);
   x_board_last_event = x_board;
   t_last_event = time;
end when;
[...]
end MotivatingExample;

function ActionOnBoard
  input Real x "x position of ball";
  input Real y "y position of ball";
  input Integer state "state";
  [...]                                    // further input variables
  output Real t_c "event condition 1";
  output Real x_c "event condition 2";
  output Real a "parameter of set point of trajectory";
  output Real b "parameter of set point of trajectory";
  output Integer next_state "next state";
algorithm
  if state == 0 then
    [...]                                  // state-depended behavior
  elseif  [...]
    [...]
  end if;
end ActionOnBoard;
```

Listing 4.1: MODELICA implementation of the motivating example.

```
model MotivatingExampleAlternative
parameter Real delta_t;
[...]
equation
[...]
when (sample(0, delta_t)) then
    x_s_board = ActionOnBoardAlt(x, x_board, b_l, b_r);
end when;
[...]
end MotivatingExampleAlternative;
```

Listing 4.2: Code Fragment for *Adapter* using fixed time steps.

4.4 IMPLEMENTATION DETAILS

As described in the previous section, in theory it would be possible to represent the concepts of MOCES as described in Sections 4.1 and 4.2 in MODELICA solely. However, this is not commended, as MODELICA is designed to model SYSTEMS that can be described by ordinary DAES and MODELICA IDES using SOLVERS which have been optimized for DAES have serious issues to handle large models with many events as described in Section 3.3.3.

MOCES thus uses DAE-based models under the usage of the modeling language MODELICA only for the PHYSICAL LAYER. For the BUSINESS PROCESS LAYER, it employs an agent-based modeling approach as presented in Section 3.4 implemented from scratch in C++. The INFORMATION LAYER is also implemented in C++, and it makes use of a relational database.

4.4.1 *Extending a Modelica Model by an Agent*

MOCES supplements the MODELICA model that realizes the *Physical Node* with a C++ class that implements the *Business Node*. This fundamental structure is given in Figure 41.

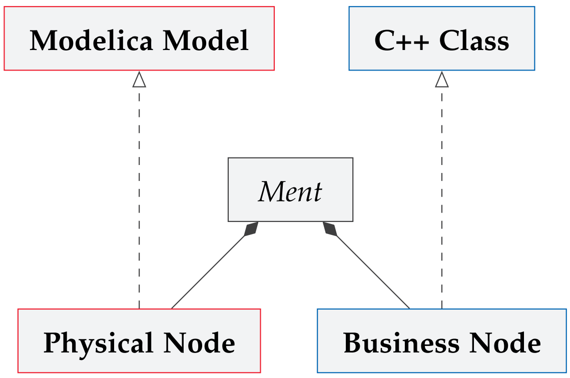


Figure 41: Basic implementation structure of a MENT.

One design decision of MOCES is to keep the inheritance structures of the MODELICA models and the C++ classes in parallel. This is sketched in Figure 42. As a result, every MODELICA model that represents the *Physical Node* part of a type of MENT is linked to exactly one C++ class representing the *Business Node* part of that MENT. Another design decision, also shown in Figure 42, is that there are three levels of hierarchy. The models and classes of the two upper levels are abstract. On the uppermost level there is exactly one abstract base model and class. The abstract models of the second level describe different types of MENTS and serve to simplify the implementation of concrete types of MENTS

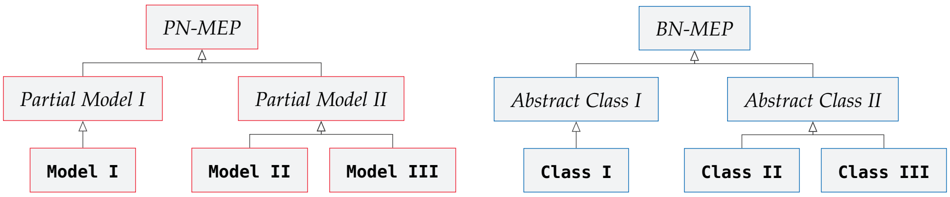


Figure 42: Parallel class structure of the MODELICA models and the C++ classes used by MOCES. *PN-MEP* and *BN-MEP* are abbreviations for *Physical Node - Moces Entity Partial* and *Business Node - Moces Entity Partial* respectively.

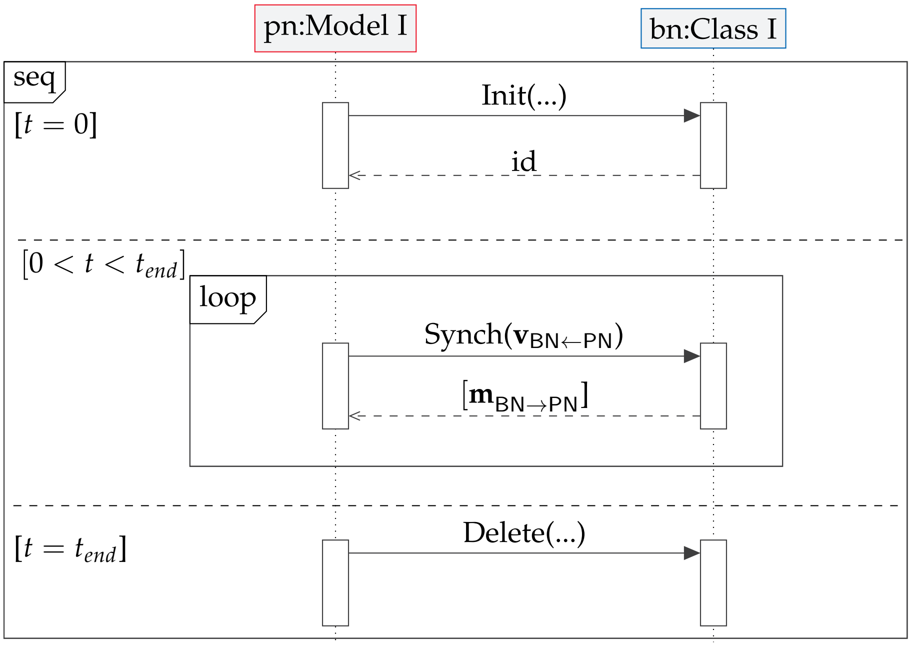


Figure 43: Fundamental interactions between *Physical Node* (pn:Model I) and *Business Node* (bn:Class I) during simulation ($0 < t < t_{end}$), its initialization phase ($t = 0$), and termination phase ($t = t_{end}$). The notation of the variables in the simulation phase aligns with the notation introduced in Section 4.3.

that share certain characteristics, e.g., the characteristics of a consumer or producer of electrical energy.

To link precisely one instance of each class to the corresponding MODELICA model instance, MOCES implements a concept similar to the *external objects* language feature of MODELICA, see MLS [120, p. 175]: objects with internal memory are dynamically initialized during the initialization phase of the simulation by a function call. During simulation, functions can be used to communicate with these objects. At the end of the simulation, the objects are destroyed. The basic idea and the resulting interactions are given in Figure 43.

```
// BNE_businessnodeexample.h
class BNE_BusinessNodeExample {
 public:
  BNE_BusinessNodeExample(int init_parameter);        // constructor
  ~BNE_BusinessNodeExample();                         // destructor
  void Synch(double v_0, double& m_in, double& m_c);  // synchr. (Adapter)
 private:
  int local_state;
}
```

Listing 4.3: Pattern used to implement a *Business Node* in C++.

From the MODELICA perspective, an external object is *opaque* as the inner structure is not known. This is basically a very helpful characteristic, as the complexity of the *Business Node* does not enlarge the model to be handled by the MTP and has therefore no adverse influence on the required translation time;

Since MODELICA does not provide a direct interface for C++, its C interface must be used. In the context of MODELICA, functions implemented in C and called by a MODELICA model are named *external functions*, cf. MLS [120, p. 165]. External functions are thus external routines invoked by the model.

A short code example best describes the implementation pattern used by MOCES to extend a MODELICA MODEL by an AGENT. For illustrative purposes, we limit ourselves to a minimum number of exemplary variables. An AGENT respectively a *Business Node* is declared in C++ by the pattern given in Listing 4.3.

The C wrapper that is invoked by the MODELICA model as an external function is declared and implemented as specified in Listing 4.4. The C wrapper introduces a handle (`id`) for every object. Since the C interface of MODELICA does not offer the possibility to exchange a handle in the form of a pointer or similar, the position within the storing vector `bne_vec` is used as the handle instead.

The three functions declared by the C wrapper can be used by the *Physical Node* to implement the interactions with the *Business Node* visualized in Figure 43. This is shown in the code fragment given in Listing 4.5. The functions `BNE_Init` and `BNE_Delete` are not explicitly listed; they follow the same pattern as `BNE_SYNC`.

```
// mocesinterface.h
extern "C" {
  int BNE_Init(int init_parameter);
  void BNE_Delete(int id);
  void BNE_Synch(int id, double v_0, double* m_in, double* m_c);
}

// mocesinterface.c
// vector holding references to Business Nodes
std::vector<BNE_BusinessNodeExample*> bne_vec;

int BNE_Init(int init_parameter) {
  bne_vec.push_back(new BNE_BusinessNodeExample(init_parameter));
  return(bne_vec.size() - 1);
};
void BNE_Delete(int id) {
  BNE_BusinessNodeExample* ptr = bnv_vec.at(id);
  delete ptr;
};
void BNE_Synch(int id, double v_0, double* m_in, double* m_c) {
  bne_vec.at(id)->Synch(v_0, m_in, m_c);
};
```

Listing 4.4: C wrapper used along with C++ pattern of *Business Node*, cf. Listing 4.3.

```
package Example
[...]
  function BNE_Synch
    input Integer id "unique id of BN/Ment";
    input Real v_0 "variable given to BN/Ment";
    output Real m_in "variable given to PN/Ment";
    output Real m_c "event condition given to PN/Ment";
    external "C" BNE_Synch(id, v_0, m_in, m_c);
  end BNE_Synch;

  model PhysicalNodeExample
    parameter Integer init_parameter=42;
    discrete Integer id "id of BN/Ment";
    discrete Real m_in;
    discrete Real m_c(start = 0.5);
    Real v_0;

  equation
    when initial() then
      id = BNE_Init(init_parameter);
    end when;
    v_0 = der(cos(time));
    when (pre(m_c) > v_0) then
      (m_in, m_c) = BNE_Synch(id, v_0);
    end when;
    when terminal() then
      BNE_Delete(id);
    end when;
  end PhysicalNodeExample;
end Example;
```

Listing 4.5: Modelica pattern used to implement the interaction with a *Business Node*.

4.4.2 *The Abstract Base Class MEP*

Except for the *Ment* `System`, all *Ments*, meaning *Business Nodes* and *Physical Nodes*, are derived from the abstract base class respectively partial model `MEP`. This base class defines certain basic parameters and implements the *Adapter* and the *Business Interface* that enables communication between *Business Nodes*. The simplified class diagram is shown in Figure 44. The `PN-MEP` inherits the *Adapter* that will be discussed in detail in Section 4.4.7.

The virtual method `PutIntoInbox(...)` implements the *Business Interface*. It is in the responsibility of the class extending the base class how to handle arriving messages, but typically they are merely put on a message deque and processed independently. While the `Synch(...)` method implements the generic *Adapter* of the `BN-MEP`, the two other methods (`Observe(...), LogMeasurement(...)`) are related to specialized types of Adapters: the *Business Node Observer* and the *Meter*. The parameters of the specialized *Adapters* (`log_id, ob_id`) are used as handles to enable the observation respectively metering of various variables.

4.4.3 *Handling of Time- and Position-Dependent Data*

The handling of time and time- and position-dependent data is part of the INFORMATION LAYER of MOCES, cf. Figure 33. Before going into details

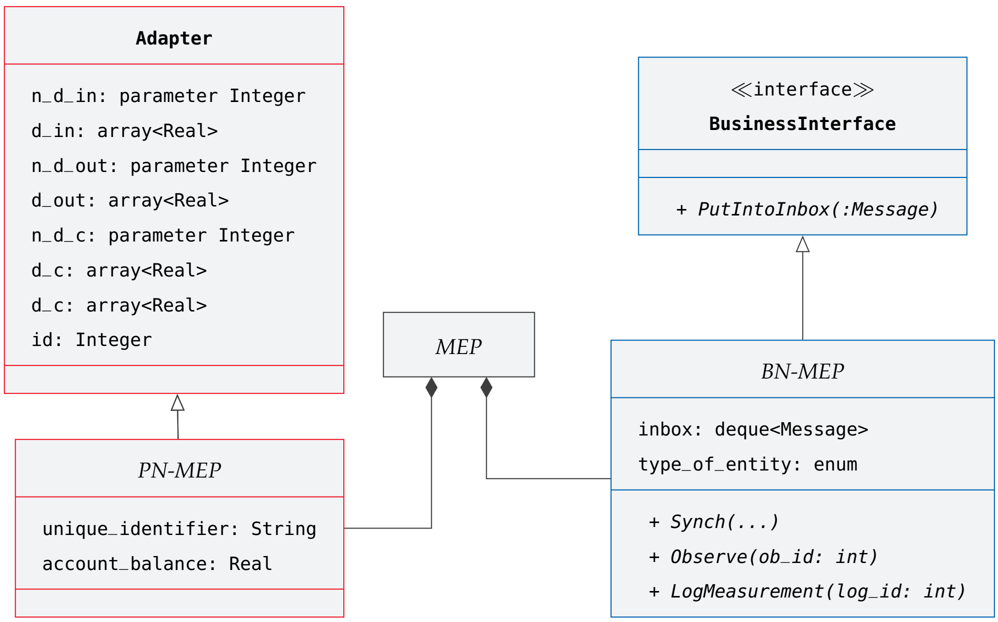


Figure 44: Simplified class diagram of the abstract base class `MEP` showing the most important attributes and methods.

of the implementation in Sections 4.4.4 and 4.4.5, we will first provide some basic background considerations.

TIME We would like to discuss how *time* is handled in MOCES and emphasize the importance of a plain concept.

As we know, "time is the indefinite continued progress of existence and events that occur in apparently irreversible succession from the past through the present to the future" (Cited after [152]). Mankind invented devices to measure this progress, we call them *clocks*, and defined standards for TIME INTERVALS and systems to synchronize the clocks such as the COORDINATED UNIVERSAL TIME (UTC). While definitions of TIME INTERVALS are required to agree on a duration of time, synchronized clocks are required to agree on a specific *point in time*.

Further on, we will use the terms WALL CLOCK and WALL TIME if we refer to this *true* time, whose progress we cannot bias. For the *time* that is valid *inside* a MODEL, we use the terms SIMULATION CLOCK and SIMULATION TIME. From an OBSERVER's perspective, the progress of the SIMULATION TIME does not have to progress continuously; typically it is squeezed or expanded. In most cases, the SIMULATION CLOCK is initialized at the beginning of the simulation run with zero, and the SIMULATION TIME proceeds continuously from the models perspective.

Things get a bit complicated if the model's behavior depends on a point in time of the WALL CLOCK, such as a model of the sun position, or on a characteristic of this point in time, e.g., that it is a Sunday or Christmas Eve.

Since models in the field of ENERGY SYSTEMS often have this type of dependency, the SIMULATION TIME used in MOCES always has a defined relation to the WALL TIME. Technically, this is done by anchoring the SIMULATION TIME t_{sim} to an arbitrary calendar date and providing an interface to it that is accessible by all MENTS. In other words, a MOCES model always includes a MODEL of the WALL CLOCK. The individual *Business Nodes* can, in turn, introduce further SIMULATION CLOCKS in reference to this clock if needed.

TIME- AND POSITION-DEPENDENT DATA The behavior of many MENTS depends on ambient conditions that are present at their locations; e.g., the power output of a PV plant mainly depends on the solar irradiation. To allow an easy modeling of such MENTS and to include the impact of these relations in the simulation, MOCES offers a minimalistic, database-based GLOBAL INFORMATION SYSTEM (GIS). It enables an easy access to weather and other data from within every model by just using the geographic location and point in time. This approach also ensures that the boundary conditions are consistent and easily exchangeable for all models.

In general terms, the interface provided by the INFORMATION LAYER of MOCES can be formulated as follows:

$$v = f(v_{id}, p_{\text{lat}}, p_{\text{long}}, p_{\text{mosl}}, t^*); \quad (4.11)$$

the interface for values depending on the time solely has the following formulation:

$$v = f(v_{id}, t^*), \quad (4.12)$$

with:

v_{id} unique identifier for value of interest,
t^* time of interest,
p_{lat} latitude,
p_{long} longitude,
p_{mosl} meters above sea level.

Both interfaces are implemented by a component that exists exactly once in each MOCES model. From the model's perspective and depending on the relation between the current SIMULATION TIME t_{sim} and t^*, the values provided are either current values ($t^* = t_{\text{sim}}$), lie in the future ($t^* > t_{\text{sim}}$) or in the past ($t^* < t_{\text{sim}}$).

Access to values of the future should be prohibited, as these values are not yet known. However, we decide to allow *Business Nodes* to access values from the past *and* future to enable an easier modeling of *Ments* looking into the future. An example is given in Section 5.4.4 which describes a model providing forecasts to other *Ments*. It is therefore the responsibility of the modeler to decide how to deal with the possibility to access data of the future. A *Physical Node*, in contrast, only has access to the current values in MOCES. However, this restriction could be removed if deemed necessary and useful.

4.4.4 *Spanning the Business Process Layer and the Information Layer*

The *Ment* `System` initializes the BUSINESS PROCESS LAYER and the INFORMATION LAYER at the beginning of the simulation. As Figure 45 shows, the *Ment* `System` uses the pattern described in Section 4.4.1. The `System` spans the information layer by implementing the interface for location- and time-dependent data described in Section 4.4.3 and by providing an anchored SIMULATION TIME respectively model of the WALL TIME with `m_epochs`.

To span the *Business Layer*, the *Business Node* of the *Ment* `System` implements two registries; the first manages the addresses of all *Business Nodes*, the second all *Meters*. These two registries enable communication between the *Business Nodes*; to do this, all *Business Nodes* must register with the *System*, as shown in Figure 45 during simulation initialization.

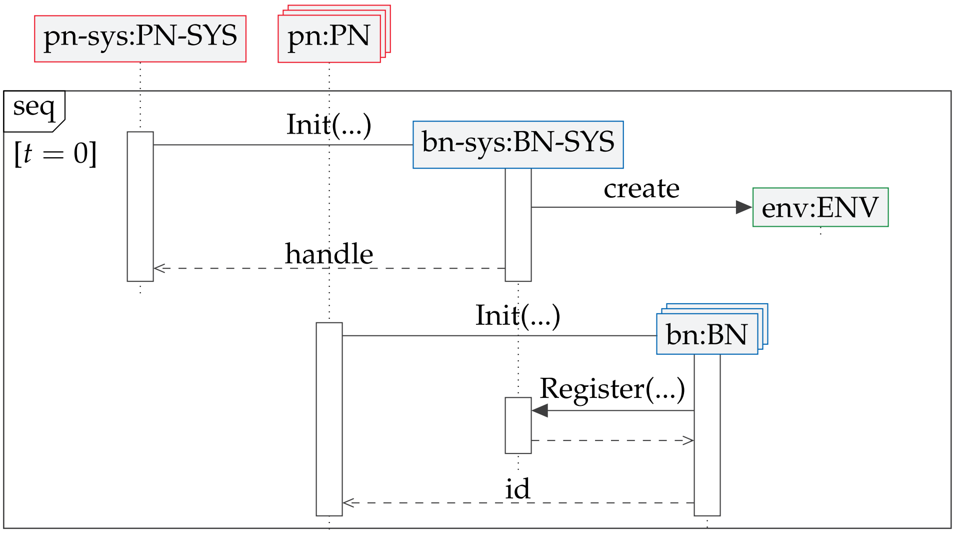


Figure 45: Basic interaction at $t = 0$ to initialize the BUSINESS PROCESS LAYER and the INFORMATION LAYER and to assign unique ids to the MENTS, see Figure 46 for the used abbreviations.

Figure 46 gives an overview of the basic implementation structure of the *Ment* `System`. Note that the diagram is not complete but focuses on the essential characteristics. The classes/models shown in red are implemented in MODELICA, those in blue and green in C++. Due to its central task, the `PN-SYS` model is a mandatory component of every MOCES scenario model, which contains exactly one such instance, and makes its variables available globally by using the inner/outer language construct of MODELICA, cf. MLS, [120, p. 175].

The `PN-SYS` is also the central entity for configuration. As the parameters in Figure 46 show, besides the start date of the simulation (`start_date`), the data basis of the INFORMATION LAYER can be configured (`weather_data`, `data`). Besides, logging and debug configurations can also be carried out.

The interfaces to the INFORMATION LAYER (`GetWeather()`, `GetData()`) are realized as independent models. In this way, they can be easily included into *Physical Nodes* to provide them location- and time-dependent values.

The core functionality of the two registries, which are implemented and made available by `BN-SYS`, is the provision of information about the structure of the system and the relations between metering points, MENTS, and BRPS which are required to model the BMP as presented in Section 2.2.8. This kind of information is made available through methods. `GiveMentIDByName()`, for example, allows all *Ments* to search for the handle of a *Ment* for which a user-defined identifier is known. In the current implementation, the registries are implemented by unordered maps[10]. Although the objects implementing the environment (EnvWeather and EnvData) are connected to the BN-SYS by composition, they belong to the INFORMATION LAYER and are thus discussed in Section 4.4.5.

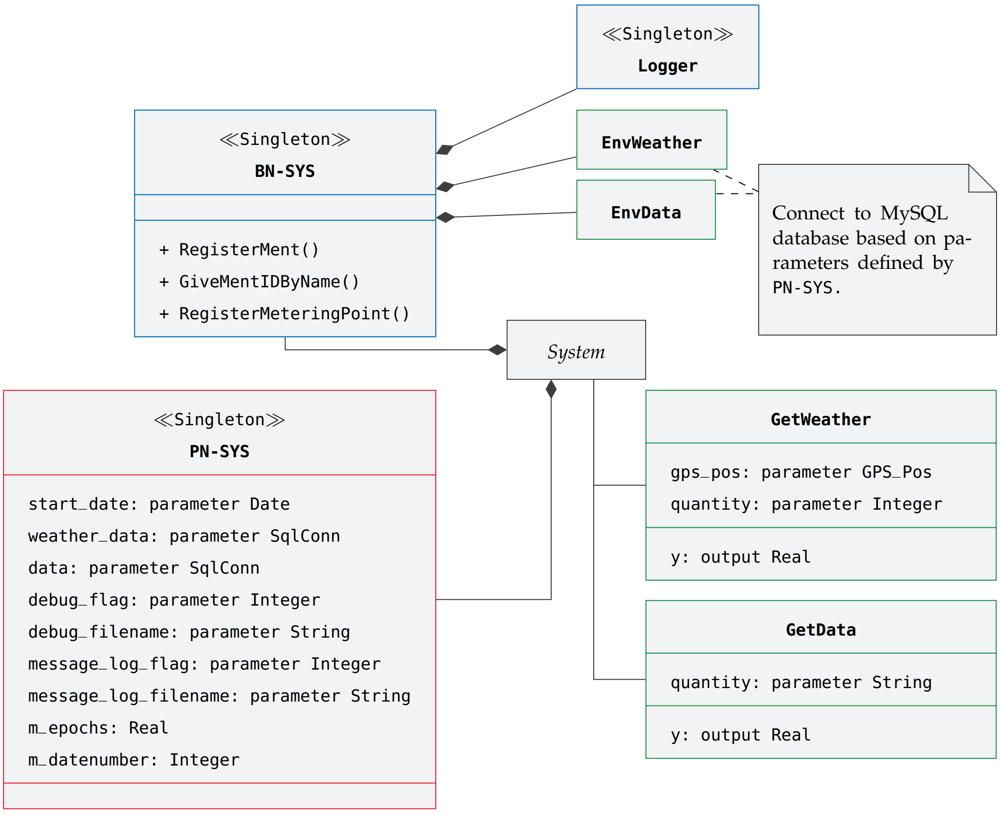


Figure 46: Simplified class diagram of the MENT System showing crucial elements.

4.4.5 *Information Layer*

The basic structure of the implementation of the INFORMATION LAYER providing the interfaces as discussed in Section 4.4.3 composes of three parts: a MySQL database providing the underlying data; EnvWeather and EnvData classes that interpolate and buffer the data; and, the GetWeather and GetData MODELICA models wrapping the functionality of EnvWeather respectively EnvData in Modelica.

As MOCES uses measured weather data from specific weather stations, the database schema used for providing the weather data consists of two tables. The first table stores information about the measuring stations, mainly the location and the provided meteorological measurands.

10 See http://www.cplusplus.com/reference/unordered_map/unordered_map/(lastaccessed:3June2021) for reference.

```
// key value map buffering weather data (object attribute)
unordered_map<Identifier, TimeSeries> weather_map_

TimeSeriesPointWithTimespan EnvWeather::GetValueAndNextEvent(
    const simtime& time,
    const TypeOfClimateValue& cv,
    const GPSPos& gps_pos) {
  // check if gps_pos cv combination is in weather_map_
  // if not available
  // > add to key value map weather_map and buffer sql data
  it = weather_map_.find(Identifier(gps_pos, cv));
  if (it == weather_map_.end()){
    it = AddElementToWeatherMap(time, cv, gps_pos);
  }
  // if current time is larger than buffered time series
  // > update time series
  if (!it->second.IsInRange(time)){
    it = UpdateElementInWeatherMap(it, time);
  }
  // return value (parameters of piecewise linear functions)
  return it->second.get_time_series_point_modelica(time);
};
```

Listing 4.6: Code fragment to describe the implementation of the environment providing weather information.

The second table provides the actual data by using a key-value pattern with the key being composed of the station identifier, the measurand identifier, and a time stamp. This pattern was chosen for its flexibility, as it allows the subsequent insertion of new stations, measurands and data. Furthermore, it is also flexible about the temporal resolution of the measurands; those may vary even within a measurement series.

The behavior and functionality of the `EnvWeather` class is best described by the code fragment of the method `GetValueAndNextEvent`, see Listing 4.6, that is served as an external function via the C wrapper to Modelica.

At its core, the `EnvWeather` class creates a time series for each requested combination of measurand and location. The time series is created on the basis of the values available in the database and a suitable local and temporal interpolation method. In the simplest case, a linear interpolation is used for the temporal interpolation, and a 1-nearest neighbor interpolation is used for the spatial interpolation. As indicated by the code fragment, the time series are created in batches for a certain period of time and are only renewed if the current simulation time lies outside the period of time covered by the prepared time series. The main reason for this procedure is its significantly higher performance compared to a

Table 6: Summary of weather data provided by MOCES by default.

Measurand	Number of Stations	Time Covered	Temporal Resolution
Wind Speed	244.00	2013 – 2015	1.00 h
Wind Direction	244.00	2013 – 2015	1.00 h
Air Temperature	474.00	2013 – 2015	1.00 h
Global Horizontal Irradiation	47.00	2013 – 2015	1.00 h
Global Diffuse Irradiation	47.00	2013 – 2015	1.00 h

solution in which the required data is queried directly from the database each time `GetValueAndNextEvent` is called.

The `EnvWeather` class also provides the position of the sun, precisely the solar azimuth angle and zenith angle. Contrary to the weather data, these values are calculated on-the-fly using NREL's solar position algorithm, cf. Reda and Andreas [153].

Figure 47 illustrates how the component can be used by a modeler in MODELICA. It shows its model icon and the dialog used to select the needed measurand. The implementation is designed to enable an easy

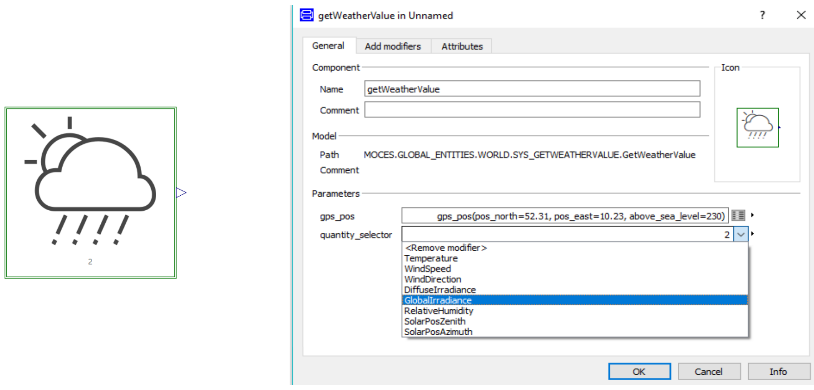


Figure 47: Model icon of `GetWeather` model (left) and its dialog showing the parameters (right).

integration of existing weather data by adding the data into the database. By default, MOCES provides weather data based on data from more than 500 weather stations made available by the DWD[11]. The data set covers the years 2013–2015; details are given in the Table 6. The measurement values are used as provided without any modification, only obviously corrupt time series are not used.

11 `https://www.dwd.de/EN/ourservices/opendata/opendata.html` (last accessed: 3 June 2021).

Table 7: Excerpt of time series provided by MOCES. Time series marked with * are added for validation purposes, cf. p. 196 for a usage example.

Data	Time Covered	Temporal Resolution	Source
Hard Coal Price Index	2010-01 – 2016-06	1 month	[154]
Natural Gas Price Index	2010-01 – 2016-06	1 month	[154]
Lignite Price Index	2010-01 – 2016-06	1 month	[154]
Oil Price Index	2010-01 – 2016-06	1 month	[154]
Total Load Germany*	2013-01 – 2015-12	1.00 h	[155]
Total PV Feed-In Germany*	2014-01 – 2014-12	15.00 min	[156]
Total PV Feed-In Tennet*	2014-01 – 2014-12	15.00 min	[156]
Total Wind Feed-In Germany*	2014-01 – 2014-12	15.00 min	[156]
EPEX Spot Day-Ahead Auction DE/AT*	2012-01 – 2016-08	1.00 h	[156]

In addition to the weather data, MOCES provides a number of other time series, which can be accessed via the `EnvData` component. An excerpt of these time series is given in Table 7.

HANDLING OF SCENARIOS By default, MOCES only supplies data from the past. If MOCES is to be used to investigate scenarios with different boundary conditions, the authors recommend to provide a separate database for each scenario. The selection of the different scenarios can then be made via the parameters `weather_data` and `data` of the MENT `System`, cf. Figure 46. This ensures that all MENTS refer to a consistent environment.

4.4.6 *Physical Layer*

As stated, MOCES uses MODELICA to describe the *Physical Nodes* and their interactions. With regard to the holistic description of the POWER SYSTEM given in Section 2.2, the interactions that take place in the PHYSICAL LAYER reflect the SYSTEM ACTOR perspective on the POWER SYSTEM. For intuitive modeling, MOCES provides a series of *Base Models* on the basis of which models for specific scenarios can be derived easily. An excerpt of these base models and their relations to each other is given in Figure 48; as indicated, the models of the two upper levels are partial. The class diagram shows that currently most model types are derived from the partial model *`Prosumer`*. Models derived from this partial model assume at least the role of a PCG as introduced in Section 2.2.7. The models derived from the other partial models typically have no interaction with other *Physical Nodes* on the PHYSICAL LAYER; their main interactions take place on the BUSINESS PROCESS LAYER. We therefore concentrate on the type *`Prosumer`* in this section.

The partial model *`Prosumer`* extends *`PN-MEP`* by two mandatory parameters: `gps_position` defining the position of the model, and `name_of-`

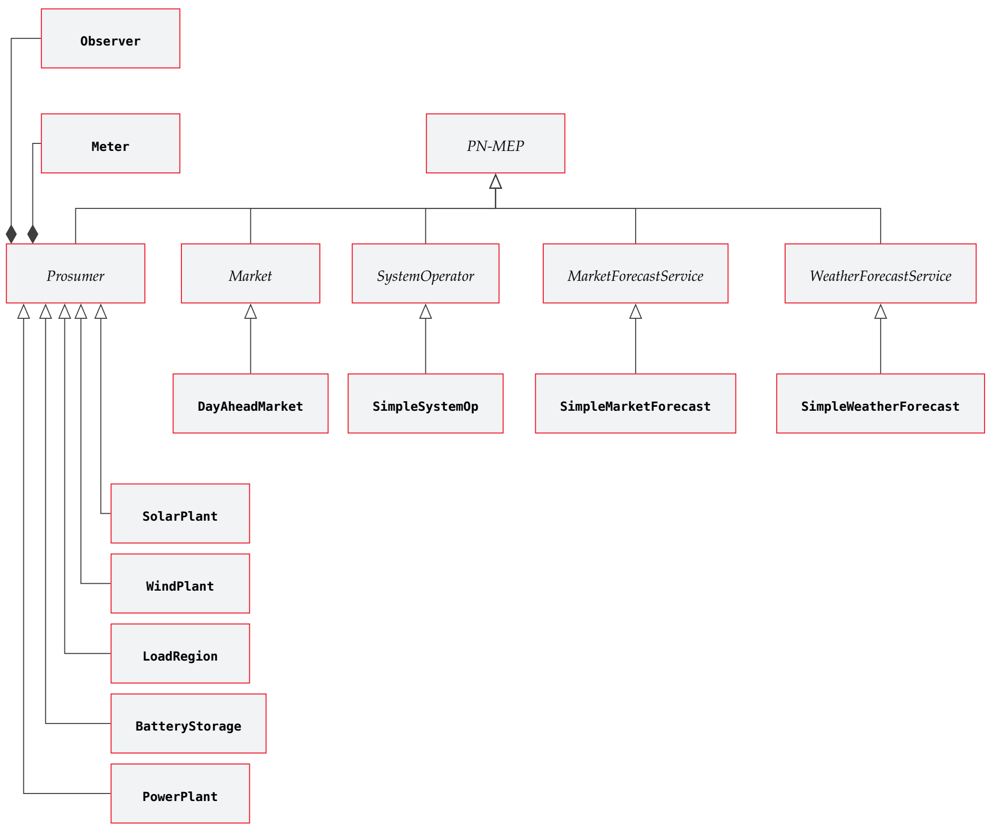


Figure 48: Base MODELICA models provide by MOCES to model the *Physical Node* part of a MENT.

`_aggregator` assigning the model to a BALANCE GROUP. This assignment is required to model the BMP as described in Section 2.2.8. The term *aggregator* is a synonym for the ROLES BRP and BALANCE SUPPLIER, which are not distinguished in MOCES. The assignment to a BALANCE GROUP also applies to the associated `Meter`, cf. Figure 48, that is used to determine and record the energy consumption or production of the prosumer and enables the modeling of the *settle imbalance phase* of the BMP, cf. Section 2.2.8.3.

The component `Meter` realizes the special *Adapter* type *Meter*. In theory, implementing an energy meter in MODELICA is straightforeward, as only the integral of the power flow to be measured has to be formed. With large models, however, this procedure leads to performance issues, since each meter introduces a state variable. Therefore, MOCES uses the synchronous language features of MODELICA, see MLS [120, p. 211], to determine the energy values based on sampled power values.

A design decision of MOCES is that each `Prosumer` holds at least *three* time series which represent:

- the actual energy consumption/production measured by the linked `Meter`,

- the forecasted energy consumption/production, and
- the scheduled energy consumption/production sent to the SOP as requested by the BMP.

The latter two time series are implemented in the *Business Node* part of a MENT. However, the current value can be accessed by an `Observer` as given in Figure 48. This model implements the *Adapter* type *Observer*. Especially MENTs that, under standard operating conditions, follow a given SCHEDULE, such as conventional power plants or battery storage systems, use this adapter to adopt the SCHEDULE as set point.

MINIMAL EXAMPLE A minimal example for a complete model of a *Physical Node* in MOCES derived from the partial model *`Prosumer`* is given in the following Listing 4.7:

```
model LBP_LoadByParameter
  import TY = MOCES.TYPES;
  extends MPP_ProsumerPartial;
  parameter Init_LBP init_LBP;
  TYPES.ErrorCode er_01_LBP;
  TYPES.ErrorString er_me_01_LBP;

equation
  el_terminal = init_LBP.load;

  when initial() then
    (id, er_01_LBP, er_me_01_LBP) =
          LBP_Init_IINM(init_MEP,
                        init_MPP,
                        init_LBP,
                        CONSTANTS.kLengthErrorMessage,
                        sys_system.Init_Sequence);
    assert(er_01_LBP > 0, er_me_01_LBP);
  end when;
end LBP_LoadByParameter;
```

Listing 4.7: Minimal example to implement a *Physical Node* in Modelica. `el_terminal` is the electrical connector of the model.

The listing describes an electrical load whose power is specified via the parameter `init_LBP.load`[12]. As the code example shows, this MODELICA model does not directly specify the behavior of the linked *Business Node* and there is no way to do so. How to model this behavior is discussed in Sections 4.4.8 and 4.4.9. However, by inheriting from the abstract model *`Prosumer`*, which in turn inherits from the abstract model *MEP*, the behavior of the respective connected business nodes is adopted as well. For

12 We use the MODELICA notation here, `init_LBP.load` is an element of the struct `init_LBP`.

the model shown here, this includes among other things the registration of the metering point and the daily transmission of time-resolved consumption data.

4.4.6.1 *Running Example*

As a further example, we will describe the modeling of an energy storage system in MOCES. The model is named `BatteryStorage` and inherits from the partial model *`Prosumer`*, cf. Figure 48. We will return to this example when describing the modeling of the behavior of the linked *Business Node*, cf. Section 4.4.9.1.

More specifically, we present the *Physical Node* of a MENT who pursues the strategy of purchasing and storing electricity in times of favorable prices and reselling this electricity in times of high prices. It is assumed that this strategy is provided to the *Physical Node* in form of a schedule $p_{\text{schedule}}(t)$ by the *Business Node*. To enable the *Business Node* to determine a feasible schedule, the *Physical Node* communicates the current state of charge of the storage system to the *Business Node*.

A simple, yet non-linear model, of the storage system and its physical behavior can be described with the following set of equations:

$$\frac{dE}{dt} = \begin{cases} \eta_{\text{charge}} \cdot p(t) & \text{if } p(t) > 0 \\ \frac{1}{\eta_{\text{discharge}}} \cdot p(t) & \text{if } p(t) \leq 0 \end{cases}, \tag{4.13}$$

$$p(t) = \begin{cases} p_{\text{schedule}}(t) & \text{if } 0 < E < E_{\text{capacity}} \\ \min\left(0, p_{\text{schedule}}(t)\right) & \text{if } E \geq E_{\text{capacity}} \\ \max\left(0, p_{\text{schedule}}(t)\right) & \text{if } E \leq 0 \end{cases}, \tag{4.14}$$

with:

E	energy stored in storage system,
E_{capacity}	capacity of storage system,
$p(t)$	power flow into storage system,
$\eta_{\text{discharge}}$	discharge efficiency of storage system,
η_{charge}	charge efficiency of storage system.

The implementation in MOCES is given in the following Listing 4.8:

```
model SBS_SimpleBatteryStorage
  // definition of number of inputs/outputs/conditions
  extends MPP_ProsumerPartial(n_d_in=2, n_d_out=0, n_d_c=1,
                              Schedule(redeclare function GetValue =
                                       FUN.MEP_GetTSRampy_SPYC),
                              final observe_schedule = true,
                              mTP_LogMeasurement(measurement_value =
                                                 el_terminal);
```

```
9   parameter Init_SBS init_SBS;
10  Real SOC "state of charge";
11  // energy stored in battery
12  Modelica.SIunits.Energy E(start=init_SBS.cap_storage/2);
13  Modelica.SIunits.Power P "power flow into battery";
14  Real P_s "schedule for battery";
15
16 equation
17  // definition of input/outputs/conditions
18  SOC = d_in[2];
19  // physical behavior
20  P_s = Schedule.y;
21  el_terminal = P;
22  SOC = E/init_SBS.cap_storage;
23  if (E < init_SBS.cap_storage and E > 0) then
24    P = P_s;
25  elseif (E >= init_SBS.cap_storage) then
26    P = noEvent(min(0,P_s));
27  else
28    P = noEvent(max(P_s,0));
29  end if;
30
31  if noEvent(P>0) then
32    der(E) = init_SBS.eta_charge*P;
33  else
34    der(E) = P/init_SBS.eta_discharge;
35  end if;
36
37  when initial() then
38    [...]
39  end when;
40 end SBS_SimpleBatteryStorage;
```

Listing 4.8: Modelica code of battery storage system implemented in MOCES.

As the implementation shows, the model inherits from the model *Prosumer*, line 3–8. The `Adapter` is configured by the parameters `n_d_in`, `n_d_out` and `n_d_c`. Since the battery follows the schedule created by the *Business Node*, an `Observer` is activated for the schedule with `Schedule(...)` and `final observe_schedule = true`. A `Meter` is connected to the electrical interface with `mTP_LogMeasurement(measurement_value = el_terminal)`.

All parameters of the model, such as the storage capacity $E_{capacity}$, are wrapped by the struct `init_SBS`, cf. line 9. In the following lines 10–14, the variables needed to describe the storage system are introduced.

Line 18 links the state of charge with the *Adapter* and thus synchronizes it with the *Business Node*. Line 20 adopts the schedule as the target value. The remaining equation section is a straightforward implementation of Equations 4.13 and 4.14. The `noEvent` operator ensures that the relations are taken literally and do not trigger events, see [120, p. 30].

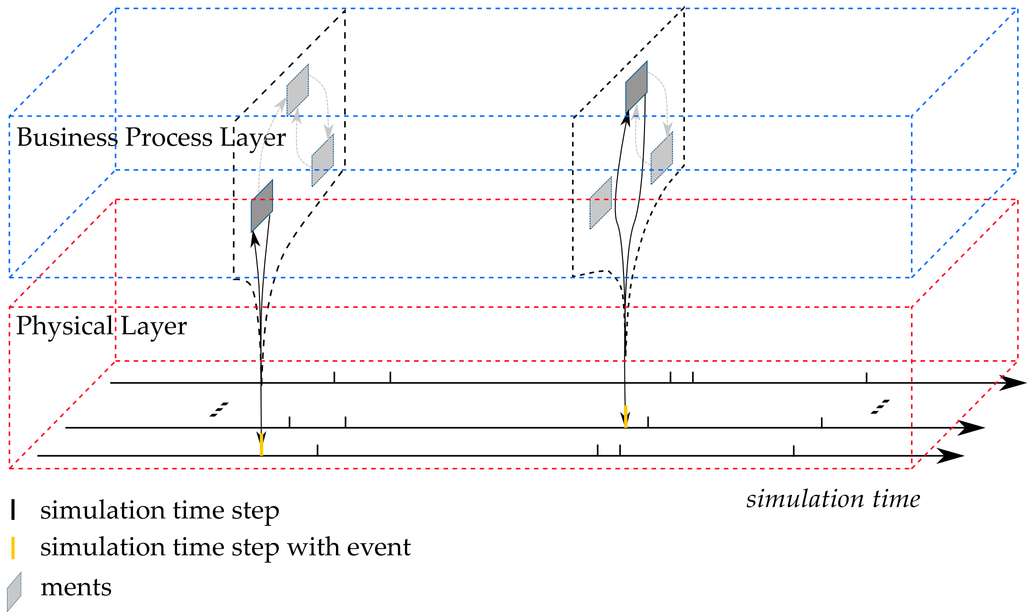


Figure 49: Overview of the synchronization concept used by MOCES. The synchronization performed by the *Adapter* is triggered by TIME EVENTS or STATE EVENTS. The black arrows visualize information exchange via the *Adapter*.

4.4.7 *Adapter*

The design of the *Adapter* that links the *Physical Node* of a MENT with its *Business Node* by means of synchronized exchange variables is critical. We have to ensure that the design and implementation does not violate the MLS and can thus be processed by any MODELICA IDE. As already sketched in the motivating example in Section 4.3, MOCES therefore uses a synchronization at TIME EVENTS or STATE EVENTS whose conditions are determined dynamically during runtime either by the *Physical Node* or the *Business Node*. As a useful side-effect, the concept implements the DYNAMIC SCHEDULING approach. The accepted drawback of this approach is that all interactions in the *Business Process Layer* are timeless. A conceptual overview to this approach is given in Figure 49.

In the following we would like to describe the concept and its implementation in MODELICA in detail. To this end, we would first like to introduce a few important terms and deal with the relevant properties of MODELICA.

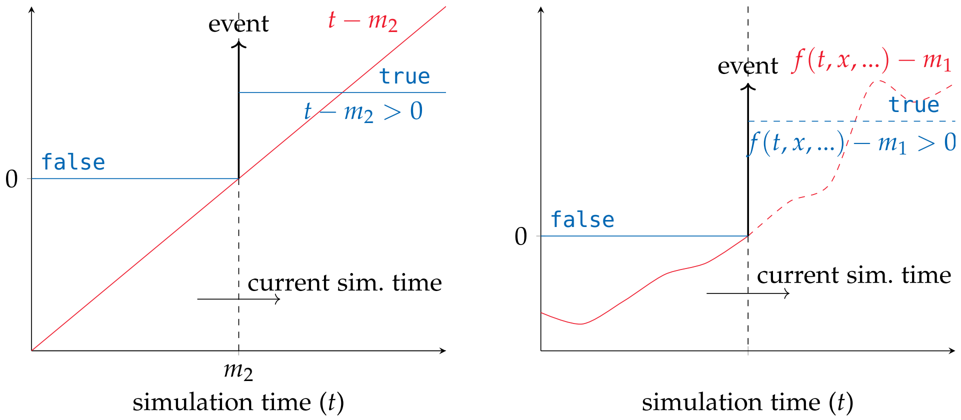
TIME AND STATE EVENTS In general, an EVENT occurs at a certain point in time and indicates that *something happens*; a concrete EVENT thus occurs exactly once. In the context of MODELICA and MOCES *something* *happens* is a synonym for an EVENT CONDITION becoming true. An EVENT CONDITION is formulated as a relational expression such as

$$c(t) := t > m_2, \quad t, m_2 \in \mathbb{R},\ c \in \{\texttt{true}, \texttt{false}\} \tag{4.15}$$

or

$$c(t) := f(t, x, \ldots) > m_1, \quad t, x, m_1 \in \mathbb{R},\ c \in \{\texttt{true}, \texttt{false}\}\ . \tag{4.16}$$

An EVENT is thus triggered if $c(t)$ shows a positive edge, meaning its value switches from `false` to `true`. In MODELICA, an EVENT CONDITION is converted into a *crossing function* by the MODELICA TRANSFORMATION PROCESS [157].[13] A crossing function here is a function that crosses zero when the condition becomes true. Depending on the type of variables included in the EVENT CONDITION, we distinguish between TIME EVENTS and STATE EVENTS.



(a) TIME EVENT (b) STATE EVENT

Figure 50: STATE EVENT VS. TIME EVENT. The *crossing function* is given in red; the value of the related relational expression in blue.

Equation 4.15 and its graphical representation in Figure 50a give an example for a TIME EVENT. The particular characteristic of TIME EVENTS is that their EVENT CONDITION contains only the time itself and constant parameters. The point in time at which the TIME EVENT occurs is thus known a priori, respectively can be calculated easily without the use of a root finding algorithm to determine zero crossing. In other words, the curve of the crossing function is known at time point t^* for the period $t^* > t$ as indicated by the solid curve type in Figure 50a.

Equation 4.16 and Figure 50b give an example for a STATE EVENT. Contrary to TIME EVENTS, the EVENT CONDITION of STATE EVENTS can include any type of variable. Detecting STATE EVENTS is thus harder, and the occurrence cannot be determined until the point in time of the event at

13 The rationale behind this transformation is that the root finding mechanisms of the DAE/ODE SOLVERS can be used to localize the EVENT.

the earliest. This characteristic is shown by the dashed curve in Figure 50b. MODELS including STATE EVENTS pose certain challenges for solving HDAE, e.g. events can easily be missed if an EVENT CONDITION switches twice within a single SOLVER time step; see Lundvall et al. [157] for a more detailed discussion.

PURE AND IMPURE FUNCTIONS The MODELICA language knows two different types of functions: PURE FUNCTIONS and IMPURE FUNCTIONS.

Following the current definition of the MODELICA specification a PURE FUNCTION "[...] always give the same output values [...] for the same input values and only the output values influence the simulation result, i.e. is seen as equivalent to a mathematical map from input values to output values. " (Modelica Association [120])

Functions that do not comply with the given definition, such as a function returning a random number, are impure. To ensure a correct handling of such functions, the MLS restricts their usage, see Modelica Association [120, p.148] for details. Relevant for MOCES is the possibility to call IMPURE FUNCTIONS at STATE EVENTS and TIME EVENTS. This restriction is mandatory to ensure the correct operation of the SOLVER. In this special case, it is ensured that the IMPURE FUNCTIONS are called once and that the solver restarts afterwards.

Pure as well as impure functions can have desired side-effects, meaning they modify a state outside the scope of the model described by the MODELICA model. Such desired side effects are typically described and implemented by using the external function interface of MODELICA. Typical side-effects are, e.g., the logging of variables in files or any other input/output operations. Of particular importance in MOCES are impure functions that influence the local or INTERNAL STATE of the *Business Node* connected to the *Physical Node*.

4.4.7.1 *Formal Description of Adapter*

To describe the *Adapter*, we develop a mathematical representation based on the general mathematical representation of a MODELICA model, cf. Appendix A.1 for details. A *Physical Node* is represented by a HDAE as follows:

$$\mathbf{c}(t_e) := \mathbf{f_c}\left(relation\left(\mathbf{v}\right)\right), \tag{4.17a}$$

$$\mathbf{m}(t_e) := \mathbf{f_m}\left(\mathbf{v}, \mathbf{c}\right), \tag{4.17b}$$

$$\mathbf{0} = \mathbf{f_x}\left(\mathbf{v}, \mathbf{c}\right), \tag{4.17c}$$

with $\mathbf{v} := [\dot{\mathbf{x}}\ \mathbf{x}\ \mathbf{y}\ t\ \mathbf{m}^-\ \mathbf{m}^+\ \mathbf{p}]^\intercal$ a vector of variables.

We introduce the conventional notation that calligraphic letters, such as $\mathcal{V}$, represent the set of variables of the respective vector, such as $\mathbf{v}$. To describe the *Adapter*, we define specific subsets: $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}}$ is the set of variables *communicated* from the *Physical Node* to the *Business Node*; from the *Physical Node*'s perspective, these variables are outputs. For $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}}$, it holds that $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}} \subset \mathcal{V} \setminus \mathcal{M}^{+}$. Further, we define $\mathcal{V}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ as the set of variables *communicated* from the *Business Node* to the *Physical Node*; from the *Physical Node*'s perspective, these variables are inputs. It holds that $\mathcal{V}_{\mathsf{BN}\rightarrow\mathsf{PN}} \subset \mathcal{M}^{+}$.

An upper index x on a set of variables, such as $\mathcal{V}^{x}_{\mathsf{BN}\rightarrow\mathsf{PN}}$, denotes a subset of $\mathcal{V}_{\mathsf{BN}\rightarrow\mathsf{PN}}$. We postulate that for any two subsets of $\mathcal{V}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ the following holds:

$$\mathcal{V}^{1}_{\mathsf{BN}\rightarrow\mathsf{PN}} \cap \mathcal{V}^{2}_{\mathsf{BN}\rightarrow\mathsf{PN}} = \emptyset. \tag{4.18}$$

This restriction ensures that the *Adapter* does not violate the single assignment rule of MODELICA, cf. MLS [120, p. 98]. Note that there is no such restriction on any subsets of $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}}$. In addition, it holds that the union of the respective subsets must result in the original set.

The *Adapter* is a function given with:

$$\mathbf{v}^{\text{adapter}}_{\mathsf{BN}\rightarrow\mathsf{PN}} := f^{\text{adapter}}_{\text{action}}(\mathbf{v}^{\text{adapter}}_{\mathsf{BN}\leftarrow\mathsf{PN}}). \tag{4.19}$$

The lower index 'action' indicates that the call typically triggers an action by the linked *Business Node*. $f^{\text{adapter}}_{\text{action}}(\cdot)$ is one element in the vector of functions $\mathbf{f_m}(\cdot)$, cf. Eqn. 4.17b. As indicated by the assignment operator :=, $\mathbf{f_m}(\cdot)$ is a vector of possibly IMPURE SIDE EFFECT FUNCTIONS; $f^{\text{adapter}}_{\text{action}}(\cdot)$ is generally of this type of function. Due to the restrictions on the usage of IMPURE SIDE EFFECT FUNCTIONS, we need a way to formulate the EVENT CONDITIONS triggering $f^{\text{adapter}}_{\text{action}}(\cdot)$. For this purpose, we further subdivide the set $\mathcal{V}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ into three subsets that do not intersect:

$$\mathcal{V}^{\text{adapter}}_{\mathsf{BN}\rightarrow\mathsf{PN}} = \mathcal{V}^{\text{adapter,info}}_{\mathsf{BN}\rightarrow\mathsf{PN}} \cap \mathcal{V}^{\text{adapter,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}} \cap \mathcal{V}^{\text{adapter,debug}}_{\mathsf{BN}\rightarrow\mathsf{PN}}. \tag{4.20}$$

$\mathcal{V}^{\text{adapter,info}}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ are the actual values communicated from the *Business Node* to the *Physical Node*; $\mathcal{V}^{\text{adapter,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ are the conditions triggering an EVENT; $\mathcal{V}^{\text{adapter,debug}}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ are information used for debugging purposes during the modeling process, they do not affect the state of the model.

We introduce the following conventions to form the relations that are included into $\mathbf{f_c}$, cf. Eqn. 4.17a, and thus trigger events:

- the EVENT CONDITIONS are formed from the elements of $\mathcal{V}^{\text{adapter,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ and a subset of $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}}$, precisely:

$$\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}[i] > \mathbf{v}^{\text{adapter,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}}[i],\ i = 1\ldots|\mathcal{V}^{\text{adapter,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}}|, \tag{4.21}$$

- the first element of $\mathbf{v}_{\mathsf{BN}\leftarrow\mathsf{PN}}$ is the SIMULATION TIME t.

The chosen convention implies that all EVENT CONDITIONS are known by the *Business Node*.

The full implementation of the *Adapter* is given in Listing 4.9 below:

```
1  partial model SynchModel "handle exchange with Business Node (Adapter)"
2  import TY = MOCES.TYPES;
3  import K = MOCES.CONSTANTS;
4
5  parameter Integer n_d_in = 1 "# of double variables communicated to BN";
6  Real d_in[n_d_in](start = fill(0, n_d_in)) "variables communicated to BN";
7  parameter Integer n_d_out = 0 "# of double variables communicated by BN";
8  Real d_out[n_d_out](start = fill(0, n_d_out)) "variables comm.ted by BN";
9
10 parameter Integer n_d_c = 1 "# of conditions given by BN";
11 Real d_c[n_d_c](start = fill(0, n_d_c)) "conditions given by BN";
12 Real d_c_s[n_d_c](start = fill(1, n_d_c)) "sign for cond. given by BN";
13
14 Integer id "unique identifier of ment";
15
16 TY.ErrorCode error "error code returned by BN";
17 TY.ErrorString error_message "error message returned by BN";
18
19 equation
20   when ({d_in[k] > pre(d_c[k]) * pre(d_c_s[k]) for k in 1:n_d_c}) then
21          (d_out, d_c, d_c_s, error, error_message) =
22            adapter(id,
23                    n_d_in,
24                    n_d_out,
25                    n_d_c,
26                    d_in,
27                    K.kLengthErrorMessage);
28          assert(error > 0, error_message);
29   end when;
30 end SynchModel;
```

Listing 4.9: Source code of the *Adapter* implementing the described synchronization mechanism.

The crucial points are line 20, which represents Equation 4.21, and lines 21 through 27, which reflect Equation 4.19.

4.4.7.2 *Formal Description of Meter*

The *Meter* as a special type of *Adapter* is represented by the following function:

$$\mathbf{v}^{\text{meter}}_{\mathsf{BN}\to\mathsf{PN}} := f^{\text{meter}}(\mathbf{v}^{\text{meter}}_{\mathsf{BN}\leftarrow\mathsf{PN}}). \tag{4.22}$$

If we divide $\mathcal{V}^{\text{meter}}_{\mathsf{BN}\to\mathsf{PN}}$ in the same way as $\mathcal{V}^{\text{adapter}}_{\mathsf{BN}\to\mathsf{PN}}$:

$$\mathcal{V}^{\text{meter}}_{\mathsf{BN}\to\mathsf{PN}} = \mathcal{V}^{\text{meter,info}}_{\mathsf{BN}\to\mathsf{PN}} \cap \mathcal{V}^{\text{meter,cond}}_{\mathsf{BN}\to\mathsf{PN}} \cap \mathcal{V}^{\text{meter,debug}}_{\mathsf{BN}\to\mathsf{PN}}, \tag{4.23}$$

the following applies to the subsets:

$$\mathcal{V}^{\text{meter,info}}_{\text{BN}\rightarrow\text{PN}} = \emptyset, \tag{4.24a}$$

$$|\mathcal{V}^{\text{meter,cond}}_{\text{BN}\rightarrow\text{PN}}| = 0 \vee 1. \tag{4.24b}$$

Contrary to the general *Adapter*, the *Meter* is designed as a PURE FUNCTION that does not *not* trigger any action by the *Business Node* but may modify the *local* state of the connected *Business Node*; the INTERNAL STATE is not affected by this type of *Adapter*.

As it is designed as PURE FUNCTION the call of the meter function, Eqn. 4.22, is not necessarily restricted to STATE EVENTS or TIME EVENTS. For performance reasons and to keep the implementation simple, however, it was decided to restrict the call to time events. Such time events are either triggered by one specific condition that is dynamically given by the *Meter* ($|\mathcal{V}^{\text{meter,cond}}_{\text{BN}\rightarrow\text{PN}}| = 1$) or at predefined points in time with constant step width ($|\mathcal{V}^{\text{meter,cond}}_{\text{BN}\rightarrow\text{PN}}| = 0$).

The call of the function representing the *Meter* can thus be included into different code structures. If using a constant step width, a code structure as given in the following code fragment is advantageous:

```
when sample(0, delta_t) then
  meter(time, value_to_meter);
end when;
```

Listing 4.10: Possible code fragment to call a *Meter*, option 1.

If the conditions of the time events are to be kept dynamic, the following two patterns are possible:

```
when (time > pre(t_c)) then
  t_c = meter_alt1(time, value_to_meter);
end when;
```

Listing 4.11: Possible code fragment to call a *Meter*, option 2.

or

```
when (sample(0, pre(t_c)) then
  t_c = meter_alt2(time, value_to_meter) - time;
end when;
```

Listing 4.12: Possible code fragment to call a *Meter*, option 3.

Option 2 in Listing 4.11 and option 3 in Listing 4.12 are the same in content, but are typically processed differently by the SIMULATOR and show performance differences depending on the selected SOLVER. The event condition in option 2 is not processed as a TIME EVENT, but as a STATE EVENT. When using option 3, the condition is processed as a TIME

EVENT. Contrary to expectations, option 3 does not necessarily perform better than option 2; the performance depends at least on the solver used.[14] By default, MOCES uses the first of the two patterns (option 2) for dynamic conditions.

4.4.7.3 *Formal Description of Observer*

The *Observer* as a special type of *Adapter* shares several characteristics with the *Meter*. In analogy, we define it using the following function:

$$\mathbf{v}^{\text{observer}}_{\mathsf{BN}\rightarrow\mathsf{PN}} := f^{\text{observer}}(\mathbf{v}^{\text{observer}}_{\mathsf{BN}\leftarrow\mathsf{PN}}). \tag{4.25}$$

Compared with the general *Adapter*, the *Observer* adds additional restrictions on the variables $\mathcal{V}^{\text{observer}}_{\mathsf{BN}\leftarrow\mathsf{PN}}$. It is restricted to be a subset of the time and constant parameters:[15]

$$\mathcal{V}^{\text{observer}}_{\mathsf{BN}\leftarrow\mathsf{PN}} \subseteq \{t \cup \mathcal{P}\}\,. \tag{4.26}$$

The restrictions on $\mathcal{V}^{\text{observer}}_{\mathsf{BN}\rightarrow\mathsf{PN}}$ are identical to those of the *Meter*:

$$|\mathcal{V}^{\text{observer,cond}}_{\mathsf{BN}\rightarrow\mathsf{PN}}| = 0 \vee 1. \tag{4.27}$$

As the *Meter*, the *Observer* is implemented as a PURE FUNCTION; in addition, it does not trigger any action of the *Business Node* nor does it change the *local* or *internal* state of the connected *Business Node*. The call of the function is therefore not strictly limited to events. For performance reasons, however, it is not recommended to call the *Observer* outside a `when` structure. The *Observer* is thus used within similar code structures as those given in Listings 4.10, 4.11 and 4.12 for the *Meter*. Again, the option given in Listing 4.11 is preferred.

4.4.7.4 *Interaction with Solver and Simulator*

Figure 51 shows how the different types of *Adapters*, and therefore the synchronization process of MOCES, play together with the standard structure of an algorithm used by a SIMULATOR to perform a SIMULATION by utilizing a SOLVER. For completeness, the processes for initiating and deleting *Business Nodes* described in Section 4.4.1 are also given. As the figure shows, the MOCES synchronization process affects only that part of the solution algorithm that concerns the action of events –marked green in the figure. The part that handles the DAE –marked blue in the figure– is not affected by the designed mechanism.

---

14 A study with DYMOLA 2018 64bit showed that when using the DASSL solver, option 2 is significantly more performant, whereas when using LSODAR, option 3 is significantly more performant.

15 In the concrete implementation, the variables that are passed to the function are part of $\mathcal{M}^-$, but they do not change their value during a simulation run. With a certain amount of implementation effort, $\mathcal{M}^-$ could therefore be dispensed with. We are thus using the set $\mathcal{P}$ for the description.

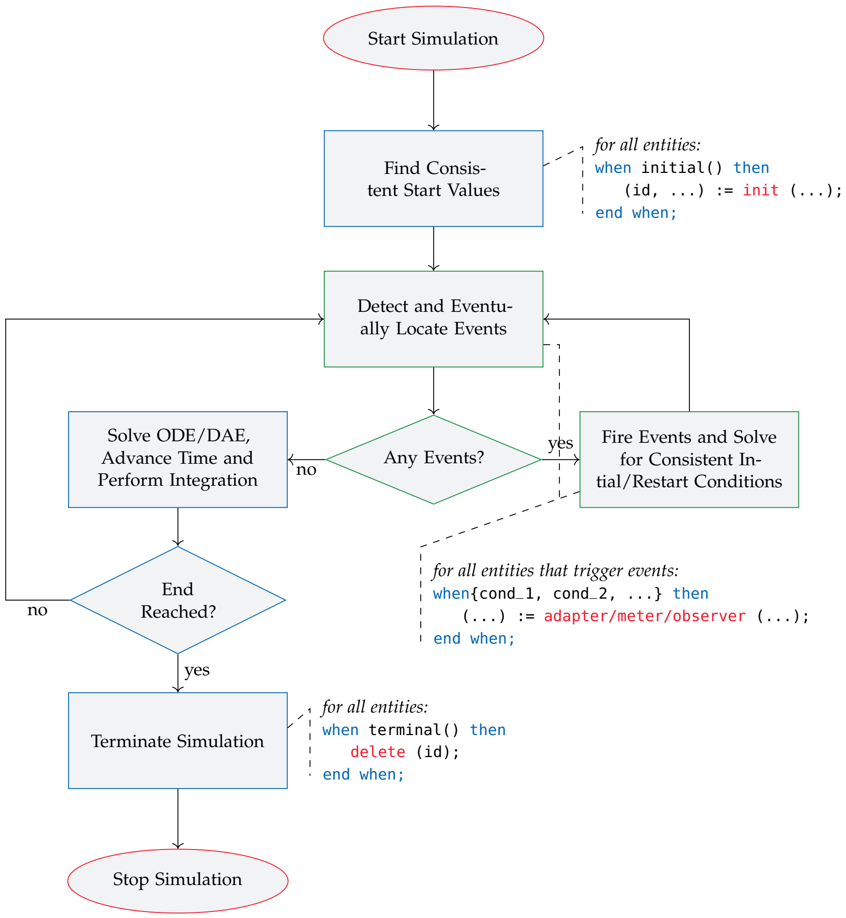


Figure 51: MOCES synchronization process and its interaction with the structure of the algorithm used to perform a simulation of a HDAE, extended from Liu [121] and Lundvall et al. [157].

4.4.8 *Business Process Layer*

As already outlined, MOCES uses an agent-based approach to model the interaction of *Business Nodes* within the BUSINESS PROCESS LAYER. This layer represents the BUSINESS ACTOR perspective on the POWER SYSTEM described in Section 2.2.3.2.

This part of MOCES is therefore designed to allow an explicit modeling of the core processes of the BMP, presented in Section 2.2.8, and the involved ROLES performed by ACTORS, described in Section 2.2.7. In addition, the agent-based modeling allows for a rather straightforward description of the behavior of individual *Business Nodes*. As shown in the example given in Section 4.3, it would theoretically be possible to implement this part of MOCES in MODELICA as well. However, the limitations of the MODELICA modeling language do not allow an efficient implementation. A serious problem, for example, is MODELICA's limitation that variables cannot be created dynamically at runtime or simulation time. This possibility was therefore rejected.

As described in Section 3.4, design decisions must be made when developing agent-based models. In the case of MOCES, these decisions are derived relatively directly from the design of the adapter and the characteristics of MODELICA. We summarize them as follows:

TIME HANDLING Derived from MODELICA, a continuous model of time is used. Actions performed by the *Business Nodes* can therefore be performed at any point of the SIMULATION TIME that thus proceeds from EVENT to EVENT in chunks variable in duration; chunks with zero duration are also possible. MOCES therefore uses a DYNAMIC SCHEDULING approach.

Handling the SIMULATION TIME, in particular controlling the simulation step size, is the sole task of the SIMULATOR used by the MODELICA IDE and cannot be directly influenced by the agent-based implementation of the BUSINESS PROCESS LAYER. Only an indirect influence via conditions for time events is possible. The actions in the BUSINESS PROCESS LAYER have no duration. This characteristic is visualized in Figure 49. All actions and interactions between AGENTS take place at events triggered by the *Adapter*.

GENERATION AND HANDLING OF EVENTS Events are fired and handled by the SIMULATOR based on the EVENT CONDITION specified by the AGENT, see Figure 51. In this way, we have found an elegant solution to the "dirty secret" (cf. page 121) of agent-based modeling and simulation. The design of the adapter results in the fact that agents can only define EVENT CONDITIONS for themselves, but not for other agents. It is thus not possible for AGENT a to enforce an action of AGENT b.

ACTIONS AGENTS are designed to have two generic actions:

- An action that is triggered by an EVENT, more precisely by the call of the `adapter` function. Only this action may modify the EVENT CONDITIONS.
- An action triggered upon the arrival of a *Message*. This generic action may modify the INTERNAL STATE of the AGENT.

None of these actions affect the environment or the INTERNAL STATE of another agent. No action is allowed to *remove* information from the LOCAL STATE of another agent. The modification of the LOCAL STATE of another agent is restricted to *adding* information. An AGENT is therefore not capable to *remove* information. This design decision is realized by the fact that agents can influence the LOCAL STATE of another agent only by sending messages which are processed by the receiving AGENT.

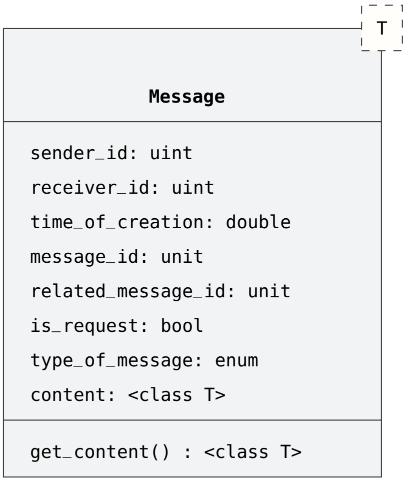


Figure 52: Class diagram of the message enabling communication between *Business Nodes*.

The adapters of the type *Meter* and *Observer*, specifically the call of the related MODELICA functions, do not trigger any action of the agent in the actual sense. The adapter of type *Meter* only adds information about the local state, the *Observer* retrieves information about the agent's LOCAL STATE without modifying it or the INTERNAL STATE.

LIFETIME OF AGENT An agent is created at the beginning of a simulation run and dies at its end.

COMMUNICATION The *Business Nodes* communicate via messages that can have any content. In principle, all *Business Nodes* can communicate with each other. Only the identifier of the agent must be known to correctly address the message. As this identifier is generated dynamically at the beginning of the simulation, cf. Figure 45, it is typically requested from the MENT `System`, cf. Figure 46, based on a known unique name.

The basic structure of the message is represented by the class diagram given in Figure 52, which contains the typical properties of a message such as addressee, sender and payload. A special feature is the `is_request` attribute of the message, which indicates that the message payload is a request for information that is typically answered directly by the recipient, as the processing of the message by the recipient does not trigger any actions that modify the INTERNAL STATE of the *Business Node*.

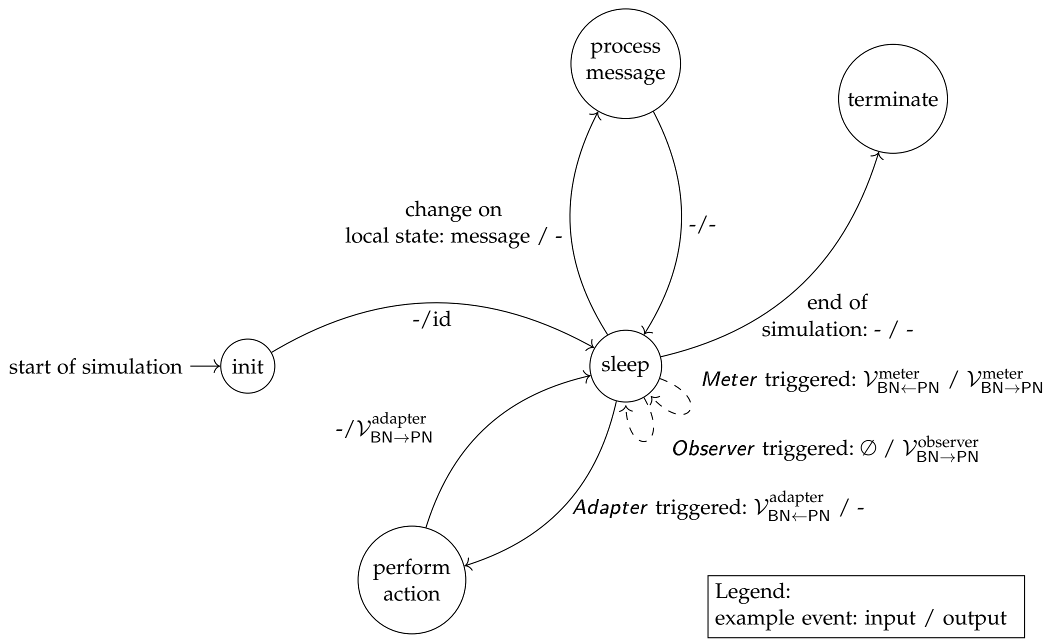


Figure 53: Simplified state machine of an agent implementing a *Business Node*.

4.4.9 *Business Node*

Based on the design decisions presented in the previous chapter, the basic behavior of all *Business Nodes*, and thus AGENTS, can be described using the state machine in Figure 53. Almost all events that trigger state transitions have their origin in the connected *Physical Node*, specifically in the event-based function calls of the different *Adapters*.

For clarification, reference is made in the figure to the corresponding set of variables, such as $\mathcal{V}^{\text{adapter}}_{\text{BN}\rightarrow\text{PN}}$, that serve as inputs and outputs of the different *Adapters*. If no condition is given for the transition, as for example with the transition between 'process message' and 'sleep', the transition fires immediately after processing the previous state.

An exception regarding the source of the events is the transition into the 'process message' state, which is triggered by the arrival of a message. This special case has been introduced so that agents can react immediately to messages and do not have to wait for the next event triggered by the connected *Physical Node*, for example to allow *Business Nodes* which mainly provide information such as weather forecasts to respond immediately. In general, the actions that a *Business Node* performs upon arrival of a message are the sole responsibility of the node itself, and they are subject only to a single restriction: an interaction with the connected *Physical Node* is not possible. Actions on the *Business Node*'s own LOCAL STATE are however possible.

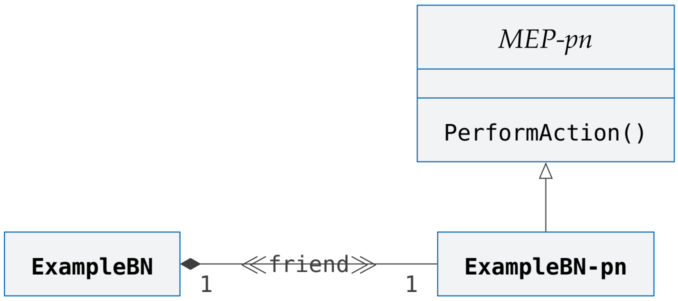


Figure 54: Pattern used to implement the behavior of a *Business Node*.

HANDLING OF MESSAGES The processing of incoming messages is the responsibility of the method `PutIntoInbox`, as shown in the class diagram in Figure 44, which is called by the sender of the message. This method invokes a method called `HandleRequestIfPossible` trying to handle the message. If it cannot be handled immediately, the message is added to the message deque and a corresponding answer is returned to sender of the message. Messages that cannot be processed immediately are typically processed the next time the agent is called by the adapter.

The method handling the request is implemented not only in the concrete class implementing the agent, but also in its base classes. These implementations of the base classes are called at the beginning of the method. In this way, the processing of requests to which all agents respond in the same way can be implemented in a modular and straightforward manner. An example is the processing of a message that represents an invoice. It is the implemented in the abstract base class `BN-MEP` and is therefore the same for all agents unless an agent overwrites this functionality.

MODELING THE BEHAVIOR (RULES) In principle, the behavior of the *Business Nodes* can be freely modeled in the MOCES framework. Free here means that, in contrast to modeling the physical part, modeling takes place directly in C/C++, without using a special description language, in order not to restrict the modeling options.

However, MOCES recommends a separation between the implementation of possible actions, such as placing a bid on the market, and the behavior that in turn makes use of these actions. MOCES also suggests to use Petri nets to describe the behavior. These recommendations are based on the following two considerations:

- Using this approach, the individual instances of *Business Nodes* of the same type can easily be linked to different, individual behavior patterns described by different Petri nets. For example, a *Business Node* that represents a battery storage can be linked to different behavior patterns that pursue different strategies for operating the storage system.

- A behavior that is conditioned on the adoption of a standardized *Role* must only be described once, independently of the concrete type of *Business Node*, and can be re-used by multiple types of *Business Nodes* assuming this *Role*. A typical example is the behavior of a BRP in the context of the BMP, cf. Figure 21.

These considerations result in the class structure shown in Figure 54, which is used by all base models given in Figure 48. The Petri nets are implemented in a child class of the abstract base class *`MEP-pn`* that defines the interface with its method `PerformAction()`. Due to the chosen «`friend`» relation, the class describing the behavior, here `ExampleBN-pn`, is allowed to use the methods of the class that models the *Business Node*, `ExampleBN`. The composition creates a close connection between both classes, which is additionally limited to a 1-to-1 relation to keep the implementation simple. In the future, a looser 1-to-n coupling might be desired here. Ideally, the coupling would even be dynamic such that it can change during simulation.

Figure 55 shows an example of the behavior of a PV system. It is a slightly simplified representation using Petri nets which served as a stating point for the implementation.[16] The three Petri nets address different tasks: the communication with the ELECTRICITY MARKET (day-ahead market) and the creation of orders for this market; the communication with a forecasting service; and, the communication with the SOP. The shown behavior that the state transitions are linked to the arrival of messages or the exceeding of a point in time is a typical pattern. In addition, state transitions can be linked to the exceeding of individual variables of limit values if the variable is accessible by the agent as it is part of $\mathcal{V}_{\mathsf{BN}\leftarrow\mathsf{PN}}$, cf. Section 4.4.7.1.

Note however that these checks can only take place if the *Business Node* is temporarily transferred from its 'Sleep' state to the 'Perform Action' state by the *Adapter* calling the synchronization function, cf. Figure 43. Typically, the variables that are part of the conditions in the Petri net are therefore included in the event conditions of the *Adapter*, cf. Equation 4.21; the variable time is included by default. The recognition of the state transition is thus transferred to the SIMULATOR of the HDAE, cf. Figure 51.

As the Petri nets in Figure 55 show, a variable, in this case the time t, is typically linked by several Petri nets with different conditions. In this example, there are a maximum of three simultaneously valid conditions:

$$t > t_{c,1}\ ,\ t > t_{c,2}\ ,\ t > t_{c,3}. \tag{4.28}$$

16 So far, MOCES does not provide the functionality to model the behavior using (graphical) tools and to derive the implementation automatically.

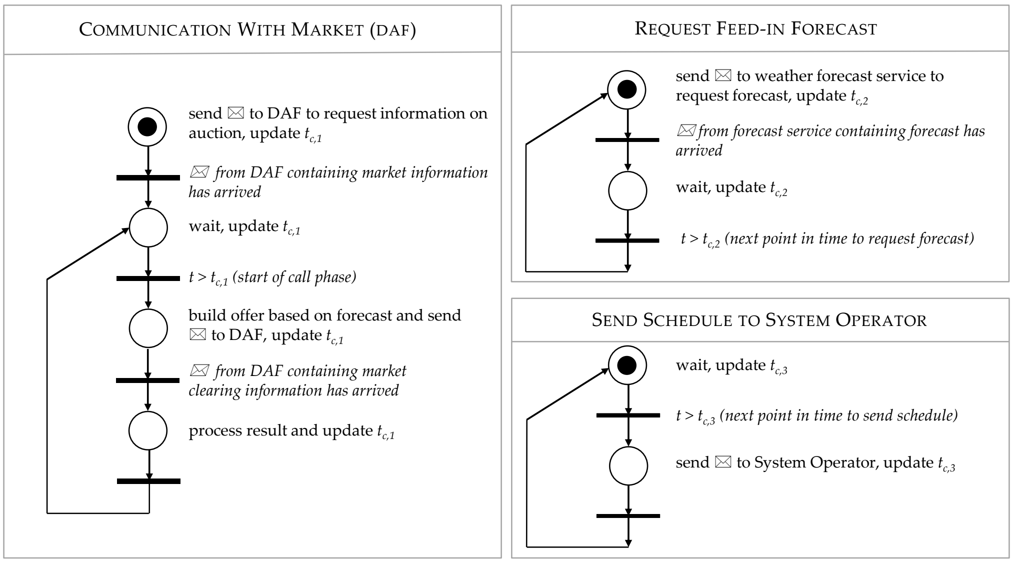


Figure 55: Exemplary behavior of the *Business Node* of a MENT describing a PV plant described by a set of Petri nets. The acronym DAF stands for 'day ahead flex' and is derived from the chosen model name for a day market in MOCES.

To reduce the number of conditions that are integrated into the *Adapter*, only one condition is integrated into the *Adapter* and the minimum of $t_{c,1}, t_{c,2}, t_{c,3}$ is transferred as $\mathbf{v}^{\text{adapter,cond}}_{\text{BN}\to\text{PN}}[1]$, cf. Equation 4.21, to the *Physical Node*.

In summary, the processing of the Petri nets within the `PerformAction` method can be best described with the following pseudo code:

```
void PerformAction(ExampleBN* ptr) {
  // ptr: pointer to Business Node
  // set all condition values to infinity
  for v_i in ptr-> V^{adapter,cond}_{BN→PN} {
    v_i = DBL_MAX;
  }
  // iterate over all petri nets
  for pn in PetriNets {
    // determine state of petri net
    pn.GetNextState();
    // execute actions linked with this state and get values for cond.
    v_i_temp = pn.ExecuteActions();
    for v_i in ptr-> V^{adapter,cond}_{BN→PN} {
      if v_i is affected by pn {
        v_i = min(v_i, v_i_temp);
} } } }
```

Listing 4.13: Processing of Petri nets describing the behavior of a *Business Node*.

The Petri nets are processed sequentially. However, the individual Petri nets only reset the conditions of the *Adapter* if their condition is reached earlier than the currently valid condition.

Some of the transitions in Figure 55 come without a condition, which means that they can fire immediately. To achieve this, t_c $(\mathbf{v}_{\text{BN}\rightarrow\text{PN}}^{\text{adapter,cond}}[1])$ is set to an infinitesimally larger value than the current simulation time.

4.4.9.1 *Running Example*

We will continue the example of the MENT `BatteryStorage`. The physically modeled part has been described in Section 4.4.6.1 (p. 160). We will now describe the modeling of the associated *Business Node* of the MENT in more detail.

The physical part of the MENT is based on the Equations 4.13 and 4.14. It follows a predefined schedule as long as this is not technically impossible, which would for example be the case if the schedule requires unloading although the storage system is already empty. In which way and with which goal this schedule is determined, depends on the individual goal of the described MENT and the possible courses of action. In the following, we assume an entity that pursues the goal of generating the highest possible yield through the sale and purchase of energy quantities by utilizing the storage capacity. We assume that the market activities of the *Business Nodes* are limited to the participation in the day-ahead market, which has the characteristics as described in Paragraph 2.2.9.1 (p. 62). The required process steps have already been shown in Figure 21, and are roughly as represented by the Petri nets in Figure 55.

In principle, a wide variety of procedures are conceivable for determining a schedule, such as simple heuristics or using an adapted form of the reinforcement learning approach described in Section 3.2.4.1. In the present modeling, however, the schedule is determined by the *Business Node* using a linear program similar to the formulation of a battery storage used by hybrid models, cf. Section 3.2.3.2. With anticipation of the time span given in Figure 56, the goal of the *Business Node* for the 'next day' is the solution to the following optimization problem:

$$\max \sum_{t=24}^{48} \Delta t \cdot \left(\underbrace{p_{\text{s}}^{\text{out}}[t] \cdot c[t]}_{\text{sold energy}} - \underbrace{p_{\text{s}}^{\text{in}}[t] \cdot c[t]}_{\text{bought energy}} \right) \tag{4.29}$$

subject to:

$$E[t+1] = E[t] + \frac{3600.00\,\text{s}}{1.00\,\text{h}} \left(p_\text{s}^\text{in}[t] \cdot \eta_\text{charge} - p_\text{s}^\text{out}[t] \cdot \eta_\text{discharge} \right), \tag{4.30a}$$

$$\forall t \in \{24 \dots 48\}$$

$$soc[t] = E[t] / E_\text{capacity}, \forall t \in \{24 \dots 48\} \tag{4.30b}$$

$$soc[24] = soc_\text{start} \tag{4.30c}$$

$$soc[48] = 0.5 \tag{4.30d}$$

$$soc[t] \geq 0, \forall t \in \{24 \dots 48\} \tag{4.30e}$$

$$soc[t] \leq 1, \forall t \in \{24 \dots 48\} \tag{4.30f}$$

with:

$p_\text{s}^\text{out,in}[t]$	planned schedule ($p_s = p_s^\text{in} - p_s^\text{out}$),
$c[t]$	market price at time period t,
$\eta_\text{discharge}$	discharge efficiency of storage system,
η_charge	charge efficiency of storage system,
$E[t]$	energy stored in system,
$soc[t]$	state of charge of storage system,
soc_start	expected state of charge of storage system at start of considered time span.

This at first sight trivial task has its difficulty in the fact that the information required to find an optimal schedule is only partially known at the time the decision is made. In the given, realistic, example in Figure 56, the GCT of the day-ahead market is at 12:00, which means that the schedule must be determined before that time, e.g. at 10:00, which is described in the figure as the current time. At this point, however, neither the storage level at time t=24 (cf. soc_start in Eqn. 4.30) nor the market price is known. When participating in the market, it should also be noted that the *Business Node* can only market the amount of energy for a specific time slice that it has purchased before that time and is thus available in the storage system. To guarantee this, it must place bids at the market that ensure this, even if the price assumed for the optimization problem is lower than the price resulting from bringing supply and demand together.

In summary, the *Business Node* of the MENT in this example must essentially implement four functionalities:

- predicting the market price,
- predicting the storage state in the future,
- solving the optimization problem on the basis of the predictions,
- derivation of bids and offers for the day-ahead market.

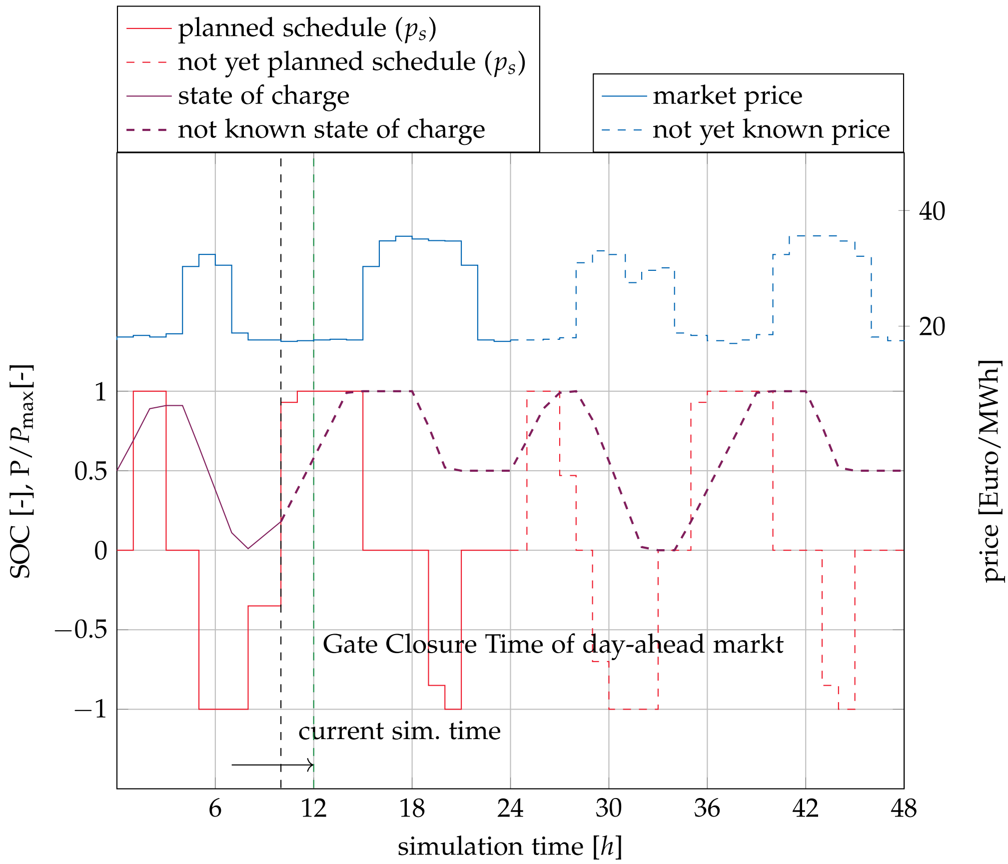


Figure 56: Visualization of the challenge of the storage system to find an optimal schedule. Dashed lines indicate, that the represented information is not known at the current simulation time that is given by the black dashed vertical line. The green vertical line indicates the GCT.

They are implemented as methods of the class representing the *Business Node* part of the MENT `BatteryStorage` or as requests to another AGENTS to provide an appropriate energy service, as described in the following.

PREDICTION OF MARKET PRICE Predicting the market price is a typical function many different types of *Business Nodes* make use of. It is therefore provided as a service by a dedicated *Business Node* such as the `SimpleMarketForecast`, cf. Figure 48. This specific *Business Node* provides forecasts with a standardized interface defined by the abstract parent class; the interface accepts a message with a defined structure and returns a response with a message containing a time-series in a defined format.[17] The ment BatteryStorage therefore simply queries this prediction by sending a message and processes the response.

prediction of state of charge The prediction of the state of charge is also implemented as a class method. This method performs a simulation based on the known schedule and uses a simplified model of the system in the form of an ode which is similar to the model described by the Equations 4.13 and 4.14 (p. 160). To be able to parameterize this model, the main parameters, such as the storage capacity, are part of the parameter structure init_SBS, cf. Listing 4.8, which is passed from the *Physical Node* to the *Business Node* in the initialization process at the beginning of the simulation. In order to be able to simulate the model correctly, the storage level is included into $\mathcal{V}_{\mathsf{BN} \leftarrow \mathsf{PN}}^{\text{adapter}}$ and therefore known by the *Business Node*.

The implementation uses the boost library odeint[18], which can simulate models based on odes using solvers provided by the library. The library expects a description of the model in a form similar to the output of the step *code generation* of the modelica transformation process, cf. Figure 28 (p. 106). The process steps of the transformation preceding this step must therefore be performed by the modeler, which is rather straightforward for the considered system. The resulting description is given in the following listing:

```
void BatteryStorage::sys(const double &x, double &dxdt, simtime t) {
  double P_S = ov_ts_schedule.get_value(t);
  if (P_S > 0) {
    dxdt = P_S * init_SBS_.eta_charge;
  } else {
    dxdt = P_S / init_SBS_.eta_discharge;
  }
};
```

Listing 4.14: Representation of the ode describing the battery storage system. The parameter structure of the method sys follows the specifications of the odeint boost library.

17 The implementation of this service will not be presented further in this thesis. It is based on a nonlinear autoregressive network with exogenous inputs (narx) model that is implemented as an ann. The input parameters are weather forecasts, which in turn are provided by a service, as well as the historical prices of the market. The ann was developed by using the matlab Neural Network toolbox and can be pre-trained with historical prices. This pre-trained model is exported as C-code from Matlab and integrated into the *Business Node* SimpleMarketForecast. Having the *Business node* re-train the network during the simulation time to achieve an improved forecast is left for future work.

18 https://www.boost.org/doc/libs/1_68_0/libs/numeric/odeint/doc/html/index.html (last accessed: 3 June 2021).

Alternative approaches would have been to reuse the MODELICA model of the *Physical Node* or to implement the feature to model with MODELICA as modeling language. However, these were rejected due to the implementation overhead and expected performance losses.

SOLVING THE OPTIMIZATION PROBLEM AND GENERATING BIDS AND OFFERS Both steps are performed simultaneously within one method. This is possible as the *Business Node* pursues a strategy according to which the optimal solution to the optimization problem for the schedule of the storage system is transferred to *sell-* or *buy-at-any-price* orders.

The method uses the LP-SOLVE library, cf. Berkelaar et al. [158], to automatically formulate and solve the optimization problem described in Equation 4.29. The parameters of the storage system passed to the *Business Node* (`init_SBS`) are again used to set up the optimization problem.

The boundary conditions used are the expected market prices determined in the previous steps and the simulated storage level at the beginning of the period under consideration.

4.4.10 *Limitations of the Current Implementation*

The current implementation still has room for improvement in terms of its performance. Above all, it does not make use of multithreading in critical places, as MOCES as well as the standard SOLVERS currently use only one thread. For the simulation of the agents described in C++, a multithreading functionality could be implemented relatively easily. The basic idea would be to build a pool of threads to which the individual agents are dynamically or statically assigned during the simulation. In this way, the actions of the individual agents could be processed simultaneously in terms of the WALL TIME.

However, this approach does not solve the challenge that the actions of all agents must be completed before the SOLVER can continue the simulation of the DAE by increasing the SIMULATION TIME. This challenge is based in the design of the *Adapter* and becomes apparent when considering Figure 49. Solving this challenge becomes particularly relevant for AGENTS that run computationally expensive steps to perform an action or to make a decision, such as solving an optimization problem or running a simulation as described in the example in Section 4.4.9.1.

In both cases, however, it is possible to take advantage of the fact that running computationally expensive tasks may take minutes to hours in terms of SIMULATION TIME without blocking the entire simulation, since the result is not required immediately. In relation to the example in Section 4.4.9.1, the agent could start solving the optimization problem at around 8:00 and use the result at 10:00 to place its offers on the market.

The idea that follows from this would be that the individual agents could outsource computationally intensive steps in a non-blocking task and assign them an end time. Only if the simulation time exceeds this point in time and the calculation is not completed, the increase of the simulation time is blocked until the task is completed. Whether a central pool should be set up for this purpose, to which the individual agents can send their computationally expensive tasks, or whether the agents should implement this functionality individually, needs to be investigated more concretely.

A similar pattern could also be used by agents who mainly provide services such as the market forecasting service mentioned in the example in Section 4.4.9.1 as well. These typically also have computationally intensive steps such as determining the predictions. Such steps could also be easily outsourced to non-blocking tasks that run parallel to the simulation.

4.4.11 *Visualization*

MODELICA offers the possibility to extend models in such a way that a 3d visualization is possible by using appropriate tools that render the information added to the models. These tools are usually already integrated in the MODELICA IDE. An example is the MULTIBODY library of the MSL [159].

Using the basic models for bodies and surfaces described in this library, 3d visualizations of models can also be created in MOCES. Figure 57 shows an example in which a model of the German energy system created in MOCES is visualized.

4.5 SCENARIO MODELING PROCESS

In this section we would like to describe how the MOCES framework presented so far can be used to model a specific scenario in order to answer a specific question. An example for such a scenario could be the *German Power System* with the following question to investigate: "What is the effect of a local optimizer to the transmission grid?". Both will be tackled in Chapter 5.

This modeling step, which is at the beginning of the MS chain, cf. Figure 3, is a time-consuming and challenging process that furthermore depends on the specific scenario to model and addressed question. Nevertheless, MOCES offers a somewhat general approach to it. We will provide a high-level description of the proposed procedure.

The basic idea of the procedure is to break down the entire scenario into its individual sub-aspects. In the scenario *German Power System*, these include i.a. the behavior of all wind power systems, PV-systems,

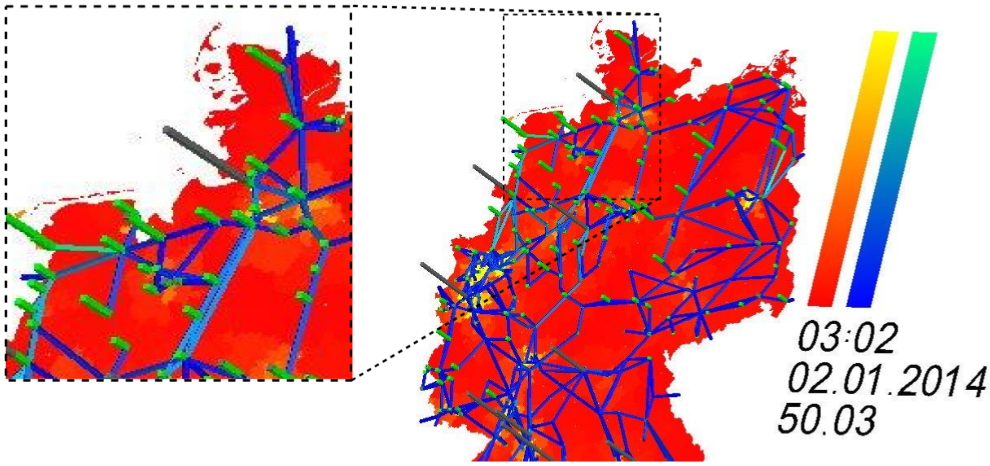


Figure 57: Visualization of simulation results in DYMOLA. Besides the line usage (right color bar) and the load density of the regions (left color bar), it shows the feed-in at the respective nodes with stacks perpendicular to the plane. The color indicates the source (green: wind, grey: conventional, yellow: PV). The numbers on the right show the time, date and grid frequency.

conventional power plants and also the transmission grid. Ideally, models can be created for these partial aspects that can be simulated independently of the complete scenario and whose quality can be determined easily. The models for the individual aspects are then combined to form an overall model. The entire modeling process therefore consists of two main steps:

1. Building several *Optimized Scenario Aspect Models*, cf. Figure 58.
2. Merging different *Optimized Scenario Aspect Models* into a *Scenario Model*, cf. Figure 59.

The proposed procedure can best be explained with an example. For this purpose, we will outline the modeling of the aspect *wind power plants in Germany* and its inclusion into a holistic model of the POWER SYSTEM.

PHASE 1 As shown by the flow diagram in Figure 58, the creation of an *Aspect* model should begin with the modification and adaption of an existing MOCES *Base Model*, cf. Figure 48, into a *Scenario Base Model* if an appropriate base model exists.

In the concrete example, a *Scenario Base Model* represents the aggregated behavior of all wind turbines in a region that has about the size of a NUTS-3 area. This model is derived from a basic model for wind turbines available in MOCES, which uses wind power curves. These type-specific power curves are provided by the manufacturers and reflect the power as a function of the wind speed. More precisely, the *Scenario Base*

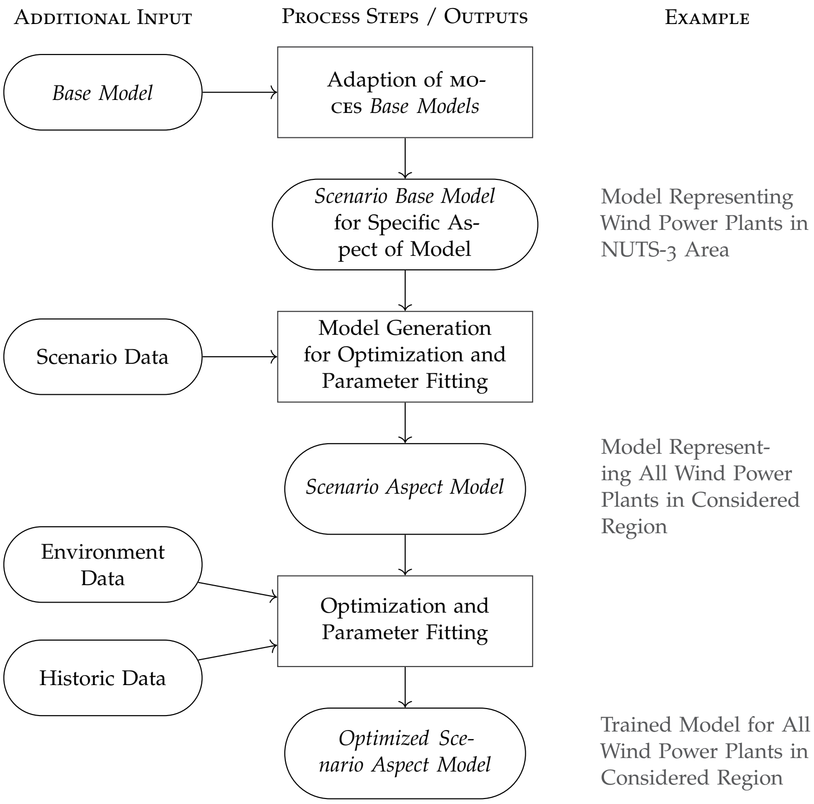


Figure 58: First phase of the modeling and simulation process in moces.

Model uses two instances of the moces base model for wind turbines; one is parameterized with a power curve that is typical for wind turbines used in less windy regions, the other one with a curve that is typical for wind turbines in strong wind regions. The respective models can be weighted with a parameter.

To describe the behavior of all wind turbines in the considered region (Germany), this basic model can now be instantiated as often as required, for example several hundred times, and each instance can be parameterized with known scenario data. In our example, scenario data mainly refers to the capacity of the installed power per region. This step is typically supported by a model generator; in the simplest case, this is a script generating the model code in Modelica on the basis of given scenario data.

In the next step, which is optional, the quality of the model is therefore improved by fitting parameters based on known historical data for model input and expected model output. In the example, these are historical weather data and electricity production from wind turbines. Among other things, the weighting factor of the *Scenario Base Model* is fitted.

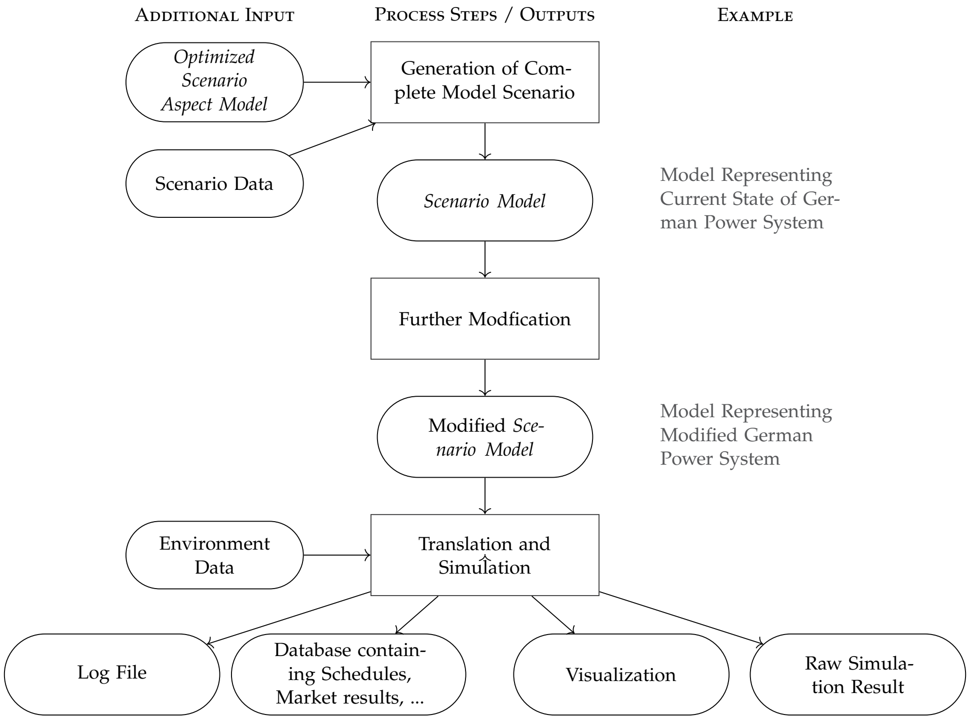


Figure 59: Second phase of the modeling and simulation process in MOCES.

In the case of using DYMOLA as MODELICA IDE, the model fitting process is supported by the optimization library, see Pfeiffer [160] for an introduction. Other MODELICA IDES provide similar toolboxes.

Not every aspect of a scenario can be modeled in this way, as the bottom-up approach described is not always the most appropriate. The modeling of the aspect *grid* that will be described in Section 5.4.6 is an example of this; guidance on why deviations from the preferred approach may be necessary and appropriate will be provided there.

PHASE 2 Figure 59 shows the second phase of the modeling and simulation process. First, the different, potentially *Optimized*, *Scenario Aspect Models* are united. Due to the component-oriented characteristic of MODELICA, this only requires the instantiation and linking of the individual models via connectors. In the example scenario for the *German Power System*, those are the models for wind power plants as just described, PV, grid, etc. The result is a validated model of the current power system, which will be modified in the next step to answer specific questions. Depending on the questions, the modification can be limited to parameters of the different models, e.g. when investigating the influence of the expansion of wind power plants on electricity prices. In this case, only the parameters representing the installed capacity would be scaled. Typically, however, the structure of the model is also affected during this step, for example, when adding components that did not exist before in the first phase, e.g. battery storage systems. The behavior of the *Business Nodes* can also be modified in this step.

Finally, the translation and simulation step is on behalf of the MODELICA TOOLCHAIN (MTC), which also enables a visualization of the simulation results, cf. Figure 57 for an example. Besides the simulation results generated by the MTC, MOCES generates further data, such as market data or schedules. This data is typically stored in databases and can be used for further investigations of the simulation results.

4.6 SUMMARY

Chapter 4 described the modeling framework MOCES in detail, with its core idea to introduce the MENT as the main building block, which consists of an HDAE model and an agent-based model, called the *Physical Node* and the *Business Node* respectively. While the HDAE can be used to model the physical behavior of a component, the agent-based part is dedicated to represent the interactions with other entities in the context of energy management processes such as the BMP as well as individual complex behaviors such as placing bids and offers on the energy market to pursue a certain goal, e.g. operating a power plant in a most profitable manner. In this way, the two perspectives on the power system –the SYSTEM ACTOR and the BUSINESS ACTOR perspective– described in Chapter 2 can be considered elegantly in one holistic model. This applies in particular to the typical characteristic that the *Business Node* part of a MENT influences the behavior of its *Physical Node* part by set points.

As both parts of the MENT have clearly defined interfaces, the creation of a complex model, e.g. a model of the German power system, is basically nothing more than parameterizing, and potentially optimizing, the existing *Base Models* that have been developed as part of this work, instantiating them hundreds of times and linking them with each other.

In order to use MOCES not only for modeling, but also for efficient SIMULATION of the developed models using existing MODELICA IDES, a novel synchronization mechanism was developed, which allows for a bidirectional information exchange between the two parts of a MENT, complies with the MLS and enables fast simulation.

The chapter has also provided implementation details, using the example of a storage system to demonstrate how it can be modeled within the framework in a simple way. A detailed example of MODELING AND SIMULATION (MS) with MOCES will follow in the subsequent chapter.

5 A MOCES SIMULATION

In this chapter we will present a simulation model developed in MOCES and its creation based on the proposed modeling process in detail. The model is based on the ELMOD-DE model for the German POWER SYSTEM developed by the DIW. In the context of this chapter, the model, which is originally an OPTIMIZATION MODEL, is not only adapted into a SIMULATION MODEL, but also extended to include future aspects and used to investigate the ideas for a more decentralized POWER SYSTEM discussed in Section 2.3 (p. 64).

The chapter begins with the presentation of the scenarios that will be investigated in detail (Section 5.1) and a brief description of the ELMOD-DE model (Section 5.2). The three following Sections 5.3 - 5.5 apply the modeling procedure described in Section 4.5, and provide an insight into the modeling depth of the developed model and restrictions due to a limited data basis.

Finally, Section 5.6 presents the results and describes the influence of decentralized actors on the energy mix, network utilization and stability as well as the individual profits of different actors.

5.1 INVESTIGATED SCENARIOS AND GOALS

In this chapter, we will investigate how well MOCES is suited to simulate a complex energy system like the one of Germany using several thousand MENTS interacting with each other. Starting from a baseline model based on the ELMOD-DE model, we would like to discuss the following two questions:

- How does the overall performance of the system change if individual elements strive for a more decentralized behavior with the goal to consume VRE-based energy as much as possible?
- What is the effect of erroneous weather forecasts?

To investigate those questions, we supplement the model with so-called LOCAL OPTIMIZERS (LOPS) attached to certain nodes of the simulation model. A LOCAL OPTIMIZER is a storage system that pursues the strategy to increase the self-consumption of renewable energy sources. In this context, a node represents a certain region of Germany; the model consists of more than 400 nodes.

The term *overall performance* in the question formulated above is deliberately kept vague here. With references to Section 1.5 and especially Figure 3, it should be noted that this chapter primarily describes the results of the simulation and only partially interprets them. We deliberately avoid making statements about the described German power system (cf. *Findings* in Figure 3) or even recommendations (cf. *Design Decision* in Figure 3) for its further development based on the simulation results. This must also not take place, since some of the models used have not been validated.

We will investigate the following four scenarios and provide insights and provisional findings:

`3Lop` 3 nodes are equipped with a LOCAL OPTIMIZER, no forecast error.

`3LopErr` 3 nodes are equipped with a LOCAL OPTIMIZER, forecast error on wind speed and irradiance.

`ManyLop` 10.00 % of all nodes are equipped with a LOCAL OPTIMIZER, no forecast error.

`ManyLopErr` 10.00 % of all nodes are equipped with a LOCAL OPTIMIZER, forecast error on wind speed and irradiance.

In the scenarios `ManyLop` and `ManyLopErr` an overall generation capacity of 7.00 GW and a storage capacity of 42.00 GW h is added by the LOCAL OPTIMIZER entities to the system.

5.2 ELMOD-DE

The main purpose of the ELMOD-DE model [161] is to determine cost-optimal electricity generation under the constraints of the electric grid, especially the maximum power flows of the transmission lines. The structure of the grid and the installed power plant capacity are assumed to be given. The model thus corresponds approximately to the operation models presented in Section 3.2.3.1. The model uses a formulation for the so-called *local marginal pricing* or *nodal pricing* schema and thus generates an individual price for energy for each node. Prices between nodes may differ due to transmission congestion.[1] These prices express what the purchase of an additional energy unit at a node would cost. For this purpose, the system-wide costs are determined that would be incurred in the cheapest case and allocated to the demand at the node.

ELMOD-DE uses a linear grid model and a linear cost-minimizing objective function; the overall optimization problem can therefore be solved efficiently by appropriate algorithms, such as the SIMPLEX METHOD. The model is implemented as a GENERAL ALGEBRAIC MODELING SYSTEM

1 Classically, line losses also lead to different prices, but these are neglected by ELMOD-DE.

(GAMS) model. It focuses on the transmission grid and totally neglects the underlying grid levels, cf. Figure 13. As the name of the model indicates, it models the German electricity system. It represents its grid structure and the connected power plants in the year 2012. The transmission network is modeled by means of 438 nodes, interconnected with 697 lines. The nodes represent the individual regions and all consumers and producers that are present in this region. A detailed description of the data processing performed to model the node-specific behavior is given in Egerer et al. [162]. In the following, we outline the modeling approach of ELMOD-DE:

- The power consumption per node is modeled by a time series with a temporal resolution of one hour. The basis of the time series for the individual nodes is the load profile of Germany. It is distributed to the individual nodes on the basis of the population of the region and its GROSS DOMESTIC PRODUCT (GDP).
- The feed-in of VRE-based power plants at each node is modeled by a time series with a resolution of one hour. They are based on the time series of the type of power plant (PV, wind onshore, wind offshore) determined by the transmission system operators for their respective grid areas, and then distributed to the individual nodes according to their installed capacity.
- The feed-in of all other renewable energy is considered to be largely constant over time. Only run-of-river power plants change their feed-in from month to month, thus representing the fluctuating river levels over the year. The known capacity of the renewables is scaled by an availability factor to adjust the feed-in to the actual yearly feed-in.
- Cross-border power flows to neighboring countries are modeled with predefined time series based on data provided by the TSO and ENTSO-E.

Thus, with respect to the optimization problem, the feeders and consumers mentioned represent fixed boundary conditions without feedback to the modeled system. This means that consumption is assumed to be price-inelastic and generation from renewable energies is always fed into the grid when it is available.

Conventional power plants and pumped storage systems are included into the optimization model in the following way:

- All conventional power plants are modeled individually and connected to the nearest node. The basis is a list of conventional power plants provided by the BNETZA. The behavior of the power plants is described by a strongly simplified formulation of Equation 3.1 (p. 90). The region of feasible production, cf. Equation 3.2, is only limited by the maximum capacity; restrictions regarding the load gradient or minimum load are completely ignored. The costs of the feed-in per energy quantity are determined by a fixed but power plant-specific efficiency rate and the costs for fuel, which is determined by the sum of the fuel costs, the location-dependent transport costs and the time-variable costs for the emission of CO_2.

- All pumped storage power plants are also modeled individually and connected to the nearest node. Just like conventional power plants, storage systems are also modeled in a simplified way. Their operation is limited only by the storage capacity and a maximum power that is assumed to be symmetrical for charging and and discharging. The cycle efficiency is assumed to be 75.00 % for all plants. In the boundary conditions of the optimization problem,

  the storage level is set to zero at the beginning and end of each calendar week. The authors justify this decision with the need to be able to compare the results of the respective weeks with each other. As a positive side-effect, the complete optimization model is broken down into many smaller problems (one for each week) that can be solved independently. However, any more optimal operating strategies that provide for storage across week boundaries cannot be found through these limitations.

Based on these properties, a model run of ELMOD-DE provides the following findings about the considered system for every hour of the considered time span:

- The levels of generation of each individually modeled power plant and pumped storage system.
- The utilization of the individual transport grid lines.
- The nodal electricity prices for every node.

These raw data serve as a basis to derive statements about the performance of the entire system, such as the hourly national generation and its composition according to energy sources.

The quality of the model is difficult to judge. The authors perform only one validation, comparing the model's aggregated annual power generation by type of power plant with the actual historic values for 2012. As expected, there is no deviation in the case of renewable producers, since their actual feed-in is also reflected in the model. In the case of conventional generators, there is a deviation for those types of power plants that have a relatively high marginal price and are thus no base-load power plants. The use of coal-fired power plants is clearly overestimated (≈ 30.00 %), while the use of gas-fired power plants is significantly underestimated (≈ 50.00 %). Possible reasons are discussed by the authors. The biggest model error is probably due to the disregard of the must-run conditions of gas-fired power plants if used not only for power supply but also for heat supply via a district heating network.

The ELMOD-DE model has served as a basis for several investigations of specific research questions. The authors of [161], e.g., mention [163] and [164] as examples. [163] discusses different investment strategies and their effect on the total cost of the system as well as on the emission of CO_2. For this purpose, projections for 2024 and 2034 are considered. [164] examines the effects that splitting the single German price zone into a northern and a southern price zone would have on the market. The focus is on the resulting market prices and the changing shares of the different types of conventional and renewable producers.

5.3 OVERVIEW OF ELMOD-DE EXTENSION

An overview of how the MOCES model adapts and extends the original ELMOD-DE model is given in the Tables 8 and 9. As shown there, the MOCES version adopts many of the structural parameters provided by ELMOD-DE and the approach to accumulate the load and the feed-in of renewables per node; the main difference is in the type of modeling.

The following parameters are adopted:

- The list of all conventional and non-VRE-based renewable power plants, their technical and economical parameters and their assignment to the corresponding network nodes.
- The list of all hydro-storage power plants and their technical parameters.
- The distribution of the capacity of the renewable producers among the respective nodes.
- The parameters for splitting the time series representing the total load of Germany among the individual nodes.
- The structure of the German grid as well as the assumed line parameters.

In modeling, there are differences in the following points in particular:

- The feed-in of VRE-based power plants is modeled based on the location-dependent weather provided by the INFORMATION LAYER.

Entity / Aspect	Number	Modeling approach ELMOD-DE	MOCES Modification
power lines / nodes	697 / 438	linear model for transmission lines without losses	no modification
power system dynamics	-	not modeled	swing equations based on model and parameters given in Weißbach [51]
frequency control	-	not modeled	frequency control with primary and secondary control; control power is provided by individual power plants
renewable energy sources	1387	time series for each type of source and node	individual models for representative plants, cf. Sec. 5.4.1
conv. power plants	562	ideal power source	inclusion of primary and secondary controller for grid frequency; limitation of power ramp
storage (hydro storage)	32	idealized storage with three parameters: storage capacity, maximum power, overall efficiency	no relevant modifications
load	394	time series for each individual node	no relevant modifications
cross-border exchange	-	time series for each cross-border transmission line, based on historic values	not yet modeled
local optimizer (LOP)	3|46 *	not modeled	storage system with: $\eta_{\text{in}} := 0.98$, $\eta_{\text{out}} := 0.98$, $C := P_{\text{peak}}^{\text{wind+pv,node}} \cdot \frac{24\text{h}}{10}$, $P_{\max} := \pm C \cdot \frac{1}{6\text{h}}$

Table 8: Overview of the MOCES extensions to ELMOD-DE that are part of the PHYSICAL LAYER of MOCES. * Number of LOCAL OPTIMIZERS in scenarios `3Lop`, `3LopErr` | `ManyLop`, `ManyLopErr`.

- The BMP and related processes are modeled explicitly. This results, i.a., in an explicit modeling of weather and market forecast services as well as a modeling of the day-ahead market and its daily clearing mechanism based on the provided bids and offers. The new model therefore neither assumes a PERFECT MARKET nor perfect forecasts of VRE-based production and consumption.
- The static grid model of ELMOD-DE is extended to include the response of the grid frequency to mismatches between production and consumption of electricity. The control mechanisms restoring the frequency are also modeled.

Individual aspects and their modeling will be described in more detail in the following section.

Entity / Aspect	Modeling Approach ELMOD-DE	MOCES Modification
energy market	The interactions between the entities are not modeled explicitly. They remain abstract and are part of the model constraints respectively the objective function if at all.	daily double-sided blind auction with individual contracts for each hour of day, clearing at 01:00 each day
planning phase		individual behavior, partly based on forecasts and linear optimization
settlement phase		simplified version of reBAP [50]
bidding strategy of conv. power plants		individual bidding strategy based on marginal price
bidding strategy of renewable power plants		sell at any price
bidding strategy of storage systems		individual bidding strategy based on market forecast for next day
strategy of local optimizer (LOP)	not modeled	increase internal consumption of node, cater for deviations

Table 9: Overview of the MOCES extensions to ELMOD-DE that are part of the BUSINESS PROCESS LAYER of MOCES.

5.4 PHASE 1: MODELING OF ASPECTS

In the following, the modeling of *Scenario Aspect Models* for the MOCES extension of ELMOD-DE is described. It shows how most individual aspects are modeled strictly according to the scenario modeling procedure described in Section 4.5, but also that it is possible to deviate from it if required (cf. Section 5.4.6) or judicious.

5.4.1 *Renewable Energy Sources*

For modeling VRE-based power plants, we follow the scenario modeling process described in Section 4.5. In the following, we will limit ourselves to a detailed description of the aspect of *wind power plants*, and additionally sketch the modeling of PV systems.

ADAPTION OF BASE MODEL To model the accumulated feed-in of all wind power plants assigned to one node, we use the sum of two instances of the MOCES *Base Model* `WindPlant`, cf. Figure 48. Those two instances are modeled in a way that they are representative for the behavior of all plants; they will thus also be called representative plants in the following. As Figure 48. shows, this *Base Model* is of type ***`Prosumer`*** and thus has the properties described on p. 157f. in Section 4.4.6.

Both *Base Models* model the power output of a wind turbine with its power curve, which is typically available in corresponding data sheets. Such a curve is mainly influenced by the physical property that the power increases with the third power to the wind speed, but can also be influenced by the concrete design of the rotor blades and the rest of the turbine. Therefore, one instance of the *Base Model* has the characteristics of a plant designed for areas with lower wind speeds, the other one the characteristics of a plant designed for areas with higher wind speeds. The differences between the curves used are shown in Figure 60.

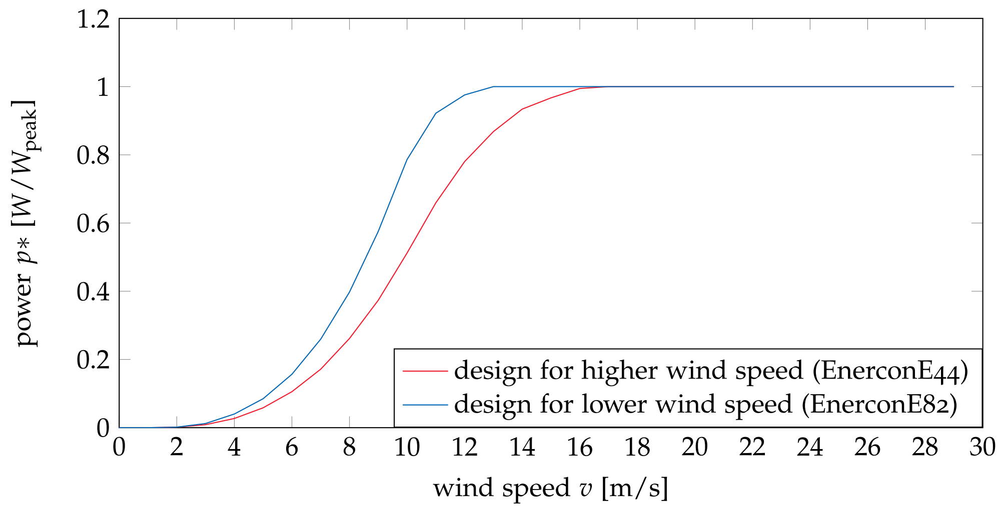


Figure 60: Typical power curves for wind power plants used in *Scenario Base Model*, data taken from [165].

The resulting *Scenario Base Model* for one node uses the following set of equations:

$$v_s = f(p_{\text{lat}}, p_{\text{long}}, t), \tag{5.1a}$$

$$v_d = f(p_{\text{lat}}, p_{\text{long}}, t), \tag{5.1b}$$

$$z_0 = f(v_d), \tag{5.1c}$$

$$\hat{v}_s = v_s \frac{\ln(z/z_0)}{\ln(z_r/z_0)}, \tag{5.1d}$$

$$p = p_{\text{peak}} \cdot p_t \cdot \left((1 - c_t) \cdot f_{\text{pc},1}(\hat{v}_s) + c_t \cdot f_{\text{pc},2}(\hat{v}_s)\right), \tag{5.1e}$$

with:

t	time,
$p_{\{\text{lat,long}\}}$	geographical positions,
v_s	wind speed at reference height z_r,
v_d	wind direction,
$\hat{v}_s$	wind speed at hub height,
z_0	roughness length/factor,
z	hub height,
z_r	reference height of wind speed,
$f_{\text{pc},\{1,2\}}(\cdot)$	normalized wind turbine power curve,
p_{peak}	accumulated peak power of wind power plants of node,

c_t tuner parameter $c_t \in \mathbb{R} \mid 0 \leq c_t \leq 1$,
p_t tuner parameter $p_t \in \mathbb{R} \mid 0.9 \leq p_t \leq 1.1$.

The model thus uses the values for wind speed and wind direction at the reference height of 10.00 m (Eqn. 5.1a and 5.1b) provided by the INFORMATION LAYER, and transforms the wind speed on the basis of the log wind profile, (Eqn. 5.1d) to the speed at hub height. This transformation depends on the parameter z_0, the so-called roughness factor, which describes the surface condition of the location where the wind speed is determined. The parameter z_0 is an example for a model parameter that is intended to be fitted in the step *Optimization and Parameter Fitting*, cf. Figure 58, to improve the accuracy of the model. In order to be able to consider the influence of different wind directions on the feed-in of the wind turbines, we formulate the surface roughness as a piece-wise linear function of the wind direction with the change points $(z_{0,k}, v_{d,k})_{k=0,\dots,18}$ with $v_{d,k} = (0°, 20°, \dots, 360°)$:

$$f(v_d) = \begin{cases} z_{0,0} + \frac{z_{0,1} - z_{0,0}}{20°} \cdot (v_d - v_{d,0}) & \text{if } v_d \in [v_{d,0} = 0°, v_{d,1}) \\ z_{0,1} + \frac{z_{0,2} - z_{0,1}}{20°} \cdot (v_d - v_{d,1}) & \text{if } v_d \in [v_{d,1}, v_{d,2}) \\ \dots & \\ z_{0,17} + \frac{z_{0,18} - z_{0,17}}{20°} \cdot (v_d - v_{d,18}) & \text{if } v_d \in [v_{d,17}, v_{d,18} = 360°] \end{cases} \tag{5.2}$$

The power output p_i for an individual node i therefore depends on a set of parameters from which only some can be derived from the existing ELMOD-DE data set:

$$p = f\left(\underbrace{p_{\text{lat}}, p_{\text{long}}, t, z_r, p_{\text{peak}}}_{\substack{\text{derived from}\\ \text{known parameters}}}, \underbrace{p_t, c_t, z, z_{0,0}, \dots, z_{0,18}}_{\substack{\text{tuned during}\\ \text{optimization}}}\right) \tag{5.3}$$

Consequently, the parameters of the representative plants are not known and can only be estimated based on assumptions and available data.

MODEL GENERATION FOR OPTIMIZATION AND PARAMETER FITTING
In the model generation process, an external tool, cf. Section 5.5.1, generates four *Scenario Aspect Models*. Each represents one of the four MARKET BALANCE AREAS and includes all nodes belonging to this area. The models of the individual nodes are configured with the parameters provided by ELMOD-DE if available.

OPTIMIZATION AND PARAMETER FITTING The *Scenario Aspect Models* generated in the previous step each have between 704 (Transnet BW) and 2552 (Amprion) parameters that have to be tuned in the optimization step. In theory, the parameters of the nodes could be tuned individually and independently of each other. However, the node-specific historic feed-in values required for this approach are not available. We therefore resort to feed-in time series for the entire control area $p_{\text{tso},k}^{\text{meas.}}(t)$, made available by the transparency platform of the TSO [156], and compare them with the sum of all nodes that belong to the respective area $p_{\text{tso},k}^{\text{model}}(t)$ during the model parameter fitting process.

The published time series $p_{\text{tso},k}^{\text{meas.}}(t)$ are themselves subject to errors, as they are not measured values. They are based on metered reference wind power plants that are scaled up to represent all units. Unfortunately, neither the algorithm used nor any analysis of its quality are publicly available. Obviously, however, there is a significant difference between the yearly feed-in values, calculated by integrating the time-series, and the official amounts published in the *EEG Jahresabrechung*[2], also provided by the TSO [156]; Table 10 exemplary shows the difference for wind-onshore systems. It is noticeable that the energy quantities of the time series are systematically lower, sometimes significantly, than the actual feed-ins.[3]

	50 Hertz		**Amprion**		**Tennet**		**Transnet BW**	
	Energy [TWh]	Diff. [%]	Energy [TWh]	Diff. [%]	Energy [TWh]	Diff. [%]	Energy [TWh]	Diff. [%]
E_{ts}	19.77		7.99		21.85		0.70	
E_{EEG}	21.57	+9.08	10.06	+25.93	23.55	+7.75	0.74	+5.50

Table 10: Difference between the integral of the feed-in time series E_{ts} ($\int p_{\text{tso},k}^{\text{meas.}}(t)dt$) and the values given in the *EEG Jahresabrechung* E_{EEG}.

Since the energy quantities in the *EEG Jahresabrechung* are based on actual measurements, we have decided to scale the time series $p_{\text{tso},k}^{\text{meas.}}(t)$ linearly such that the energy quantities of the time series correspond to the quantities of the annual statement.

For fitting the parameters of the *Scenario Aspect Models*, we use a bounded BROYDEN–FLETCHER–GOLDFARB–SHANNO (BFGS) algorithm, which is available within the optimization library for DYMOLA, see Pfeiffer [160].

2 German for annual account of the EEG.

3 In a telephone conversation with an employee of 50 Hertz, who was named as contact person for the portal netztransparenz.de, the systematic deviation was justified by the fact that only those systems are included in the extrapolation of the feed-in which were reported to the grid operator. Since there is a certain period of time between system commissioning and the reporting date, the feed-in is systematically underestimated.

Two error measures are used during training, the ROOT MEAN SQUARE ERROR (RMSE) in a continuous formulation normalized by the installed peak power p^{peak}:

$$\text{rRMSE}^{\text{conti.}} = \frac{1}{p^{\text{peak}}} \sqrt{\frac{\int_{t1}^{t2} \left(p^{\text{model}}(t) - p^{\text{meas.}}(t)\right)^2 dt}{t_2 - t_1}} \tag{5.4}$$

and the energy bias:

$$\text{BIAS} = \int_{t^1}^{t^2} p^{\text{model}}(t)dt - \int_{t^1}^{t^2} p^{\text{meas.}}(t)dt. \tag{5.5}$$

For the training, we employ a two-step approach. First, we minimize the RELATIVE ROOT MEAN SQUARE ERROR (RRMSE). For this step, we split the whole data set of 2014 into a training and a validation set. The training set covers the first half of the year; the validation set covers the second half of the year. Second, we minimize the energy bias by using the whole data set of one year. In the second step, only one parameter that scales the model output linearly is tuned.

The result of the optimization process is summarized in Table 11. It clearly shows an improved model accuracy in comparison to the baseline. In contrast to the continuous formulation of RRMSE used for training, cf. Equation 5.4, a discrete formulation is used here:

$$\text{rRMSE}(\Delta T) = \frac{1}{p_{\text{tso},k}^{\text{peak}}} \sqrt{\frac{\sum_{t^*=1}^{N-1} \left(\overline{p_{\text{tso},t^*}^{\text{model}}} - \overline{p_{\text{tso},t^*}^{\text{meas}}}\right)^2}{N-1}}, \tag{5.6a}$$

$$\overline{p_{t^*}} = \frac{1}{\Delta T} \int_{t^*\Delta T}^{(t^*+1)\Delta T} p(t)dt, t^* \in \mathbb{N} | 0 \leq t^* \leq N-1. \tag{5.6b}$$

It represents the error between time-equidistant time series based on piece-wise constant mean values of the continuous time-series $p(t)$ and a chosen interval ΔT. The error thus decreases with larger ΔT.

The scenario modeling process described for the wind turbines was also used to model the entirety of all PV systems. The corresponding data for parameter fitting also originates from the four TSOs. The performance is shown in Table 12.

In general, we expect to obtain even better results with the same models if feed-in time series are available with higher spatial resolution. In that case, the models of the representative plants could be trained individually.

5.4.2 *Conventional Power Plants*

The *Scenario Base Model* for a single conventional power plant used in the simulation is based on MOCES' *Base Model* for conventional power

	-	RRMSE [%] / Improvement [%]							
	-	$\Delta T = 900s$		$\Delta T = 3600s$		$\Delta T = 14400s$		$\Delta T = 86400s$	
Area	**Model**	Train	Vali.	Train	Vali.	Train	Vali.	Train	Vali.
Tennet	Moces	**7.15**	**6.40**	**7.11**	**6.36**	**6.66**	**5.98**	**2.91**	**2.99**
	Baseline	18.39	16.65	18.37	16.64	18.22	16.53	16.62	15.33
	Improvement	−61.10	−61.56	−61.31	−61.77	−63.44	−63.85	−82.46	−80.48
50 Hertz	Moces	**8.18**	**6.69**	**8.13**	**6.65**	**7.65**	**6.25**	**3.07**	**2.69**
	Baseline	18.61	17.56	18.59	17.55	18.41	17.44	16.85	16.15
	Improvement	−56.02	−61.88	−56.25	−62.09	−58.46	−64.19	−81.76	−83.32
Amprion	Moces	**6.82**	**6.23**	**6.70**	**6.13**	**6.04**	**5.67**	**2.98**	**3.41**
	Baseline	18.55	16.77	18.53	16.76	18.29	16.59	16.16	14.68
	Improvement	−63.22	−62.86	−63.82	−63.40	−66.97	−65.82	−81.58	−76.77
Transnet BW	Moces	**8.81**	**7.60**	**8.69**	**7.47**	**8.03**	**6.85**	**3.87**	**4.22**
	Baseline	14.65	15.43	14.58	15.37	14.19	15.08	11.77	13.40
	Improvement	−39.89	−50.74	−40.42	−51.39	−43.41	−54.61	−67.08	−68.52
Germany	Moces	**6.72**	**5.55**	**6.68**	**5.52**	**6.29**	**5.19**	**2.42**	**2.08**
	Baseline	17.37	16.41	17.36	16.41	17.24	16.33	15.94	15.31
	Improvement	−61.34	−66.17	−61.51	−66.33	−63.50	−68.23	−84.80	−86.39

Table 11: Performance of the fitted wind power plant model for the year 2014 for different interval lengths ΔT. Bold numbers denote the accuracy of the fitted MOCES model in terms of RRMSE compared to a baseline model that assumes a constant power value calculated with $E_{\mathrm{tso},k}^{\mathrm{meas.,year}}/8760.00\,\mathrm{h}$ in normal typeface. Numbers in green show the reduction of the error in percent of the baseline model.

plants called `PowerPlant` , cf. Figure 48. Again an external tool is used to generate the Modelica code for the *Scenerio Aspect Model* representing all power plants based on the ELMOD-DE data. Since, in contrast to the VRE-based power plants, each power plant is modeled individually and parameters are available for each plant, it was not necessary to further optimize the parameters. In total, this aspect consists of 562 individually modeled power plants.

5.4.3 *Hydro Storage System*

The modeling of the 32 hydro storage systems is based on the MOCES *Base Model* `BatteryStorage`, presented in detail as running example in Section 4.4, cf. pages 160 ff. and 176 ff. No changes were made in the adaptation step and the parameters for the individual entities were adopted from ELMOD-DE without changes.

-	-	RRMSE [%] / Improvement [%]							
		$\Delta T = 900$s		$\Delta T = 3600$s		$\Delta T = 14400$s		$\Delta T = 86400$s	
Area	**Model**	Train	Test	Train	Test	Train	Test	Train	Test
Tennet	Moces	**2.00**	**1.89**	**1.95**	**1.84**	**1.63**	**1.56**	**0.65**	**0.59**
	Baseline	16.19	13.85	16.13	13.80	15.32	13.12	6.34	6.00
	Improvement	−87.67	−86.39	−87.93	−86.67	−89.36	−88.15	−89.76	−90.20
50 Hertz	Moces	**2.55**	**1.87**	**2.48**	**1.81**	**2.11**	**1.50**	**1.01**	**0.61**
	Baseline	17.36	14.60	17.29	14.55	16.39	13.82	6.91	6.39
	Improvement	−85.33	−87.21	−85.68	−87.55	−87.16	−89.12	−85.39	−90.49
Amprion	Moces	**2.84**	**2.79**	**2.78**	**2.74**	**2.33**	**2.35**	**1.15**	**1.04**
	Baseline	16.67	14.20	16.60	14.15	15.76	13.43	6.43	6.02
	Improvement	−82.95	−80.37	−83.27	−80.67	−85.23	−82.51	−82.06	−82.72
Transnet BW	Moces	**3.91**	**3.89**	**3.82**	**3.79**	**3.21**	**3.15**	**1.59**	**1.36**
	Baseline	17.45	14.10	17.37	14.04	16.46	13.31	6.74	5.92
	Improvement	−77.57	−72.43	−78.03	−73.00	−80.52	−76.33	−76.47	−77.09
Germany	Moces	**1.77**	**1.62**	**1.74**	**1.59**	**1.45**	**1.35**	**0.62**	**0.48**
	Baseline	16.48	13.93	16.42	13.88	15.59	13.20	6.32	5.97
	Improvement	−89.23	−88.34	−89.43	−88.54	−90.68	−89.79	−90.13	−92.03

Table 12: Performance of the fitted PV model for the year 2014 for different interval lengths ΔT. Bold numbers denote the accuracy of the fitted MOCES model in terms of RRMSE, compared to a baseline model that assumes a constant power value calculated with $E_{\text{tso},k}^{\text{meas.,year}} / 8760.00\,\text{h}$ in normal typeface. Numbers green show the reduction of the error in percent of the baseline model.

5.4.4 *Weather Forecast*

In order to be able to investigate the effect of erroneous feed-in forecasts for renewable energies, the MOCES model extends the ELMOD-DE model with a MENT which provides location-sensitive weather forecasts to all other MENTS. The feed-in forecasts for the respective renewable energy plants are determined by the individual MENTS on the basis of these forecasts.

The weather forecast is provided by one instance of the MOCES *Base Model* `SimpleWeatherForecast`, cf. Figure 48, which is used without further modifications. We will briefly describe it in the following.

In essence, the forecasts are based on actual weather data provided by the INFORMATION LAYER of MOCES, see Section 4.4.5. On top, several kinds of errors are added: a temporal shift; a RANDOM WALK describing errors in the global forecast run; and, a RANDOM WALK describing errors in the local forecast run.

The basic behavior implemented in the *Business Node* is summarized as follows: n times per day, a global forecast run is started for each forecast variable. The location-specific forecasts including the different types of error are generated upon request.

In detail, the forecast $\hat{x}(t)$ for a time series representing a weather attribute such as wind speed is calculated based on the actual time series $x(t)$ as follows:

$$\hat{x}(t) = x(t - X_{\text{t-shift}}(\sigma_{\text{t-shift}}, \mu_{\text{t-shift}})) + \underbrace{\mathtt{RW}_{\text{global}}(t - t_0, \sigma_{\text{gf}}, \mu_{\text{gf}})}_{\substack{\text{Random Walk for} \\ \text{error of global forecast run}}} + \underbrace{\mathtt{RW}_{\text{local}}(t - t_0, \sigma_{\text{lf}}, \mu_{\text{lf}})}_{\substack{\text{Random Walk for} \\ \text{error of local forecast run}}} \quad (5.7)$$

with:

t simulated WALL TIME,
t_0 current simulated WALL TIME at the time of forecast creation,
$X_{\text{t-shift}}$ random variable, fixed for all forecasts of one global forecast run,
$\sigma_{\{\text{lf,gf,t-shift}\}}$ standard deviation used by Gaussian random number generator, adjustable parameter,
$\mu_{\{\text{lf,gf,t-shift}\}}$ mean value used by Gaussian random number generator, adjustable parameter.

Both RANDOM WALKS $\mathtt{RW}_{\text{global}}$ and $\mathtt{RW}_{\text{local}}$ are formulated without offset and with a finite step length[4]:

$$x[t'] = \sum_{j=1}^{t'} X_j \ , \ t' := \frac{(t - t_0)}{\Delta T} \quad (5.8)$$

with:

X_j random variable based on a normal distribution with the parameters μ and σ,
ΔT time step length, typically 15.00 min.

4 To improve the readability, we use a representation of the relation between the continuous time t and the individual discrete steps of the random walk t' as well as the sum of a continuous signal and the discrete random walk that is not entirely accurate. In the implementation of MOCES, a pragmatic approach is followed where the random walk represents the sampling points of a piece-wise linear function.

In order to control the error in the predictions, the parameters of the distribution (σ, μ) used by the Gaussian random number generators are individually adjustable.

The chosen procedure allows for a basic description of typical error patterns of the individual forecasts, e.g.:

- Time-shifted ramps of wind speed: $\sigma_{\text{t-shift}} \gg 0$.
- Error of the temperature forecast continuously increasing over the forecast horizon, but no sudden changes: $\mu_{\{\text{gf,lf}\}} > 0$, $\mu_{\{\text{gf,lf}\}} \to 0$.
- Sudden changes in the error of the irradiation forecast due to cloudy situations: $\sigma_{\{\text{gf,lf}\}} \gg 0$.

If desired, MOCES also provides the possibility to choose the mean values randomly, again based on a normal distribution.

The method used here to model forecasting errors is motivated by representing typical error patterns as simply as possible. While certainly plausible, the method itself was not validated within the scope of this work. Therefore, it remains to be checked how well the modeled errors correspond to real errors before the model is used in concrete analyses.

5.4.5 *Market Forecast*

Market forecasts, more precisely the forecast of the price curve of the day-ahead market, are needed by the storage systems to place profitable bids on the market, cf. Section 4.4.9.1. These predictions are provided by an instance of the MOCES *Base Model* `SimpleMarketForecast`. For this model, we are confronted with a chicken and egg problem that arises from the explicit modeling of the market on the basis of binding bids; in the original ELMOD-DE model this problem does not exist due to the simplifying assumption of a PERFECT MARKET. First of all, the market prices are the result of interactions between the individual entities, whose behavior at the market is in turn affected by forecast market prices. Second, the market forecast entity strives to make a statement about an aspect of the system whose behavior is neither known nor understood.[5] While both aspects also exist in the real world, the second one is increasingly emerging in the context of MS. This is because, in the real world, market forecasts can be based on historical data and on models of the existing system, which can be validated. However, for a model of a system that deviates from the current reality, in our case the models in the scenarios `ManyLop` and `ManyLopErr`, no historical data exists.

We tackle this problem by designing the market forecast entity around a NARX model implemented as an ANN. The forecast is therefore based

5 Remember here that this is the motivation why we perform a simulation of the whole system in the first place.

on the past values of the variable to be forecast –the day-ahead market price– and other external variables that are assumed to be known. In the specific case, we use weather forecasts, from 11 locations distributed across Germany, as well as information about the hour of the day, day of the week and day of the year.

The inference of the ANN model can therefore be carried out easily during the simulation run on the basis of the market price that would be generated during the simulation and the weather forecasts provided by the `SimpleWeatherForecast` MENT. However, the training of the models is problematic, as there is no training data for the market price available.

We have therefore decided to train the model with data generated during simulation of an adapted ELMOD-DE version without the energy storage components. The motivation for this is the assumption that in this way at least the basic interrelation of the market, especially the influence of renewable energies on the fluctuating load, can be learned by the model. Figure 61 gives an impression of the quality of the forecast. It is an excerpt from a simulation run in scenario `3Lop`.

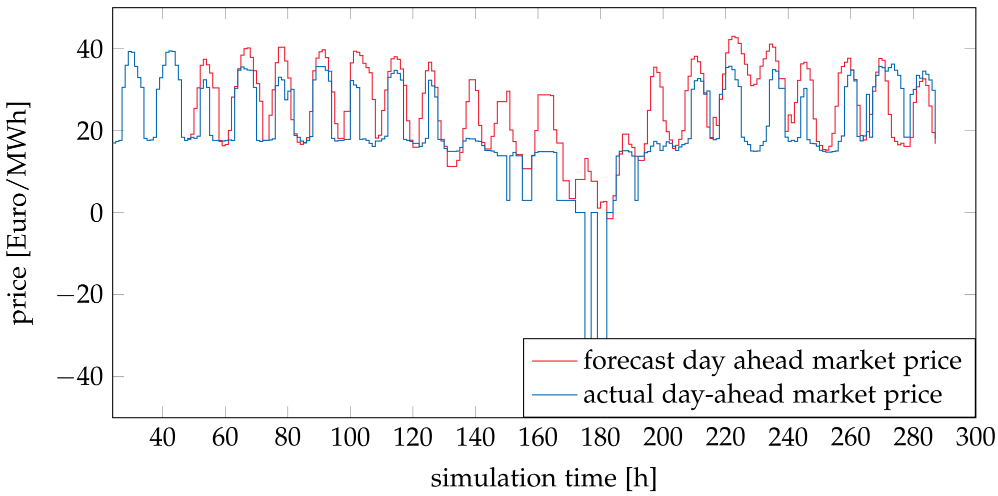


Figure 61: Predicted vs. actual market prices. The time period 144.00 h – 192.00 h, that also shows negative prices, aligns with the weekend.

5.4.6 *Grid Modeling*

The modeling of the grid is a peculiarity, since strict adherence to the scenario modeling process described in Section 4.5 as well as the modeling as a complete MENT was dispensed with. The reasons for this are as follows. The grid is modeled without actuators that influence the behavior of the grid. The linked business node could therefore not execute any action on the grid. Likewise, the grid operator's interaction with other components on the BUSINESS PROCESS LAYER is not modeled in detail. The linked component would therefore be meaningless. A component-oriented modeling of the network using parameterizable basic models for lines and transformers was evaluated, but failed due to the performance of DYMOLA with regard to large models. Therefore, the calculation of the grid state is included into a single model which uses an optimized algorithm by utilizing the external function feature of MODELICA.

The MOCES adaption of the ELMOD-DE model uses the same formulation as the original, a linear power flow model without losses, but extends it with the swing equation to calculate the dynamic behavior of the net frequency. This approach was chosen as a trade-off between simulation accuracy and simulation speed.

STATIC MODEL The linear power flow model is an analogy to a DIRECT CURRENT (DC) power flow model, where transmission lines are represented as a kind of *ohmic resistance*, and loads and power sources are ideal current sources feeding the nodes. The used formulation is based on the procedure given by Andersson [115, p. 49] and can be derived from the power balance equation introduced in Section 3.2.4.2. We start with the already simplified formulation for the power flow given by Equation 3.9 and neglect the reactive power.

$$P_{n,m} = P_{m,n} = -b_{n,m} U_n U_m \sin\left(\theta_n - \theta_m\right) \tag{5.9}$$

thus describes the power flow $P_{n,m}$ from the n^{th} node the m^{th} node and vice versa with:

- $b_{n,m}$ susceptance of the transmission line connecting node n with node m,
- $U_{\{n,m\}}$ voltage at node,
- $\theta_{\{n,m\}}$ voltage angle at node.

The line susceptance is given by[6]:

$$b_{n,m} = \frac{x_{n,m}}{r_{n,m}^2 + x_{n,m}^2} \tag{5.10}$$

with:

- $r_{n,m}$ series resistance of transmission line,
- $x_{n,m}$ series reactance of transmission line.

When using the per-unit system, choosing $U_n = 1$ pu, and assuming $U_n = U_m$ as well as $\sin(\theta_{n,m}) \approx \theta_{n,m}$, the equation system built by the line equations can be expressed in a matrix form as follows:

$$\mathbf{P}_{\text{pu}} = \mathbf{B}_{\text{pu}}^{'} \boldsymbol{\theta} \tag{5.11}$$

with the matrix and vector elements representing the set of all nodes $\mathcal{N}$:

6 Typically it holds that $x_{n,m} \gg r_{n,m}$, so the equation is often further simplified to $\frac{1}{x}$.

p_n power introduced at node n,
θ_n voltage angle at node n,

$$b'_{n,m} = \begin{cases} \sum_{\substack{k=1,2,\ldots,|\mathcal{N}| \\ k \neq n}} b_{n,k} & \text{if} \quad k = m \\ -b_{k,m} & \text{if} \quad k \neq m \end{cases}.$$

The matrix $\mathbf{B}'_{\text{pu}}$ is singular, meaning that their is no unique solution for the equation system in Eqn. 5.11. To overcome this problem, we remove p_1 and set θ_1 to 0, thus eliminating the first row and column of the matrix $\mathbf{B}'_{\text{pu}}$. By doing this, we add the so-called slack or swing bus to the system which gives the reference for the voltage angle and balances the differences between power production and consumption. As the reduced matrix $\mathbf{B}'_{\text{pu}}$ is positive definite, a Cholesky decomposition can be used for efficiently solving the equation system. With the given $\boldsymbol{\theta}$, determining the power p_1 required to balance the system is straightforward:

$$p_1 = \sum_{k=1}^{|\mathcal{N}|} y_{1,k}\theta_k. \tag{5.12}$$

GRID DYNAMICS The modeling of the network dynamics also follows Section 3.2.4.2 and the swing equation described there. This approach interprets the distributed rotational inertias of all synchronous machines connected to the grid as one single flywheel, rotating with the grid frequency f and accelerated by the difference between power injection and load consumption $P_1(t)$ at the slack node.

We use the general momentum balance as given by Equation 3.11 in its typical modified form used for grid modeling purposes, cf. Weißbach [51]:

$$T_{\text{AN}^*}\frac{df^*}{dt} - k_{\text{pf}^*}\left(1 - f^*\right) = p_1^*(t). \tag{5.13}$$

This formulation is derived from the general moment balance assuming small changes of the grid frequency f. In addition, the equation is transformed into a normalized form with: $f^* = f/f_{\text{n}} = f/50.00\,\text{Hz}$, $p_1^* = P_1/P_{\text{n}}$ with P_{n} being the nominal grid load. The parameter T_{AN^*} is known as *system start time constant*, as it describes the time needed to accelerate the generators from standstill to nominal speed with nominal torque; in the synchronous grid of continental Europe, this constant is about 10.00 s [51]. k_{pf^*} represents the self-regulation effect of the grid due to loads of the type of synchronous or asynchronous machines. Those loads decrease with the grid frequency; this is about 1.00 % up to 1.50 % according to Weißbach [51].

CONTROLLED GRID The mechanism for stabilizing the grid is outlined in Section 2.2.8.2 and sketched in Figure 22. In the developed MOCES model, the power P_{c} used to stabilize the grid frequency and to cater for deviations between production and consumption is described with its most important elements. These are the primary control power $P_{c,p}$ and the secondary control power $P_{c,s}$:

$$P_c = P_{c,p} + P_{c,s}. \tag{5.14}$$

According to the model given by Weißbach [51], the primary control power is modeled by using a proportional controller with gain $k_{c,p}$ by:

$$P'_{c,p} = k_{c,p} \cdot (f_n - f) \tag{5.15}$$

and a first order element with the time constant $T_{c,p,d}$ to sum up the dynamics of the power plants providing the primary control power:

$$\frac{dP_{c,p}}{dt} = \frac{1}{T_{c,p,d}} \cdot \left(P'_{c,p} - P_{c,p}\right). \tag{5.16}$$

Here we deviate from the model given by Weißbach [51] who does not provide full information on how to model the dynamics of the activation of the primary control power. As primary control power must be fully available within 30.00 s, $T_{c,p,d}$ is set accordingly to about 10.00 s, as the output of a first order element is 95.00 % of the input after $3 \cdot T_{c,p,d}$.

The central controller used to determine the needed secondary control power $P'_{c,s}$ has the form of a PROPORTIONAL–INTEGRAL (PI) controller [51]:

$$P'_{c,s} = k_{c,s} \cdot E(t) + \frac{1}{T_{c,s}} \cdot \int E(t)dt. \tag{5.17}$$

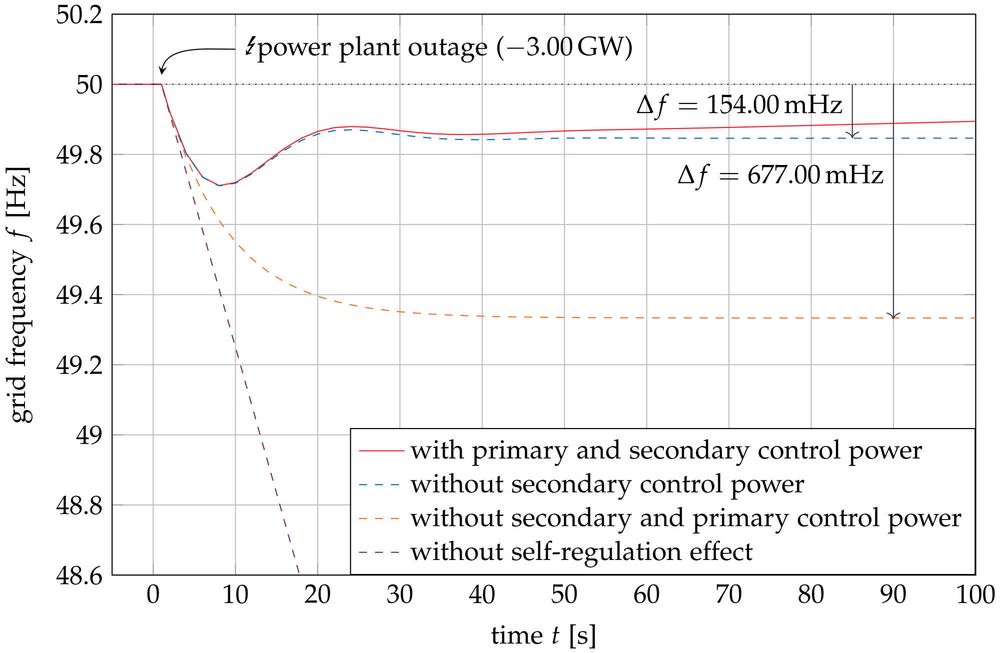
The control deviation $E(t)$, known as *area control error*, must be estimated as it cannot be measured. It is approximated with the following equation:

$$E(t) = \underbrace{P_{c,p}}_{\substack{\text{approximated activated}\\ \text{primary control power}}} + \underbrace{k_{pf} \cdot (f_n - f)}_{\substack{\text{approximated}\\ \text{self-regulation effect}}}. \tag{5.18}$$

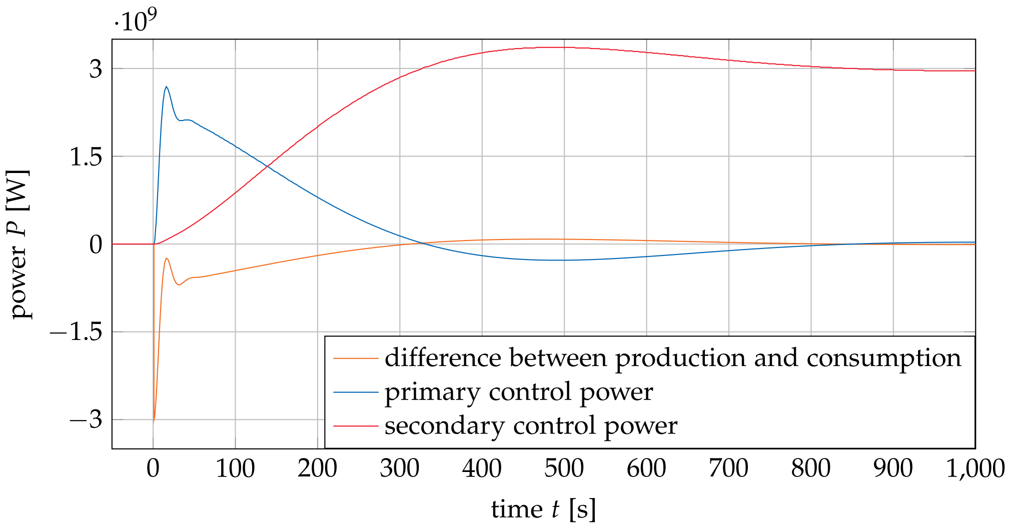
Again, unlike Weißbach [51], we use a first order element to model the delay in activation:

$$\frac{dP_{c,s}}{dt} = \frac{1}{T_{c,s,d}} \cdot \left(P'_{c,s} - P_{c,s}\right). \tag{5.19}$$

As secondary control power must be fully available within 300.00 s, $T_{c,s,d}$ can be set accordingly to about 150.00 s. Following Weißbach [51], guiding values for the parameters of the secondary controller are: $k_{pf} = 0.00 \ldots 0.50$, $T_{c,s} = 50.00\,\text{s} \ldots 200.00\,\text{s}$.



(a) Response of the grid frequency. Dashed curves show the frequency response without the named effects.



(b) Activated control power to restore the grid frequency.

Figure 62: Response of the grid frequency to a power plant outage at $t = 1.00$ s (a), and the reaction of the primary and secondary control power to it (b). The simulation uses the following parameters: $P_\text{n} = 150.00$ GW, $T_{\text{AN}^*} = 12.00$ s, $k_{\text{pf}*} = 1.50$, $k_{\text{c,p}} = 15.00\,\text{GW}\,\text{Hz}^{-1}$, $T_{\text{c,p,d}} = 10.00$ s, $k_{\text{c,s}} = 0.25$, $T_{\text{c,s}} = 125.00$ s, $T_{\text{c,p,d}} = 125.00$ s.

IMPLEMENTATION IN MODELICA The grid model is implemented in MODELICA as a single block with multiple inputs and outputs. Based on the power flows P_i for $i = 2...N$ given for the individual nodes of the grid, the model serves the utilization of each individual line as well as the grid frequency f and the activated primary and secondary control power $P_{c,p}$ and $P_{c,s}$ as output. This means that the dynamics of the power plants with regard to the activation of the control power are not modeled in the individual models of the power plants but in the central grid model. This decision was also made for performance reasons in order to reduce the number of state variables of the overall model, even if the procedure does not correspond to the idea of component-oriented modeling.

The power flow problem is solved by utilizing the Eigen library [166], the grid dynamics are formulated directly in MODELICA.

The load calculation was validated against data kindly provided by ELMOD-DE on request. The network dynamics determined by the model were validated with the results of Weißbach [51].

To give an insight into the modeling depth, Figure 62 shows the reaction to a simulated power plant outage.

5.5 PHASE 2: SCENARIO MODEL AND MODIFICATIONS TO IT

In this section, we will describe the second phase of MOCES' scenario modeling process, cf. Section 4.5, in particular Figure 59, for the scenario and research questions presented in Section 5.1.

5.5.1 *Generation of the Complete Scenario Model*

From the perspective of modeling in MODELICA, creating a *Scenario Model* from *Scenario Aspect Models* mainly involves instantiating, parameterizing and connecting individual models several hundred times. This step can no longer be performed manually by the modeler due to the high number of parameters (several thousand).

In MODELICA itself, there are some language constructs (for/while-loops, overloading of parameters/model) that basically allow to describe the generation/parameterization of the model in MODELICA itself.[7] However, in practice, the limits are reached quickly and the usage is not comfortable at all.

Apart from Nytsch-Geusen et al. [167] that use the web template engine *Jinja*[8], to the best of our knowledge, there are no publications that deal with this problem of MODELICA, preferably in a more general way. We have therefore decided to take a pragmatic approach for MOCES, in

7 In the performance test in Section 4.3.4 (p. 143), these patterns have been used. Cf. the corresponding model code in Appendix A.2.

8 `https://github.com/pallets/jinja`, last accessed: 10 April 2021.

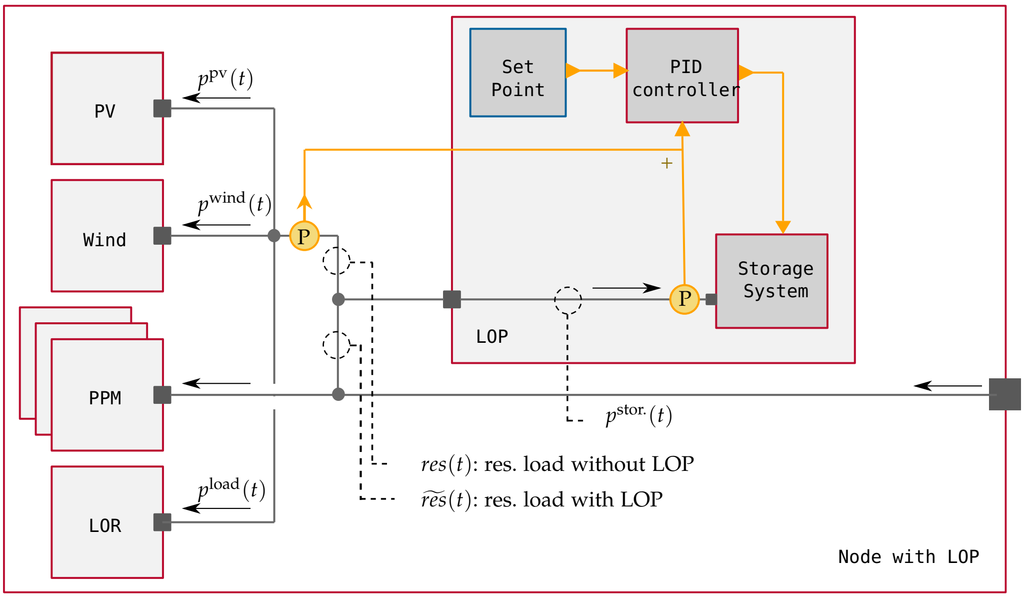


Figure 63: Structure of a node with a LOCAL OPTIMIZER entity. PPM (Power Plant Market) represents a conventional power plant assigned to the node; LOR (Load Region) models the load of the region represented by the node.

which the scenario models are created semi-automatically on the basis of configuration tables using scripts written in MATLAB. The result is a relatively flat MODELICA model without the annotations related to the graphical representation of the model, which are not helpful in our case.

To give an impression of the generated model code, Listing A.4 in the Appendix shows an extract from the overall MOCES ELMOD-DE model.

5.5.2 *Further Modifications*

To be able to model and simulate the scenarios introduced in Section 5.1, a modified version of a node representing a small region of Germany, cf. p. 189, has been developed and is shown in Figure 63. As can be seen in the figure, the core of the modification is the addition of a MENT called LOCAL OPTIMIZER (LOP). The measure which is influenced by the LOCAL OPTIMIZER is the residual load $res(t)$ which takes into account the VRE-based power plants and the load of the node. For the unaffected residual load of one node, the following applies:

$$res(t) = p^{\mathrm{PV}}(t) + p^{\mathrm{wind}}(t) + p^{\mathrm{load}}(t), \tag{5.20}$$

cf. Figure 63 for the variable names and the sign convention for the power flows ($p^{\{\mathrm{pv,wind,load,stor.}\}}, res, \widetilde{res}$) given by the black arrows.

For the residual load affected by the LOCAL OPTIMIZER, we use the notation:

$$\widetilde{res}(t) = res(t) + p^{\text{stor.}}(t). \tag{5.21}$$

A LOCAL OPTIMIZER has two objectives: to increase the local consumption of VRE-based energy and to cater for unplanned deviations from the planned residual load of the node. The *Physical Node* of the MENT for a LOCAL OPTIMIZER is therefore both a storage system and a controller which controls the storage and thus the deviations from the planned residual load, as long as the state of the storage allows. Its model is derived from the generic MOCES storage model given in Section 4.4.6.1. For the charging or discharging power set point of the battery, cf. variable `P_s` in Listing 4.8 (p. 160) therefore applies:

$$p^{\text{stor.}}_{\text{setpoint}}(t) = p^{\text{stor.}}_{\text{schedule}}(t) + \underbrace{f\left(\widetilde{res}(t) - \widetilde{res}_{\text{schedule}}(t)\right)}_{\text{Output of PID-Controller,cater for deviations}} . \tag{5.22}$$

The behavior of the corresponding *Business Node* is also based on the MOCES model of the storage system, cf. Section 4.4.9.1. However, more extensive modifications have been made here to model the LOCAL OPTIMIZER. The optimization problem to determine the schedule of the storage system $p^{\text{stor.}}_{\text{schedule}}[t]$ for the next day of has been newly formulated. It now uses a time step of 15.00 min and optimizes the exchange with the grid by minimizing the following objective function:

$$\min \sum_{t=96}^{191} |\underbrace{\left(res[t+1] + p^{\text{stor.}}_{\text{schedule}}[t+1]\right)}_{\widetilde{res}[t+1]} - \underbrace{\left(res[t] + p^{\text{stor.}}_{\text{schedule}}[t]\right)}_{\widetilde{res}[t]}|. \tag{5.23}$$

with $t \in 96...192$ representing the quarter hours of the next day.

The conditions in Eqn. 4.30 (p. 177) are mostly kept the same, only Eqn. 4.30d is replaced by a more relaxed condition:

$$0.4 \leq soc[192] \leq 0.6 . \tag{5.24}$$

The objective function essentially aims to ensures a uniform power consumption or production of the node and thus compensates for load or feed-in peaks. The boundary condition in Eqn. 5.24 guarantees that the storage device emits about the same amount of energy over the day as it consumes. In combination, both thus ensure that, if electricity production and consumption are equal for one node over the course of one day and the storage capacity of the LOCAL OPTIMIZER is sufficient, the residual load is balanced to approximately zero.

Figure 64 exemplarily shows the residual load of three nodes and thus the realization of the schedules found by the optimization algorithm.

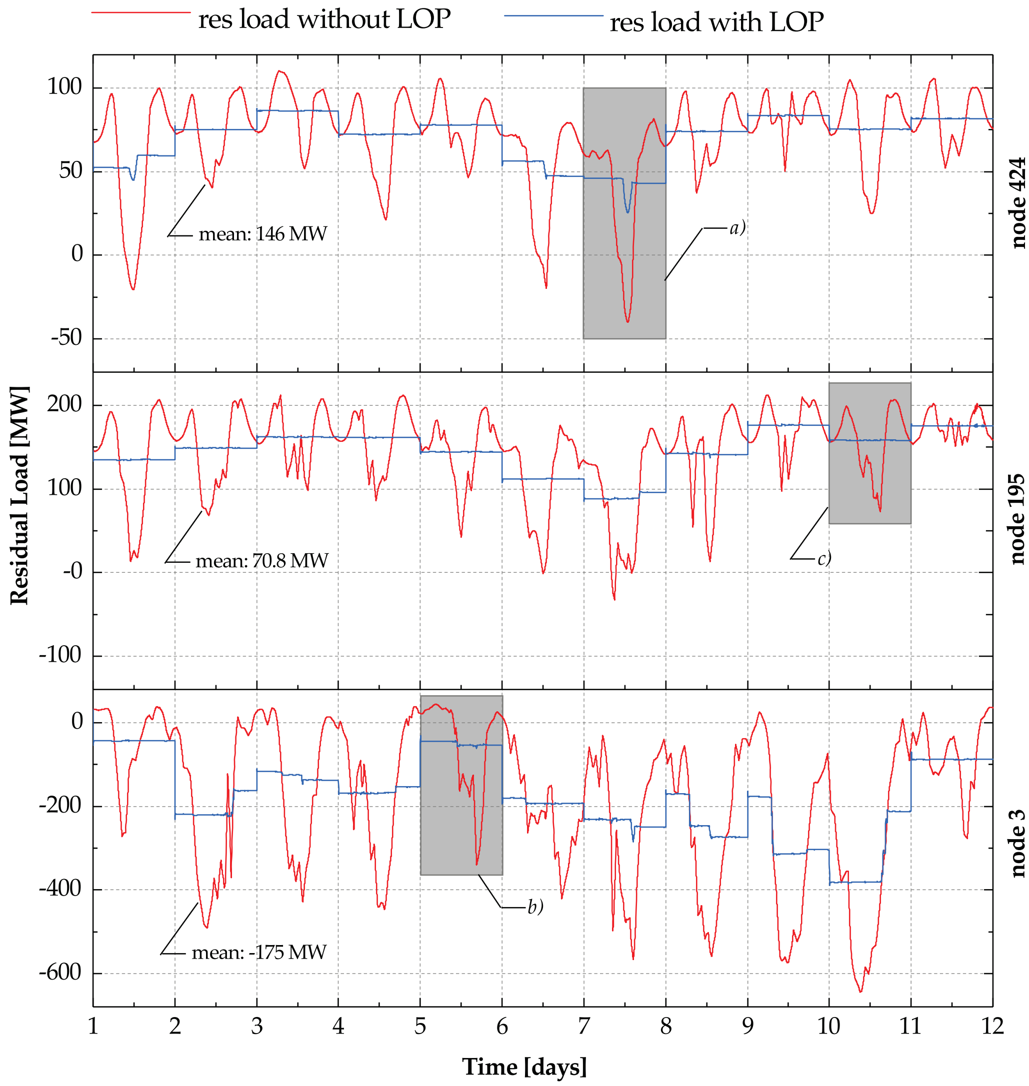


Figure 64: Exemplary temporal sequences of residual load with and without LOCAL OPTIMIZER.

The grey areas illustrate how the optimization algorithm works for different load situations. Area a) highlights a day on which the node would have been fed into the upstream grid at noon. This feed-in is avoided by the intervention of the LOCAL OPTIMIZER, and the energy is consumed locally instead. In addition, the maximum load of the node is reduced.

The area marked with b) shows the intervention of the LOCAL OPTIMIZER at node 3 that feeds on average more energy in the upstream grid than it draws from it. On the marked day, the LOCAL OPTIMIZER ensures that the electricity demand in the morning and evening is catered from the storage instead of from the grid. The day of node 195 marked with c) shows a continuous consumption of energy from the upstream grid. In this case, the LOCAL OPTIMIZER serves mainly to equalize the node energy.[9]

In order for the LOCAL OPTIMIZER to be able to carry out the optimization, it needs the planned behavior of the loads and renewable producers. As Figure 65 shows, a communication step was introduced for this purpose, in which the individual entities of the same node communicate their planned behavior to the LOCAL OPTIMIZER.

The optimization problem described so far is solved at the "do planning" step. The schedule $p^{\text{stor.}}_{\text{schedule}}[t]$ is also used by the LOCAL OPTIMIZER to place bids and offers at the day-ahead market. As given in Figure 65, the LOCAL OPTIMIZER does not send any information to the wind plant nor does a direct coupling between both entities exists. The LOCAL OPTIMIZER simply ensures that it buys or sells electricity at the global energy market if it is needed respectively available locally.

5.6 SCENARIO SIMULATION RESULTS AND FINDINGS

In this section, we will give a detailed insight into the simulation results for the four scenarios `3Lop`, `3LopErr`, `ManyLop` and `ManyLopErr`, cf. Section 5.1. All four scenarios are considered for the period from May 6, 2014 to May 16, 2014, plus two additional days before that time period to *ramp-up* the simulation. We selected this period because of the high feed-in of VRES as visualized in Figure 66. The figure also shows that the peak of feed-in is reached on Sunday, May 11. At the same time, this is also the day with the lowest load. The scenario is therefore representative for a POWER SYSTEM with a high proportion of renewable generators.

The presentation and discussion of the simulation results has two objectives. First, we would like to illustrate the simulation depth and eval-

9 Of course, there are alternative strategies that the LOCAL OPTIMIZER component could pursue to increase the amount of self-consumed energy of the node. Possibly some of those would show a better overall performance compared to the simple but robust solution presented here, or they might be more suitable for individual types of nodes or days. A comparative study of different strategies is left for future work.

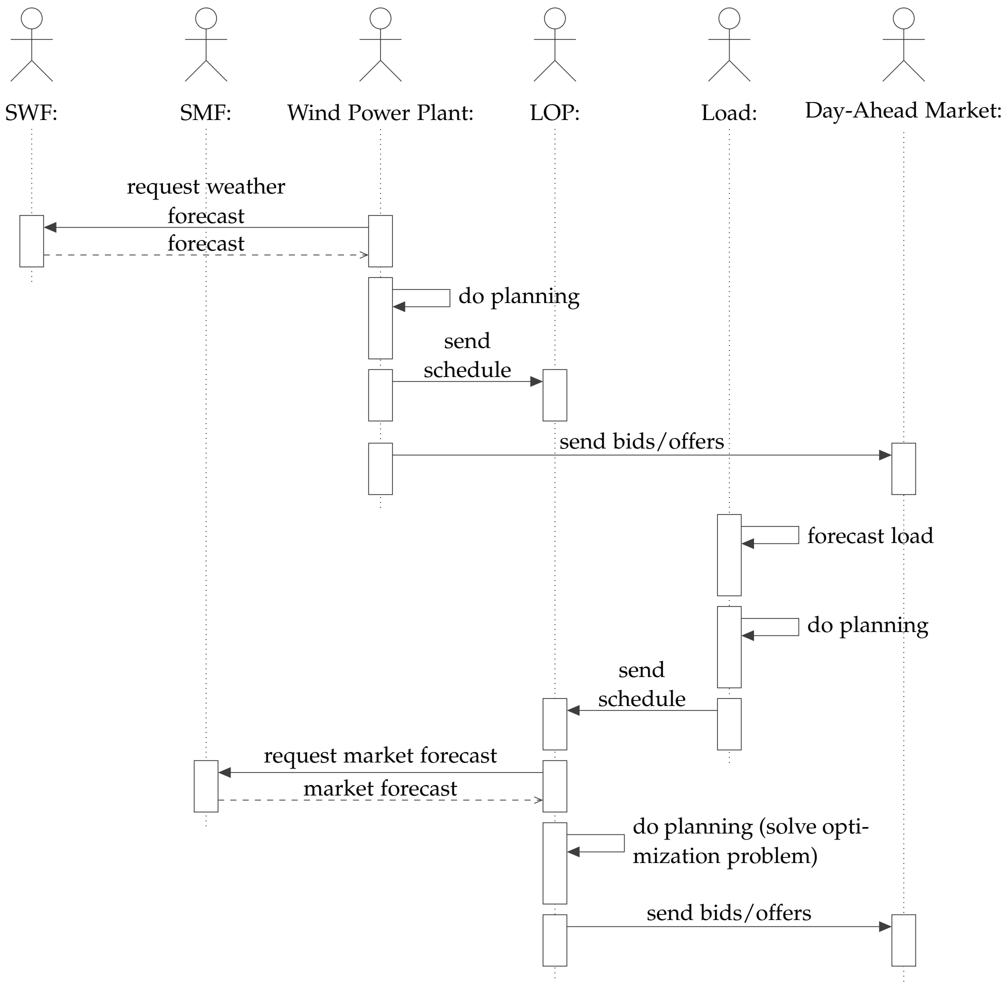


Figure 65: Sequence diagram of the planning phase of the BMP as modeled in the investigated scenarios. The diagram describes the specific communication between the PARTIES of one node, that is generically described in Figure 21. It focuses on the party-dependent behavior given by the two upper fragments of Figure 21 specifically. Accordingly, the PARTIES SMF (Simple Market Forecast) and SWF (Simple Weather Forecast) are examples of parties assuming the ROLE ESC (Energy Service Company), cf. p. 51. Wind Power Plant, Load, and the LOCAL OPTIMIZERS assume the ROLES BRP, BALANCE SUPPLIER and *Trader*.

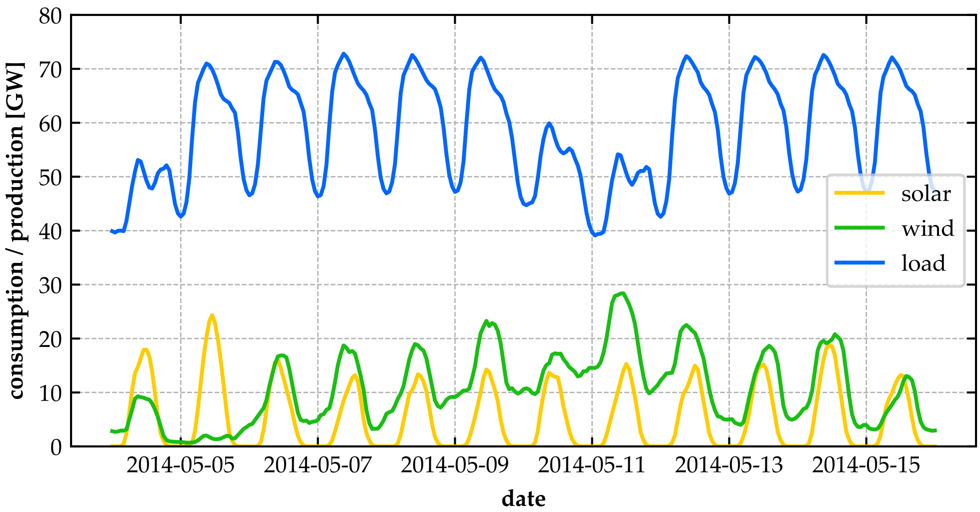


Figure 66: Scenario boundary conditions with regard to load and vre-based production.

uate the ms quality qualitatively. We will discuss these points mainly for the scenario `3Lop` because it reflects the non-modified, actual existing system. Second, we will discuss the questions posed at the beginning of the chapter about the change in overall performance when individual units formulate *decentrality* as a goal. For this purpose, we will consider the impact of the introduction of local optimizers on different aspects such as the electricity mix, the market and the grid.

5.6.1 *System-Wide Energy Mix*

general observations Figure 67 shows the typical and expected energy mix of generation. It ultimately results mainly from the different marginal costs of the various energy sources. Primarily the nuclear, biomass, run of river, and partially also the lignite-fired power plants cover the base load, while hard coal-fired power plants and also parts of the lignite-fired power plants cover the daily load peaks. However, as the graphs show, in the scenario under consideration, a large part of the load peaks is absorbed by the relatively high feed-in of the pv systems at these times. On Saturday, most of the lignite-fired power plants reduce their output, on Sunday the same holds even for the nuclear power plants.

Power plants that use the energy sources gas and oil are never "in the money" in the period under consideration, i.e. their marginal costs are always above the prices that are formed on the market. Nevertheless, these types also feed electricity into the grid, as the figure shows. The reason for this is the lower "must-run" limit of each power plant assumed in the model, which was set to 20.00 % of maximum capacity for all conventional power plants in the absence of an exact data basis. This parameter has a decisive influence on the formation of prices on the electricity market and the entire system. In the model, this parameter can be set individually and time-dependent for each power plant to increase the accuracy of the model. These more than 500 parameters should therefore be fitted in order to make reliable statements about the actual system.

INFLUENCE OF LOCAL OPTIMIZERS As the comparison of the two graphs in Figure 67 shows, the influence of the number of LOCAL OPTIMIZERS on the energy mix is not fundamental but clearly visible. There are indeed slight changes in the composition of the energy mix: the production of electricity from lignite increases by 2.00 %, that from hard coal decreases by 5.00 %. The feed-in from pumped storage power plants is also reduced by about 5.00 %. The production from other energy sources remains largely unchanged.

Interestingly, more LOCAL OPTIMIZERS do not generally lead to a reduction in peak load. This can be seen on the days May 5-7, when the peak load increases by several GW. This increase can be better understood when considering the cumulative behavior of all LOCAL OPTIMIZERS shown in Figure 68. Even though the behavior of the LOCAL OPTIMIZERS is highly individual, they tend to store electricity around noon, thus in times in which the load is high anyway, and to discharge in the evening hours. Their behavior is therefore different from that of pumped storage power plants, which is also shown in Figure 68. These have a very similar behavior but, as expected, use the night for charging the storage. Figure 69 zooms into May 8 such that the different behaviors can be better seen. This figure also shows that on this day in the evening hours about the same power is fed into the LOCAL OPTIMIZERS as is taken out of the pumped storage power plants. Whether such a behavior is reasonable from the overall systemic point of view remains to be critically examined.

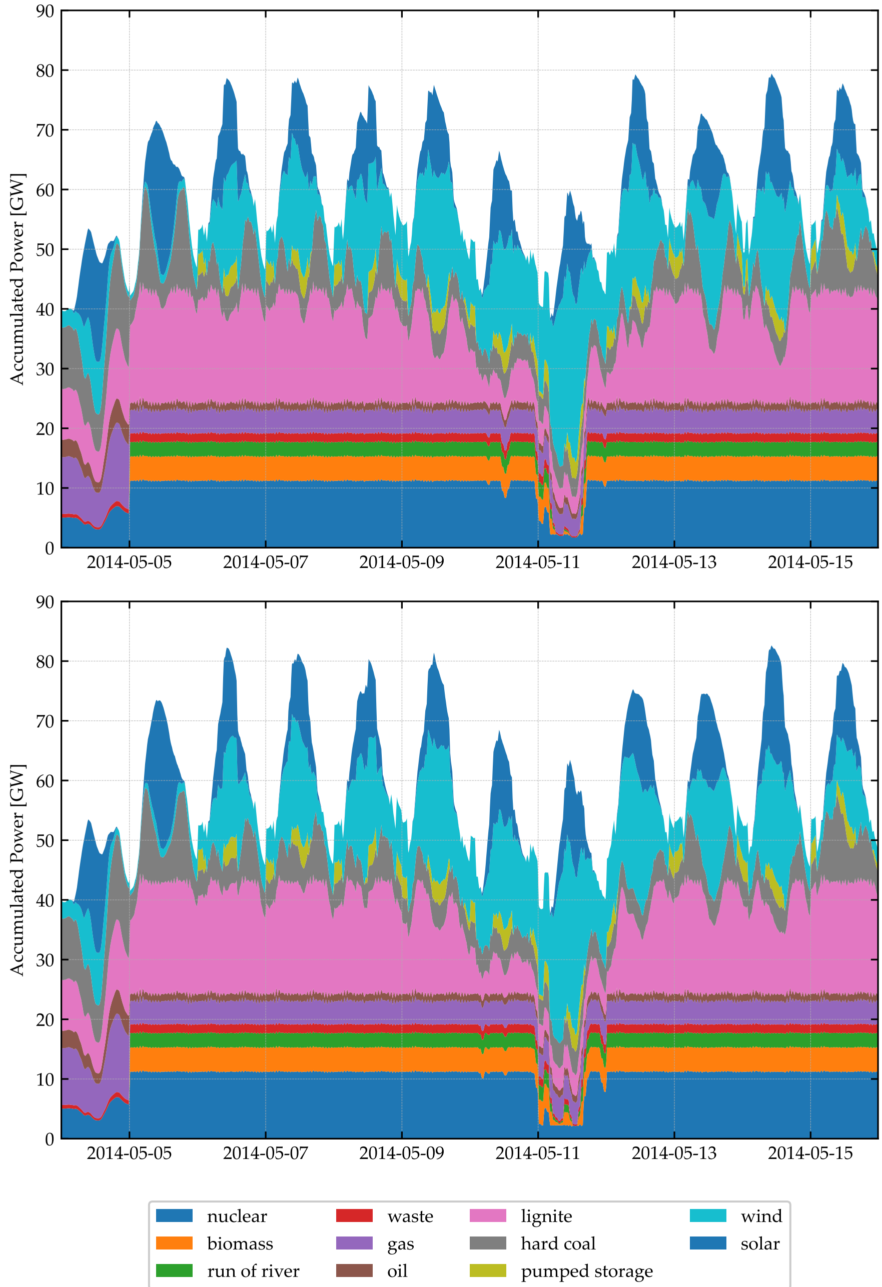


Figure 67: Accumulated power production per energy source in the scenarios `3Lop` (upper graph) and `ManyLop` (lower graph). The first two days show the "ramp-up" of the simulation. For the first day, no planning of generation via the day-ahead market took place. The total electricity demand is therefore, unrealistically, met by activating primary and secondary control energy. On the second day, the pumped storage power plants are still missing because, at the GCT of the day-ahead market at the first day, they did not have the forecast yet they needed to participate in the market.

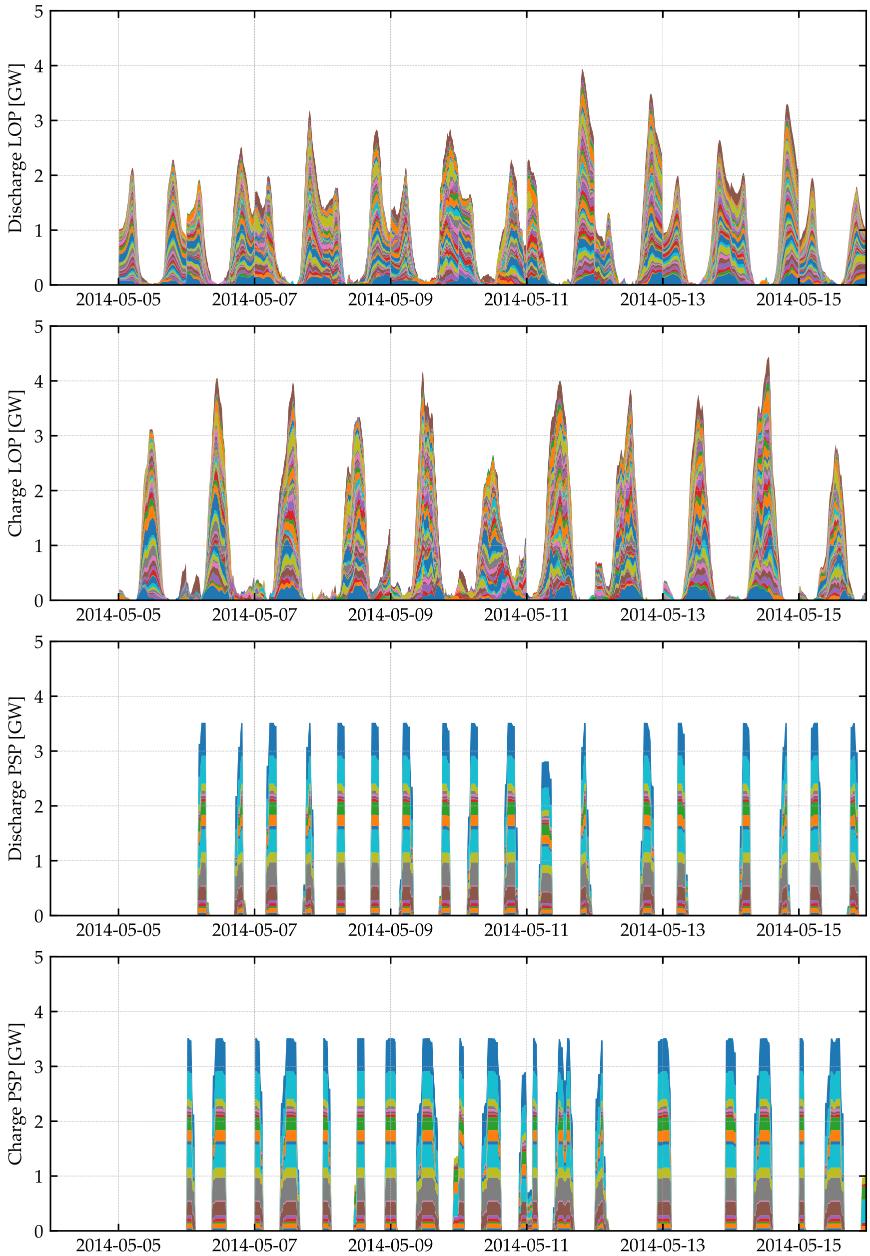


Figure 68: Accumulated behavior of all LOCAL OPTIMIZERS (upper two graphs) and pumped storage power plants (lower two graphs) in the `ManyLop` scenario. Each color shows the behavior of one of the total of 46 LOCAL OPTIMIZERS or 20 pumped storage power plants.

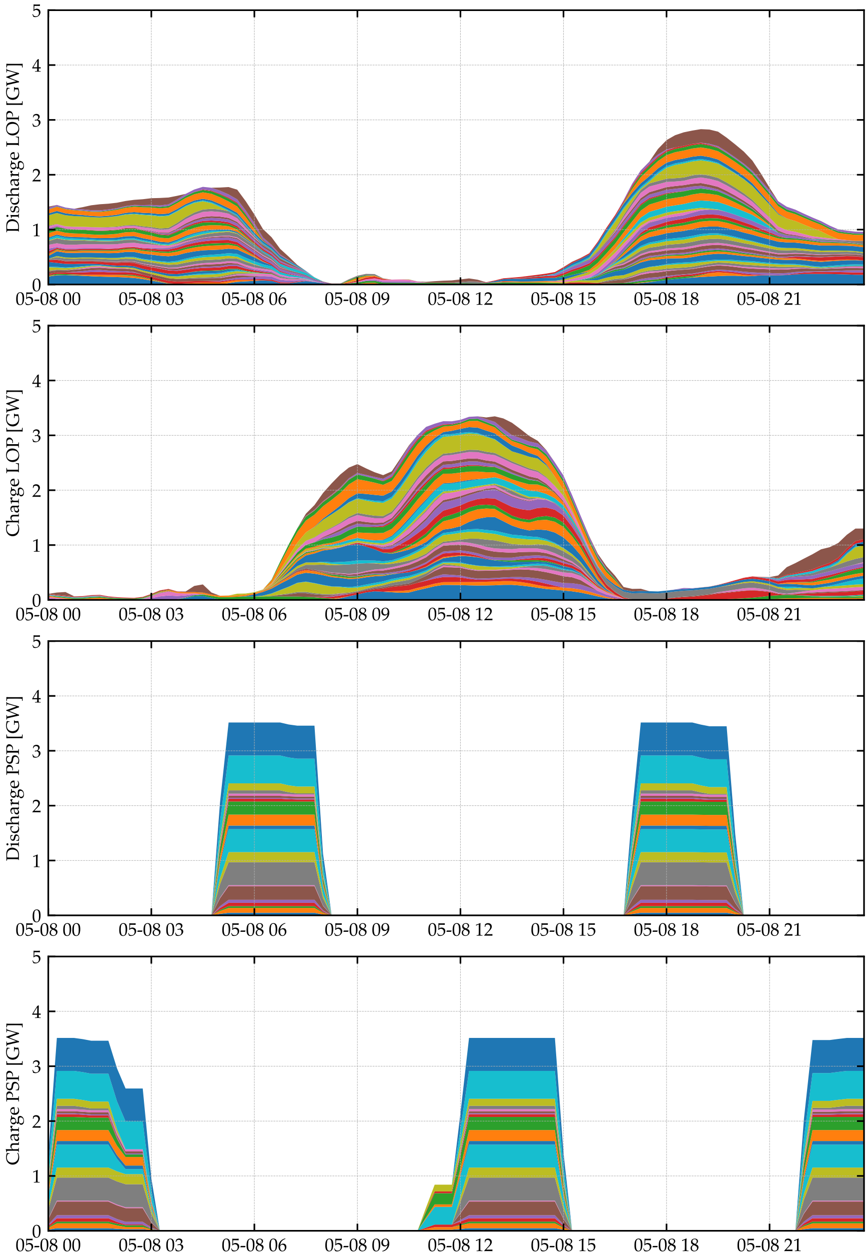


Figure 69: Same as Figure 68 but zooming into May 8.

5.6.2 *Market*

GENERAL OBSERVATIONS The course of the market price, which is shown in Figure 70, qualitatively corresponds to the known pattern of market prices under the influence of VRE-based production units, i.a. described in Sensfuß et al. [168]. In particular the histograms of the market prices shown in Figure 71 make it clear that some price ranges rarely appear; concretely, this is the price range between 5.00 and 12.50 €$MW^{-1}h^{-1}$ and the price range between 20.00 and 27.50 €$MW^{-1}h^{-1}$. This pricing is caused by the characteristics of the supply and demand curves shown in Figures 72 and 73. The intersection of two corresponding curves denotes the resulting market price given in Figure 70. The demand (load) is largely price-inelastic and, therefore, the demand curve appears as a vertical line in the diagram, which shifts horizontally with the load. The supply curve corresponds to the MERIT ORDER and thus includes jumps due to the different ENERGY CARRIERS and their specific costs. As VRE-based generation units do not have marginal costs, they shift the supply curve on the horizontal axis.

The vertical sections in the supply and demand curves lead to the observed price jumps by minimal changes in load or availability of VRE-based production units. It should be noted though that the model slightly exaggerates in comparison to reality. In fact, the price jumps are present but more diffused, which leads to a smoother distribution of market prices.

INFLUENCE OF LOCAL OPTIMIZERS As expected, the LOCAL OPTIMIZERS shift both the supply and demand curves. When they purchase electricity, they have placed a "buy-at-any-price" order on the market, thus shifting the demand curve horizontally to the right. This can be observed in Figures 72 and 73. If they are feeding power into the grid, the LOCAL OPTIMIZERS have set a "sell-at-any-price" order. This shifts the supply curve to the left. Comparing Figure 68 with Figure 67 shows that the total output of the LOCAL OPTIMIZERS is relatively small compared to the total load, but nevertheless, due to the described characteristics of the supply/demand curve, they can have a strong influence on the prices that are formed, especially at periods of low load and a high level of VRE-based production. This can be observed in Figure 73, see hour of day 12 and 13.

As given in Table 13, the MARKET VALUE (MV) of electricity from wind and PV plants is significant higher in the scenario `ManyLop` than in the scenario `3Lop`. The MV is the average price that can be achieved for electricity from a specific type of source, and typically decreases with higher penetration rates [169]. Higher market values for VRE-based electricity are generally to be viewed positively; especially since they reduce the necessary subsidies for renewable energies.

	MARKET VALUE (MV)		
	3Lop €$MW^{-1}h^{-1}$	ManyLop €$MW^{-1}h^{-1}$	percent variation [%]
solar	16.14	19.75	22.39
wind	14.83	16.86	13.68
oil	20.26	20.45	0.96
gas	20.26	20.45	0.96
nuclear	21.82	21.69	−0.60
waste	21.43	21.13	−1.41
run of river	21.75	21.31	−2.02
biomass	21.75	21.31	−2.02
lignite	23.87	23.63	−1.00
hard coal	24.09	23.57	−2.19

Table 13: MV of the different energy sources.

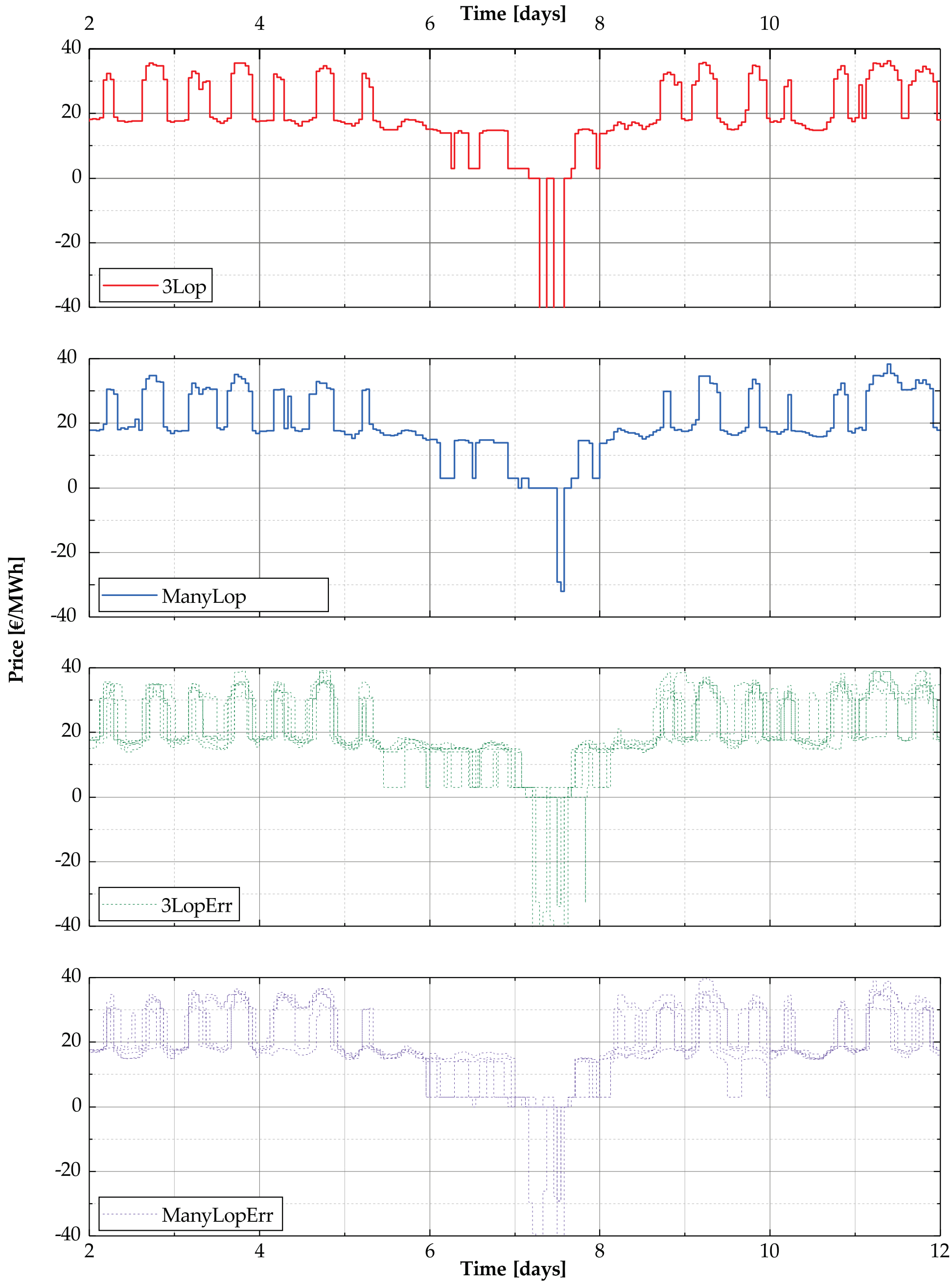


Figure 70: Day-ahead market prices in the different scenarios. For the scenarios with prediction errors, the results of 5 simulation runs are shown.

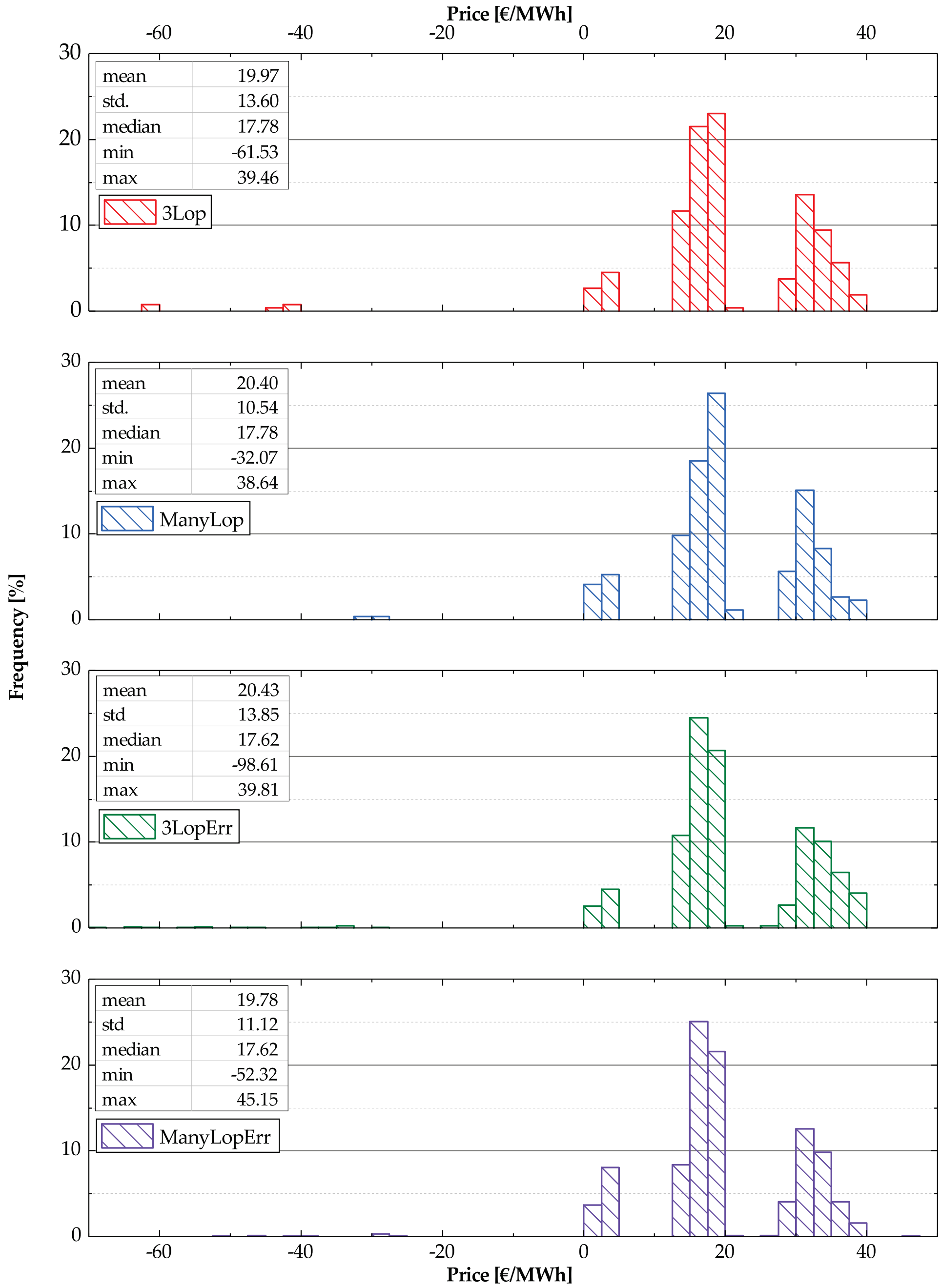


Figure 71: Distribution of day-ahead market prices of all days in the different scenarios.

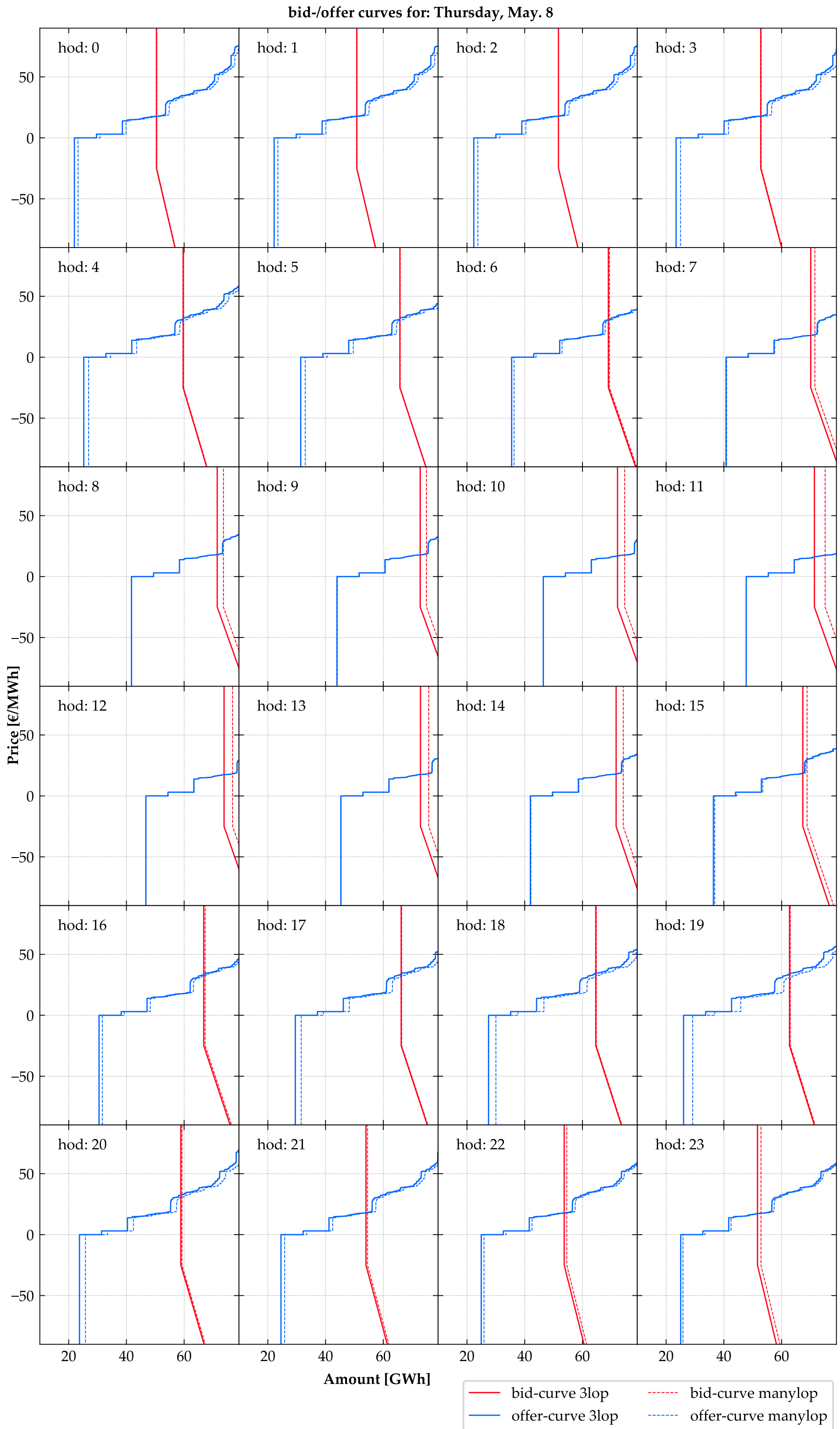


Figure 72: Bid and offer curves at the day-ahead market for each individual hour of the day (hod) on Thursday, May 8, 2014. The curves show the scenarios `3Lop` and `ManyLop`.

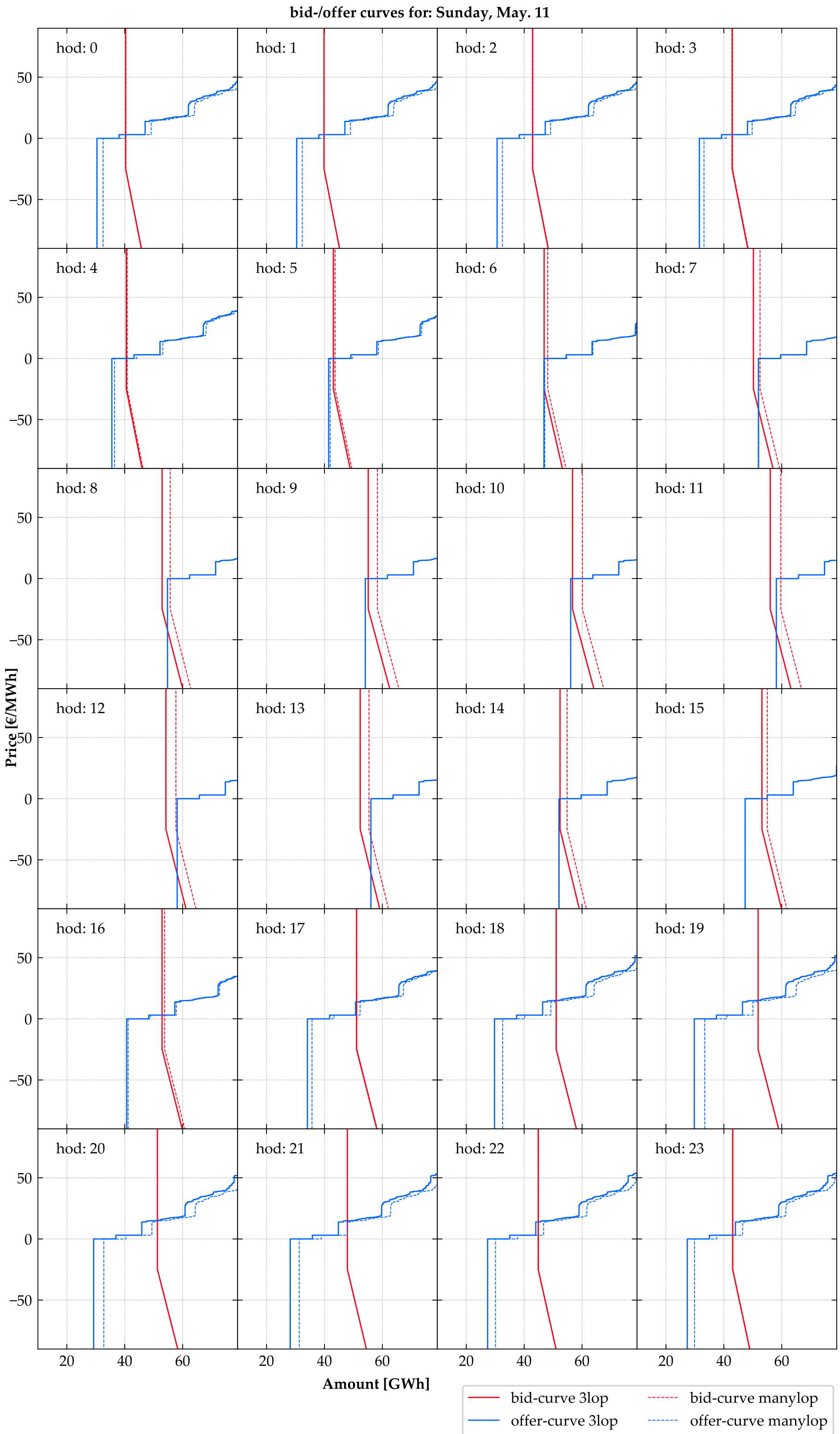


Figure 73: Bid and offer curves at the day-ahead market for each individual hour of the day (hod) on Sunday, May 11, 2014. The curves show the scenarios `3Lop` and `ManyLop`.

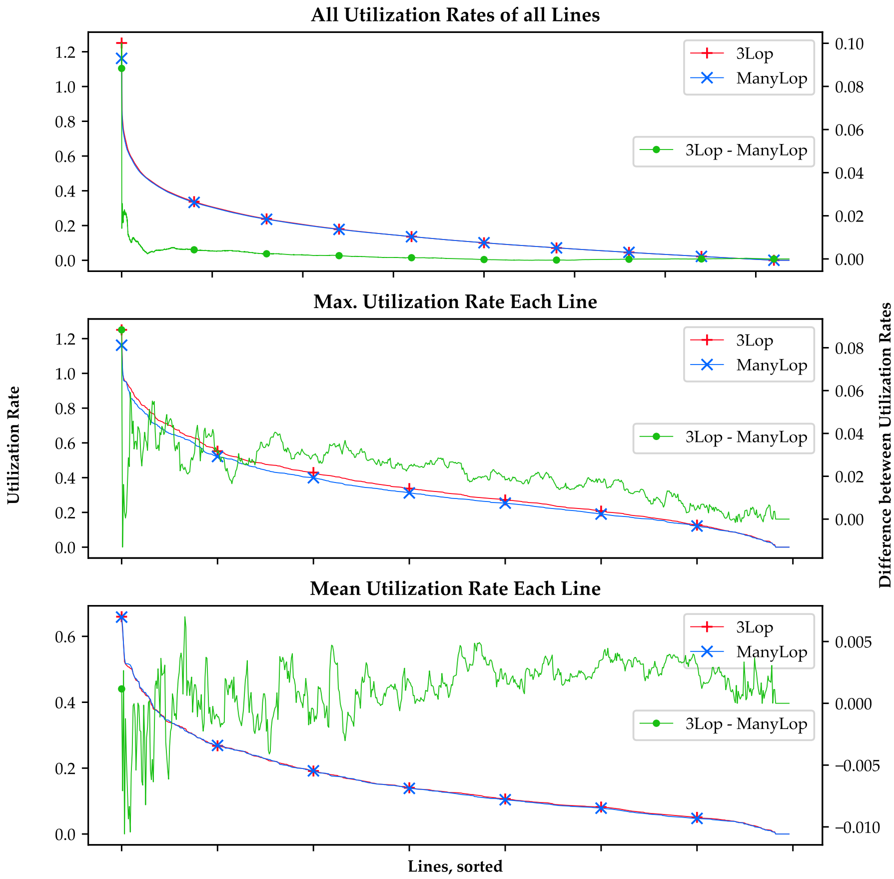


Figure 74: Details about the network load in the scenarios `3Lop` and `ManyLop`: Utilization rate per line and scenario sorted by load. The green line shows the difference between the two scenarios. The topmost graph uses all lines and all time slices. The graph in the center uses only the time slice with the maximum load from the individual lines.The lower graph uses the average value per line.

5.6.3 *Grid*

GENERAL OBSERVATIONS Figures 76, 74 and 75 show an overview of the load on the transport grid during the period under review. Some interesting findings can be discovered.

On average, the load on the transport network is relatively low. However, there are some periods of time when individual lines reach their load limit. As can be seen in Figure 76 (bottom), these lines are located around the border between former East and West Germany and in north-south direction in Northern Germany.

The first day of the observation period shows a significantly lower load on the power grid than all other days. This can be explained by the special situation that no planning in the sense of the BMP took place for

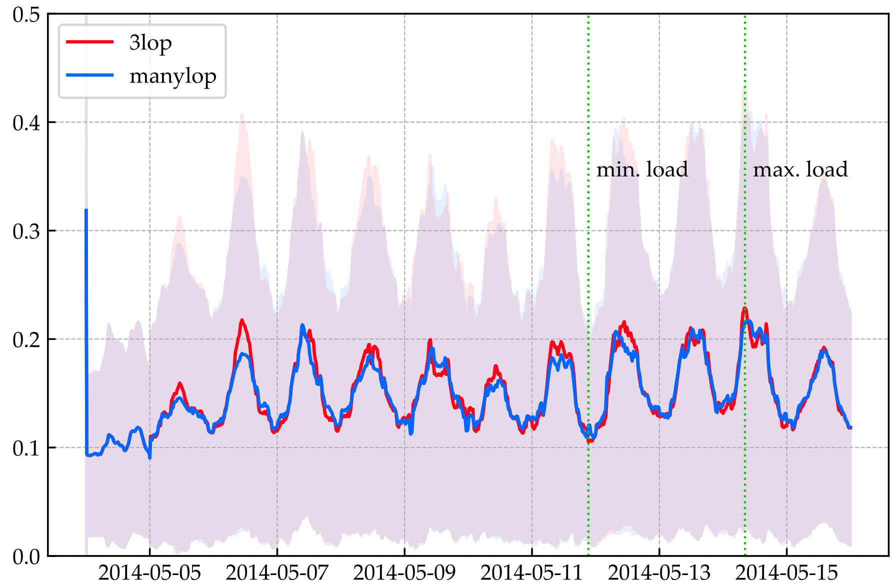


Figure 75: Details about the network load in the scenarios `3Lop` and `ManyLop`: Usage rate of the individual lines over time. The red and blue lines show the average values of all lines per time slice in the two scenarios `3Lop` and `ManyLop` respectively. The distributions of the utilization rate of all lines is indicated by the lower and upper boundary of the standard deviation given in light red and blue respectively. Cf. Figure 76.

this day. Thus, no matching of supply and demand on the day-ahead market has taken place, and the load is *not* covered by those units that offer the cheapest electricity. The electricity demand is therefore completely covered by the balancing power[10], whose mechanism is outlined in Section 5.4.6 (p. 204 ff.), and is provided equally by all power plants. Since the power plants are historically built close to the location of consumption, the load on the grid is low. The low feed-in from wind power also has a positive effect on the grid load, as *high* feed-in from wind power typically leads to a high north-south load on the power grid. Figure 77 shows the distribution of the grid frequency and the used primary control power of the simulation. They are slightly skewed, and there are two local maxima instead of the expected maximum at 50.00 Hz respectively 0.00 GW. These distribution patterns are known from measurements of the grid frequency, cf. Deng et al. [170]. Reasons can be the

10 In reality, the electricity grid would collapse without planning, as the feed-in of balancing power per power plant is technically limited. This limitation is deliberately not implemented in MOCES, as otherwise the "start-up" of the simulation would be much more complex.

effects shown by Weissbach and Welfonder [57] and the parameters of the grid controller and the machines providing the power. Nevertheless, if MOCES is to show these effects not only qualitatively but also to quantify them exactly, the models have to be tuned. This is because of the limitations already mentioned in Section 5.6.1 (p. 214). Many parameters of the models have not been validated to a required extent.

INFLUENCE OF LOCAL OPTIMIZERS When considering the influence of the LOCAL OPTIMIZERS on the transmission network, it must be borne in mind that reducing the network load is not an objective of the LOCAL OPTIMIZERS and that the sum of all LOCAL OPTIMIZERS increases the peak load by approximately 5.00 GW, cf. Figure 67. It is therefore not surprising that the influence on the high-level network situation, as shown in Figure 76 when comparing left to right, is rather small. However, if we look into the utilization rates of the individual lines in more detail, interesting differences can be found.

As Figures 74 and 75 show, the utilization of the network is generally only slightly reduced in the `ManyLop` scenario when comparing it to `3Lop`. However, a comparison of Figure 75 with Figure 67 shows that the increase of the load at noon in the scenario `ManyLop` is accompanied by a slight decrease of the network load.

A clear reduction effect exists for the points in time with high utilization rates. The reduction is up to 10.00 %, as can be seen in the graph in the center of Figure 74. The average load per line does not change significantly between the two scenarios, as the lower graph of Figure 74 shows.

A significant influence on the stability of the grid is exerted by the LOCAL OPTIMIZERS if the grid frequency and the utilized control power are taken as relevant measure. The histograms of the corresponding quantities are shown in Figure 77. Compared with the scenario `3Lop`, the standard deviation σ of the grid frequency is reduced from 15.20 m Hz to 12.90 m Hz in the scenario `ManyLop` (−15.00 %). The extreme values also show a slight improvement. The standard deviations of the used primary and secondary control power are reduced from 229.00 MW to 194.00 MW (−15.30 %) and from 940.00 MW to 722.00 MW (−23.20 %) respectively.

The sum of the positive secondary control power used in the considered time-span was reduced from 83.00 GW h to 71.00 GW h, that of the negative control power from 97.00 GW h to 71.00 GW h.

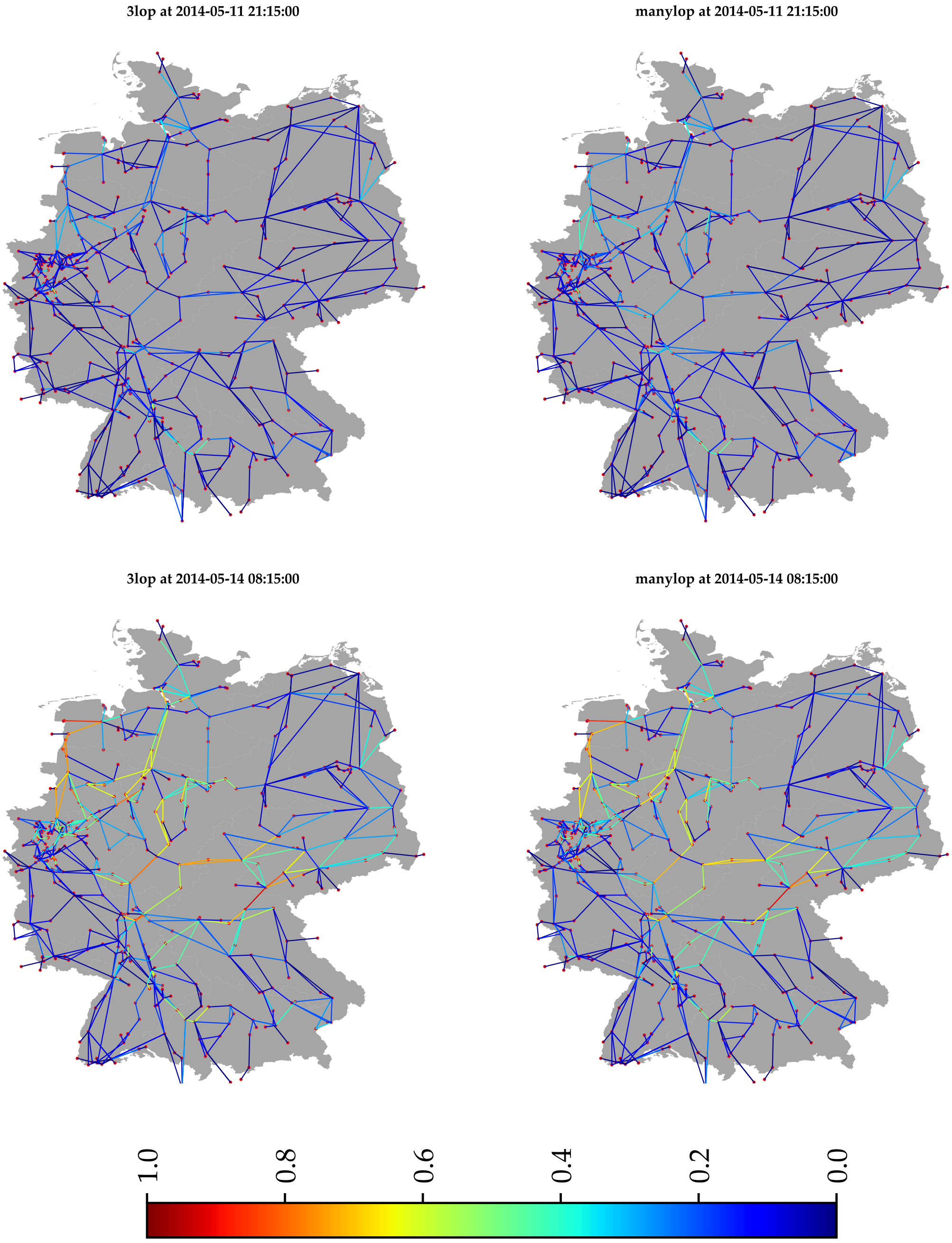


Figure 76: Network load in the different scenarios for the points in time with minimum (top) and maximum (bottom) network load. The load is given by the utilization rate of the line. See also Figures 74, 75.

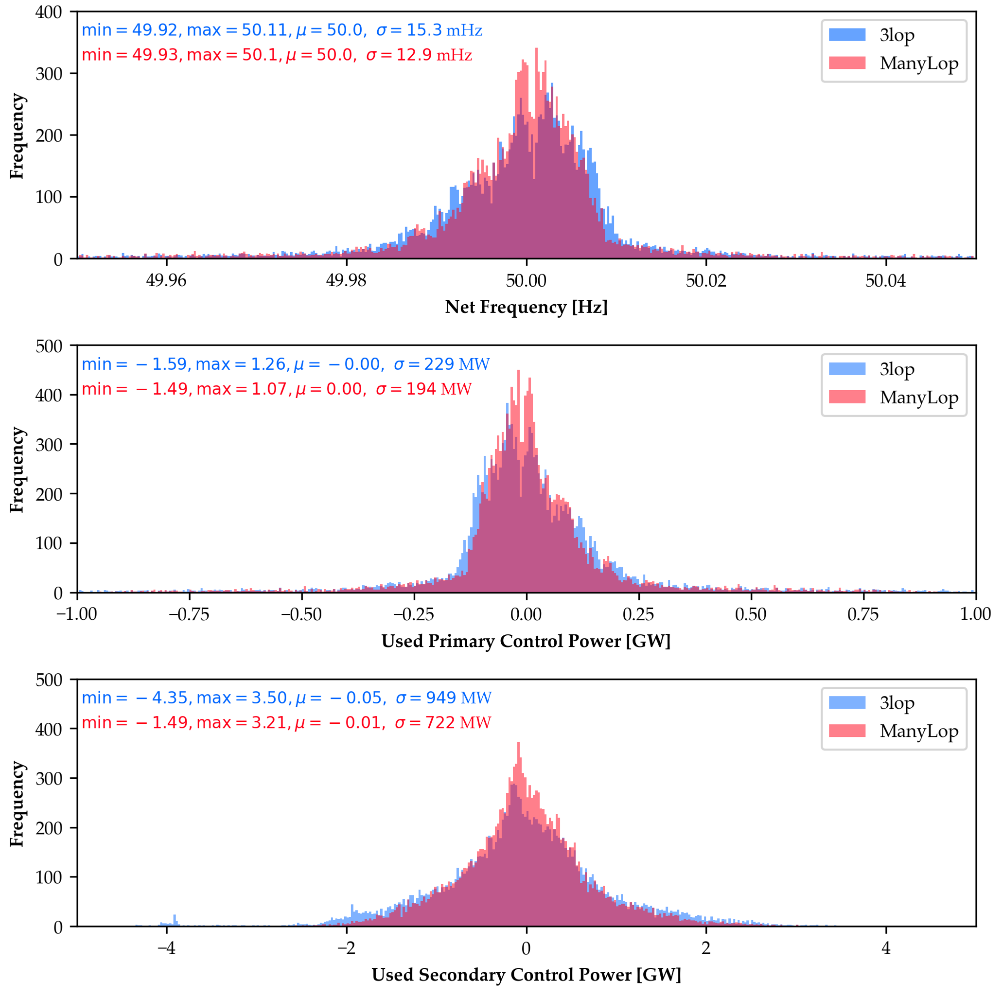


Figure 77: Histograms of grid stability metrics. To calculate the metrics, the period shown in Figure 75 without the first 24 hours was used.

Node	Scenario	Solar [k€]	Wind [k€]	Load [k€]	LOP [k€]	Sum [k€]	Perf. Gain [%]
	Baseline	82.53	647.94	−228.76	0.00	501.71	-
3	3Lop	82.53	647.94	−228.76	210.72	712.43	42.00
	ManyLop	103.88	783.76	−233.78	64.92	718.78	43.30
	Baseline	57.12	30.22	−960.96	0.00	−873.62	-
195	3Lop	57.12	30.22	−960.96	136.41	−737.21	15.60
	ManyLop	116.58	42.03	−981.48	79.61	−743.26	14.90
	Baseline	45.42	5.24	−465.04	0.00	−414.38	-
424	3Lop	45.42	5.24	−465.04	57.48	−356.90	13.90
	ManyLop	73.24	5.51	−475.20	35.96	−360.49	13.00

Table 14: Profit of selected nodes and their entities in k Euro during the given time span of eleven days. The performance gain is given as increase of income compared to the baseline.

5.6.4 *MENT and Node-Specific Performance*

Finally, we would like to give an insight into the behavior of the individual nodes. For this purpose, we consider the profit and other related metrics of the corresponding MENTS which are assigned to the respective nodes but ignore the MENTS that represent conventional power plants to focus on the VRE-based part of the POWER SYSTEM, cf. Figure 63. The profit of a MENT is the sum of all financial payments between this MENT and other MENTS. Primarily, these are the costs or revenues of the trading transactions on the day-ahead market, but also the payments within the settlement phase due to imbalances.

GENERAL PERFORMANCE OF LOCAL OPTIMIZERS In the considered time span, all LOCAL OPTIMIZERS have a positive profit in the scenario `3Lop` as well as in the scenario `ManyLop`, as given by the purple bars in Figure 78. This is in some way surprising, since it is not the goal of the LOCAL OPTIMIZERS to optimize the profit. A detailed evaluation of the three nodes which also feature a LOCAL OPTIMIZER in the scenario `3Lop` is given in Table 14.

The table shows that the profits of the individual nodes are significantly increased compared to a baseline scenario that is defined as the sum of the incomes without LOCAL OPTIMIZER in the scenario `3Lop`. Whether, in that sense, a MENTS of a node performs better in scenario `3Lop` or in `ManyLop` differs depending on how the individual MENTS perform in the different scenarios.

INFLUENCE OF LOCAL OPTIMIZERS For nodes 3, 195 and 424, it can be clearly seen from Table 14 that the higher number of LOCAL OPTIMIZ-

ERS in the ManyLop scenario significantly reduces the profit of the individual LOCAL OPTIMIZER, by between 37.00 % and 69.00 % compared to the 3Lop scenario. However, the yield of wind and PV is increased. This can be explained by the reduction of the daily price spread between times with high and low prices on the day-ahead market, cf. Figure 71, and the increase of the MV, cf. Table 13. The LOCAL OPTIMIZERS themselves are responsible for this reduction, as they shift the supply and demand curves as shown in Figure 72. They increase the supply at times of rather high prices and strengthen the demand at times of quite low prices leading to the observed smoothing. This also has a significant effect on the profits of pumped storage power plants. The changes in their profits are summarized in Figure 79. Even without closer examination, these simulation results thus show how difficult –or even impossible– it is to operate a storage system, like a LOCAL OPTIMIZER or pumped storage system, economically in an energy-only market.

In addition, the simulation results also show how relatively small changes to the overall system lead to significant effects on individual parts of the system. At the same time, these results must be interpreted with care. The significantly poorer performance of the storage systems in the scenario ManyLop compared to the scenario 3Lop could also –at least partially– be caused by a poor forecast of the expected market prices.

SELF-CONSUMPTION RATE AND SELF-SUFFICIENCY The LOCAL OPTIMIZER has a clear influence on the self-consumption rates and the self-sufficiency of the individual nodes. The SELF-CONSUMPTION RATE (SCR) indicates the percentage of electricity generated locally from renewable energies and used locally. The SELF-SUFFICIENCY RATE (SSR) defines the percentage of energy demand covered by local energy sources. In the scenario under consideration, the producers and consumers assigned to the node are considered local. Based on [171], we use the following definitions:

$$SCR = \frac{E_{\text{dir.use}} + E_{\text{stor,in}}}{E_{\text{pv,wind}}} \tag{5.25}$$

$$SSR = \frac{E_{\text{dir.use}} + E_{\text{stor,out}}}{E_{\text{load}}} \tag{5.26}$$

with:

SCR Self-Consumption Rate,
SSR Self-Sufficiency Rate,
$E_{\text{dir.use}}$ directly used energy generated by PV or wind plant,
$E_{\text{stor,in}}$ energy used to charge storage system,
$E_{\text{stor,out}}$ energy provided by storage system,

$E_{\text{pv,wind}}$ total energy generated by PV or wind plant,
E_{load} energy consumed.

Using the sign convention of Figure 63 we can rewrite the formulas:

$$SCR = \frac{\int \min\left(\left|p^{\text{pv}}(t) + p^{\text{wind}}(t)\right|, p^{\text{load}}(t) + p^{\text{stor.,in}}(t)\right) dt}{\int \left|p^{\text{pv}}(t) + p^{\text{wind}}(t)\right|}, \quad (5.27)$$

$$SSR = \frac{\int \min\left(\left|p^{\text{pv}}(t) + p^{\text{wind}}(t)\right| + p^{\text{stor.,out}}(t), p^{\text{load}}(t)\right) dt}{\int p^{\text{load}}(t)}. \quad (5.28)$$

As the equations suggest, the two metrics are typically opposite and depend on the ratio between renewable generation and load in a system without influence of a storage system. If the average load is significantly larger than the VRE-based generation, the SCR is typically high, and the SSR low. If the average load is significantly lower than the generation, this leads to a high SSR but a low SCR.

Table 15 shows the two metrics for all nodes with a LOCAL OPTIMIZER. Both SSR and SCR typically increase by double-digit percentages through the LOCAL OPTIMIZERS. Exceptions are only those nodes that already have a high value close to 100.00 % in the base scenario such as node 115 (high SCR value) or 3 (high SSR value).

5.7 SUMMARY

In this last chapter it was shown how MOCES, the simulation framework developed in this thesis, can be used to gain a more detailed insight into the behavior of a complex energy system.

We demonstrated that MOCES can simulate the power system of Germany with relatively high geographic and temporal resolution. In total, more than 2000 individually modeled MENTS interact with each other in the model. For each of these MENTS, the simulation results provide, among other things, the individual load and feed-in behaviors, the placement of bids and offers on the electricity market, and the interactions within the BMP. The simulation results also include the cash flows for revenues on the market and the payments for balancing energy within the settlement phase of the BMP.

For none of these MENTS, the load or feed-in behavior is predefined. Their behavior is the result of the given boundary conditions, especially the weather, the individual goals and the interaction with other MENTS. Even the storage systems follow an objective and adjust their behavior based on the information they receive during simulation. For this purpose, they solve an OPTIMIZATION MODEL at simulation runtime.

This modeling approach makes the developed MOCES-based model suitable to investigate how individual aspects of the system change when new actors with their individual objectives are integrated into the system. As a use-case presented in this chapter, LOCAL OPTIMIZERS were added to the model of the German power system. Those are storage units that aim to increase the local consumption of energy. We have shown how it can be investigated how these additional actors influence the different aspects of the power system such as the grid stability, the SCR of the individual nodes and the MARKET VALUE of different energy sources.

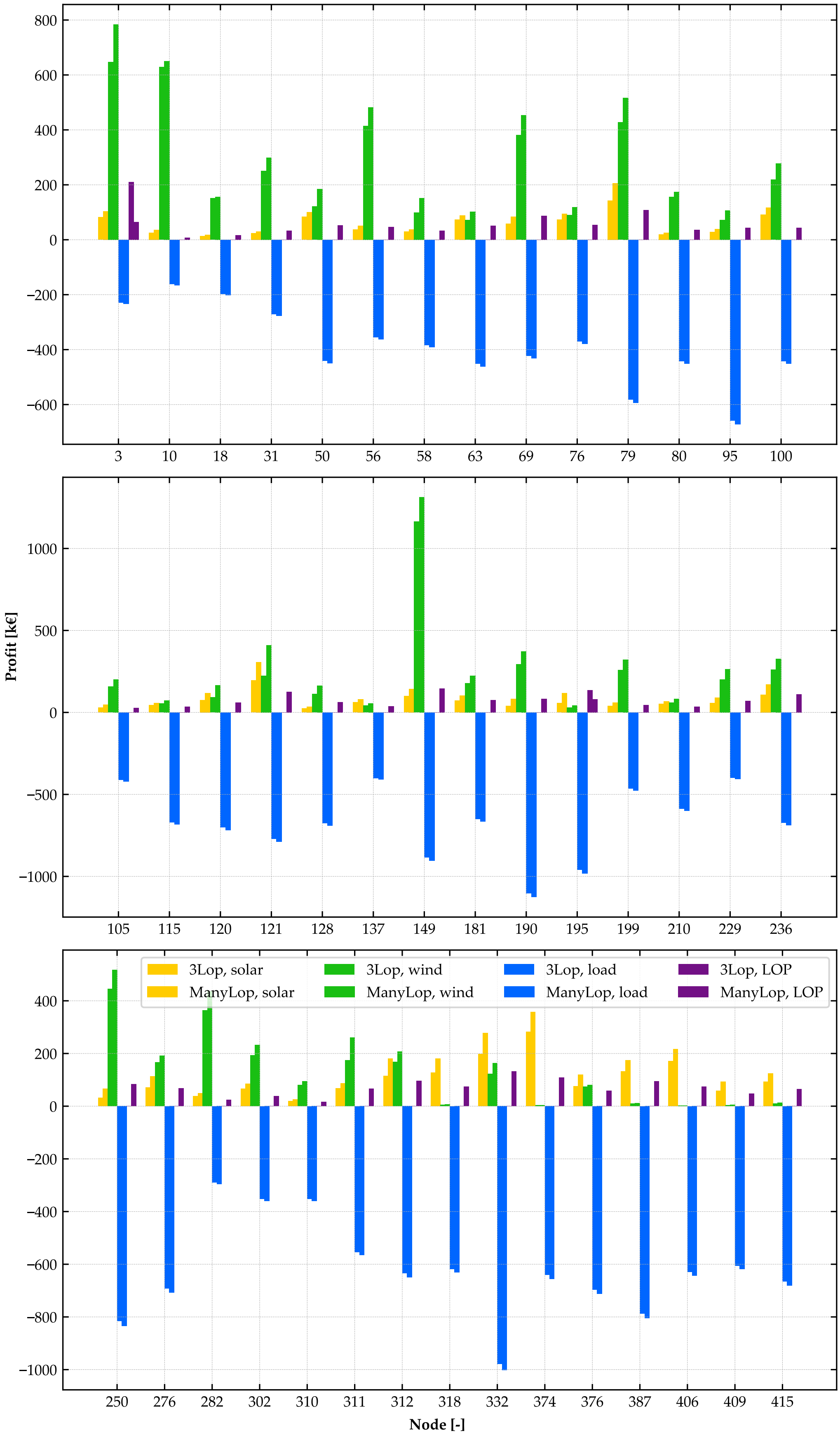


Figure 78: Profit of all nodes with a LOCAL OPTIMIZER.

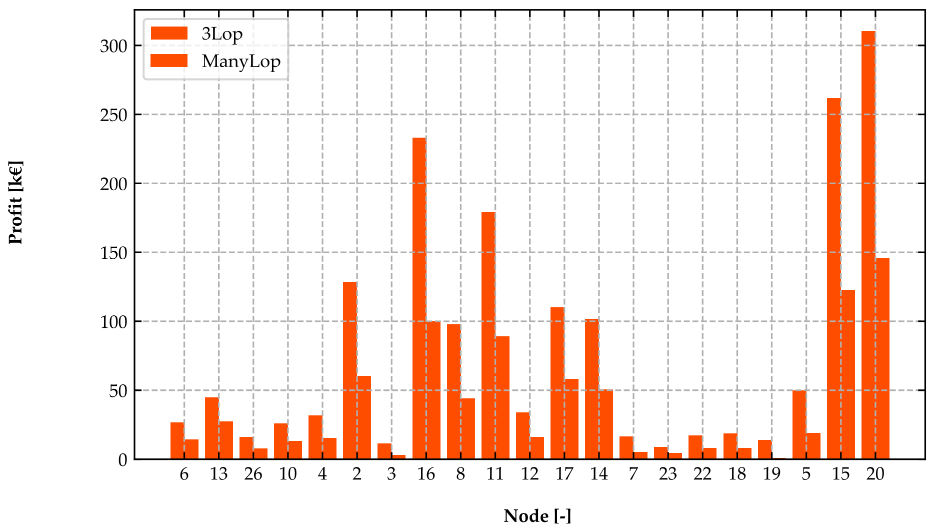


Figure 79: Profit of all pumped storage plants.

node	SCR-base	SCR	perf. gain	SSR-base	SSR	perf. gain
3.00	19.25	45.95	26.69	91.27	100.00	8.73
10.00	20.85	30.94	10.09	95.35	100.00	4.65
18.00	53.88	74.12	20.24	56.70	75.49	18.78
31.00	54.33	73.45	19.13	77.09	89.45	12.37
50.00	67.20	85.81	18.61	61.91	82.06	20.15
56.00	47.62	73.55	25.93	75.09	94.57	19.48
58.00	72.40	85.59	13.20	53.35	68.68	15.33
63.00	88.86	96.42	7.56	50.54	64.45	13.91
69.00	53.86	80.91	27.05	73.97	92.63	18.66
76.00	78.45	96.78	18.33	58.04	78.09	20.05
79.00	54.93	82.23	27.31	75.70	97.04	21.34
80.00	93.40	98.09	4.69	53.37	63.55	10.19
95.00	98.47	100.00	1.53	27.63	34.70	7.07
100.00	64.56	84.48	19.92	70.95	90.16	19.21
101.00	42.78	65.40	22.62	83.56	96.87	13.30
105.00	79.52	93.67	14.14	56.12	71.92	15.81
115.00	99.97	100.00	0.03	25.14	30.98	5.84
120.00	88.11	96.87	8.77	43.37	57.79	14.42
121.00	57.11	86.80	29.69	64.35	91.78	27.43
128.00	86.41	97.20	10.79	37.70	48.21	10.51
137.00	94.45	100.00	5.55	43.09	55.13	12.03
149.00	46.19	61.89	15.70	80.74	92.58	11.84
181.00	87.77	98.43	10.66	50.64	64.80	14.15
190.00	92.77	98.54	5.78	51.36	62.05	10.70
195.00	99.80	100.00	0.20	26.54	35.17	8.63
199.00	66.27	84.86	18.60	64.26	82.25	17.99
210.00	98.78	100.00	1.22	32.84	41.43	8.59
229.00	58.23	80.08	21.85	69.07	89.18	20.11
236.00	70.39	89.45	19.05	66.67	85.55	18.87
246.00	75.23	84.71	9.48	64.51	73.36	8.84
250.00	77.67	88.29	10.62	68.22	80.11	11.89
276.00	95.93	98.69	2.76	54.13	65.52	11.39
282.00	37.60	56.39	18.79	82.95	94.72	11.78
302.00	62.52	80.29	17.77	74.53	89.92	15.39
310.00	93.61	100.00	6.39	42.19	51.21	9.01
311.00	66.89	92.50	25.61	50.81	75.49	24.68
312.00	72.37	93.60	21.23	61.81	82.55	20.74
318.00	91.61	100.00	8.39	39.18	55.99	16.80
332.00	82.92	99.09	16.17	49.20	70.00	20.80
374.00	54.80	92.37	37.56	50.30	84.29	33.98
376.00	98.54	100.00	1.46	35.81	45.30	9.49
387.00	93.50	100.00	6.50	37.22	51.97	14.75
406.00	76.08	98.81	22.74	43.04	65.79	22.75
409.00	97.52	100.00	2.48	25.10	34.75	9.65
415.00	95.57	100.00	4.43	32.63	44.89	12.26
424.00	97.44	100.00	2.56	25.65	35.21	9.56
min.	19.25	30.94	0.03	25.10	30.98	4.65
max.	99.97	100.00	37.56	95.35	100.00	33.98
mean	74.05	87.96	13.91	55.52	70.38	14.87

Table 15: Change of SCR and SSR in the ManyLop scenario for all nodes with LOCAL OPTIMIZER. All values are given in percent. The performance gain is given as the difference between a base scenario and the ManyLop scenario. For the values in the base scenario, the influence of the LOCAL OPTIMIZERS was deducted, which is possible as the load and VRE-based generation is independent of the system behavior.

6 CONCLUSION

This thesis has been motivated by the challenge to not only consider the different aspects of the changing energy system individually, but to model and simulate them in a holistic manner to enable studying the interactions between the individual aspects. The present work and the developed framework MOCES, short for MODELING OF COMPLEX ENERGY SYSTEMS, provide a solution for this. Focusing on the POWER SYSTEM, the five key points of this thesis, each building up on the previous one, contribute to this approach.

To start, we have provided a *holistic description* of the energy and electricity system with a focus on all those aspects that have been and will be undergoing change. The elaborated description shows that, for a holistic modeling, three building blocks and their interactions have to be represented: (i) the technical units, including the dependency on wind and sun, and their coupling via the power grid; (ii) the different actors that directly or indirectly influence the technical units with their behavior and individual, typically financial goals; and (iii) the regulatory framework that defines processes and the rights and obligations of the actors.

After discussing existing modeling and simulation approaches, the thesis describes a novel *method for modeling and simulation of complex energy systems*. The core of the method is the introduction of three modeling layers, and the definition of the MOCES ENTITY (MENT). While the physical layer allows the modeling of the technical units, the business process layer allows to describe the interaction of the actuators within the regulatory framework. The information layer represents the environment that cannot be influenced but that has an effect on the behavior of the entire system, e.g. the weather. A MENT describes an entity that spans over the physical and the business process layers and thus achieves their coupling. We have also identified that for both parts of the MENT, and thus for the respective modeling layers, appropriate and structure-preserving modeling and simulation approaches exist: component-oriented and equation-based modeling and simulation for the physical layer, and agent-based modeling and simulation for the business process layer.

Next, the thesis describes our *implementation* of the aforementioned method using Modelica. By relying on Modelica, existing tool chains can be used for the description and simulation of models. We have shown in detail how Modelica models can be extended by an agent-based simulation without violating the limitations of the MODELICA LAN-

GUAGE SPECIFICATION (MLS), and how this extension can be implemented efficiently when using the the required simulation time as key measure. On the basis of that, we have *developed the modeling framework* MOCES: a MODELICA library containing basic models for the description of power systems. In addition, we have demonstrated how models can be described in MOCES and how they can be used to investigate specific scenarios.

The final contribution of the thesis is the use of MOCES to *model and simulate and thus study the German electricity system*, under the influence of a high share of renewable energies, and storage entities that lead to an increase in the share of locally consumed energy.

In the course of this work some problems were identified and described. Two of these challenges should be mentioned here specifically, since their successful handling would have an immediate positive effect on MOCES, its use and applicability.

The tools used and the underlying concepts for processing MODELICA *models reached their limits* during the investigations. Models with higher modeling depth would have been impossible to process and simulate. Accordingly, more precise models for the storage system, the grid or conventional power plants could not be used, although they could have been formulated and integrated relatively easily. The MODELING LANGUAGE itself is not the problem here, but the processing of the models. Solution approaches were briefly described in Section 3.3.3 (p. 111 ff.), but could not be implemented for reasons of scope.

Unfortunately, the developed models could *not be validated* to the extent that would have been desirable and necessary to use the developed models for reliable statements about the simulated system. This shortcoming is due to the *poor data situation* (cf. Section 2.4.3, p. 77).

Our work, including the identified challenges, leads to a number of further research questions. They are described in the following as possible directions of future work.

Enabling the *processing of large* MODELICA *models* is a fundamental task that does not only concern MOCES, but also other applications based on MODELICA. One possible approach would be to create an alternative process chain for large, typically sparse, models, possibly under a conscious decision to drop the generality claim of the current tool chain. The result would be a specialized process chain that cannot process every MODELICA model anymore, but instead large models with certain characteristics efficiently. Linked to this is a probably necessary shift from the approach of monolithic simulation of the model to a distributed simulation. Without an in-depth analysis, the author's assumption is that the underlying mathematical questions have already been comprehensively addressed and that the challenge is to apply existing concepts to MODELICA.

Obviously, within MOCES, a lot of work could still be done. One category of future work addresses *further extensions to the framework*. Here, the most important aspects are (i) a better representation of dynamic relations between the MENTS in the BUSINESS PROCESS LAYER, and (ii) the possibility of the MENTS to influence the structure of the system with their decisions. Extension (i) would allow MENTS to decide about their relationship to other MENTS and how they interact with each other during the simulation. A simple example would be a consumer who decides to change his or her electricity supplier because the new supplier offers electricity from renewable energy sources exclusively. Extension (ii) is necessary in particular to reflect investment decisions, such as the increase or decrease of production capacity, and their influence on the overall system.

Furthermore, certain additional features would *improve the usability* of MOCES. For example, the modeling of the behavior of the MENTS in the business process layer is currently described following the basic idea of Petri nets, but it does not use a formal modeling language in the actual sense. Additionally, the handling of scenarios could also be improved, both with respect to the creation and parameterization of the models and their simulation results.

Last but foremost, this thesis enables further research work in the area of the further development of our energy system supported by modeling and simulation with the goal of a sustainable energy supply. Such work could be based on MOCES.

A APPENDIX

A.1 MODELICA HDAE REPRESENTATION

Strictly speaking, the HDAE form is out of scope of the MODELICA specification. Nevertheless, an exemplified definition is given in the appendix of the specification [120]. We provide this definition in a slightly modified form, taking into account remarks from Lundvall et al. [157].

Let $\mathbf{v} := [\dot{\mathbf{x}}\ \mathbf{x}\ \mathbf{y}\ t\ \mathbf{m}^-\ \mathbf{m}^+\ \mathbf{p}]^\mathsf{T}$ be a vector of MODELICA variables with:

- $\mathbf{x}(t)$ Vector of dynamic variables in the model that occur differentiated as the `der()` $(\frac{d\mathbf{x}}{t})$ operator is applied to them. Variables have to be of type `Real`.
- $\mathbf{y}(t)$ Vector of algebraic variables of type `Real` that neither appear differentiated nor are of type `discrete`.
- t time
- $\mathbf{m}(t_e)^{-,+}$ Vector of discrete-time variables which change their values only at events (t_e). These variables have to be of any of the following types: `Real`, `Boolean`, `Integer`. $\mathbf{m}(t_e)^-$ are the values of the variables before the event, $\mathbf{m}(t_e)^+$ are the corresponding values at and after the event.
- $\mathbf{p}$ Vector of variables which are declared as `parameter` or `constant`.

The set of equations, including algebraic, differential and discrete equations, can be formulated as follow:

$$\mathbf{c}(t_e) := \mathbf{f_c}\left(relation\left(\mathbf{v}\right)\right), \tag{A.1a}$$

$$\mathbf{m}(t_e) := \mathbf{f_m}\left(\mathbf{v}, \mathbf{c}\right), \tag{A.1b}$$

$$\mathbf{0} = \mathbf{f_x}\left(\mathbf{v}, \mathbf{c}\right), \tag{A.1c}$$

where:

- $\mathbf{c}(t_e)$ Vector of type `Boolean` holding all conditions extracted from `if` and `when` statements evaluated at t_e.
- $\mathbf{m}(t_e)$ $[\mathbf{m}(t_e)^-\ \mathbf{m}(t_e)^+]^\mathsf{T}$
- $\mathbf{f_m}(\ldots)$ Set of equations activated at t_e. The variables $\mathbf{m}(t_e)^+$ are unknown.
- $\mathbf{f_x}(\ldots)$ DAE in implicit form.

A system described by the set of Equations A.1a - A.1c can represent a variable-structure system as the discrete variables $\mathbf{m}(t_e)$ can be used to *disable* parts of the state vector. However, as Zimmer [172] shows, the MODELICA language is very limited in this respect.

The simulation of such a system is performed with the following five steps:

1. Solve the system (Eqns. A.1a - A.1c) to get the initial values for **v**.
2. Solve continuous system (Eqn. A.1c) using a numerical integration method. In this phase, the variables **c** and **m** are kept constant.
3. Monitor all relations of Eqn. A.1a and stop integrating to handle the detected event. How the event is detected depends on the the concrete implementation, but a common approach is to determine the exact point in time when the event is triggered by a root finding mechanism, which is provided within several solvers. Time events can be handled in a more efficient way, as their occurrence is known a priori.
4. At an event instance, the system (Eqns. A.1a - A.1c) is solved again.
5. After a successful handling of the event, the integration restarts (step 2).

At an event, the model equations are reinitialized according to the following algorithm:

```
known variables: x, t, p
unknown variables: dx/dt, y, m_minus, m_plus, c

loop
   solve Eqns. A.1a - A.1c for the unknowns, with m_minus fixed
   if m_plus == m_minus then break
   m_minus := m_plus
end loop
```

Listing A.1: Algorithm used to handle event iteration in MODELICA.

For further details, see [120, p. 272 ff.] or Lundvall et al. [157]. The latest MODELICA specification uses a slightly modified representation [173, p. 288 ff.] of HDAE.

A.2 MOTIVATING EXAMPLE

```
model BouncingBall2Vec
  // ball
  parameter Real a_2 = -9.81;   // gravity
```

```
  // variables for ball
  Real x_1(start = 0);          // ball position;
  Real x_2(start = 10);         // ball position;
  Real v_1(start = 0);          // ball v_1;
  Real v_2(start = 0);          // ball v_2;
  Real a_1;

  // boards
  parameter Integer n = 100;    // number of boards
  parameter Real b_w = 1;       // width of board
  parameter Real k = 0.01;      // dynamics of board
  // left border of active area of boards
  parameter Real[n] b_l = (0.1:10:(n-1)*10+0.1);
  // left border of active area of boards
  parameter Real[n] b_r = (10.1:10:(n-1)*10+10.1);
  // position of boards
  Real x_1_b[n](start = (0:10:(n-1)*10));
  // velocity of boards
  Real v_1_b[n](each start=0);
  Real x_1_set[n];        // set point for board

  // variables used by conditions
  Real next_event[n];
  Real cond[n](start = linspace(0, (n-1)*10,n));
  // variables used to give trajectory for board
  Real tra_a[n];
  Real tra_b[n];

  // helper variables to avoid external objects
  Real x_1_last_step[n](start = linspace(0, (n-1)*10,n));
  Real t_last_event[n];
  Integer agent_state_next[n](each start = 0);
  Integer agent_state[n](each start = 0);

equation
  // dynamics of ball
  v_1 = 0.5;
  der(x_1) = v_1;
  der(x_2) = v_2;
  der(v_1) = a_1;
  der(v_2) = a_2;
  // boards
  for i in 1:1:n loop
    // dynamics of board
    der(x_1_b[i]) = (1*x_1_set[i] - x_1_b[i])/k;
    der(x_1_b[i]) = v_1_b[i];
    // board trajectory
    x_1_set[i] = tra_a[i]*(time-t_last_event[i])^3
                 + tra_b[i]*(time-t_last_event[i])^2
                 + x_1_last_step[i];
    // trigger action
    when {(time > pre(next_event[i])), (x_1 > pre(cond[i]))} then
     agent_state[i] = pre(agent_state_next[i]);
```

```
      (next_event[i], agent_state_next[i], cond[i], tra_a[i], tra_b[i]) =
         ActionBoard(x_1, x_2, v_1, v_2, x_1_b[i],
                     time, pre(agent_state_next[i]), b_l[i], b_r[i]);
      x_1_last_step[i] = x_1_b[i];
      t_last_event[i] = time;
    end when;
    // ball hits a board
    when (x_2 <= 0) then
      // if any board is below the ball
      if (noEvent(abs(x_1-x_1_b[i]) < b_w/2)) then
         reinit(v_2,-1*pre(v_2));
      else
      // do nothing
      end if;
    // ball misses any boards
    elsewhen  (x_2 < -1) then
      ActionBall(x_2);
    end when;
  end for;
end BouncingBall2Vec;
```

Listing A.2: Full source code for the motivating example in Section 4.3.

```
model BouncingBall2VecAlt
  // ball
  parameter Real a_2 = -9.81; // gravity
  // variables for ball
  Real x_1(start=0);          // ball position;
  Real x_2(start=10);         // ball position;
  Real v_1(start=0);          // ball v_1;
  Real v_2(start=0);          // ball v_2;
  Real a_1;

  // boards
  parameter Integer n = 100;  // number of board;
  parameter Real b_w = 1;       // width of board
  parameter Real k = 0.01;    // dynamics of board
  // left border of active area of boards
  parameter Real[n] b_l = (0.1:10:(n-1)*10+0.1);
  // left border of active area of boards
  parameter Real[n] b_r = (10.1:10:(n-1)*10+10.1);
  // position of boards
  Real x_1_b[n](start=(0:10:(n-1)*10));
  // velocity of boards
  Real v_1_b[n](each start = 0);
  Real x_1_set[n];            // set point for board

equation
  // dynamics of ball
  v_1 = 0.5;
  der(x_1) = v_1;
  der(x_2) = v_2;
```

```
  der(v_1) = a_1;
  der(v_2) = a_2;

  for i in 1:1:n loop
    // dynamics of board
    der(x_1_b[i]) = (1*x_1_set[i] - x_1_b[i])/k;
    der(x_1_b[i]) = v_1_b[i];
    // when ball is out of board, start action
    when (sample(0,0.1)) then
      x_1_set[i] = ActionBoardAlt(x_1,
                                  x_1_b[i],
                                  b_l[i],
                                  b_r[i]);
    end when;
    // ball hits a board
    when (x_2 <= 0) then
      // if any board is below the ball
      if (noEvent(abs(x_1-x_1_b[i]) < b_w*2)) then
        reinit(v_2,-1*pre(v_2));
      else
        // do nothing
      end if;
    // ball misses any boards
    elsewhen  (x_2 < -1) then
      ActionBall(x_2);
    end when;
  end for;
end BouncingBall2VecAlt;
```

Listing A.3: Full source code of the alternative approach for the motivating example in Section 4.3.

A.3 MOCES ELMOD-DE ADAPTION

```
model Elmod_2014_biomass_and_run_of_river_psp_3LOP
  import NW = MOCES.RUN.ELMOD_DE.MODELS.NodeWrapperWithLOP;

  // some global variables used for analysis purposes
  MOCES.TYPES.ActivePower sum_solarplant;
  MOCES.TYPES.ActivePower sum_windplant;
  [...<further variables>...]

  // include scenerio aspect model representing
  // all hydro strorage power plants
  extends MOCES.RUN.ELMOD_DE.SCALE_TEST.all_PSP_plants;

  // extend base model including some base models
  extends MOCES.RUN.ELMOD_DE.Elmod_Base(convScaler(HardCoal=0.86, [...]),
    dAF_DayAheadFlex(init_DAF(
      dT_itp=3600,
```

```
      N_dt=24,
      min_price=-1.3888888888889e-07)),
      [...<further parameters>...]);

  // individual nodes
  NW nodeWrapper3(
    node_pos(pos_north=54.716000, pos_east=9.317000),
    enable_LOP=1,
    lop(init_MPP(GPS_pos(pos_north=54.716000, pos_east =9.317000)),
        init_MEP(unique_identifier="3_LOP"),
        init_LOP(eta_charge=0.980000,
                 eta_discharge=0.980000,
                 cap_storage=6087099164276.736300,
                 P_max=281810146.494293,
                 name_of_auction="DAF",
                 name_of_weather_service="SWF"))
    // model for load
    lor(area=3.834659e+09,
      shape_file="modelica://MOCES/Resources/[.../node_reg_n003.dxf",
      lor(init_MPP(GPS_pos(pos_north=9.317000, pos_east=54.716000)),
        init_LOR(time_series_load="entsoe_load_germany_w_adapted",
        time_series_percentile_of_max=
          "entsoe_load_germany_percentile_of_max_adapted",
        peak_share=0.000819,
        off_peak_share=0.000732,
        name_of_auction="DAF",
        name_of_load_forecast_service="SLF",
        name_of_LOP="3_LOP",
        send_schedule_to_LOP=true))),
    // model for solar power plants
    sop([...<further parameters>...])),
    // model for wind power plants
    wip([...<further parameters>...])),
    // biomass
    bdf([...<further parameters>...])]
    // conventional power plants
    ppm(
      init_MEP(unique_identifier={"Heizkraftwerk FL196",
                                  "Heizkraftwerk FL197"}),
      init_MPP(GPS_pos(pos_north={54.716,54.716},
                       pos_east={9.317,9.317}),
                       name_of_aggregator={"Heizkraftwerk FL196",
                                           "Heizkraftwerk FL197"}),
      init_PPM(type_of_plant={MOCES.RECORDS.TypeOfConvPowerPlant.HardCoal()},
               peak_power={27000000*convScaler.HardCoal,
                           29000000*convScaler.HardCoal},
      efficiency={0.405,0.39499},
      transport_costs={1.9671e-10,2.017e-10},
      name_of_auction={"DAF","DAF"}),
      part_of_control={0.00030563,0.00032827}),
    // hydro (run of river) plant
    ror([...<further parameters>...]);
  equation
```

```
  // setting up connections to the grid model
  elmod_system.p_at_nodes[427]  = nodeWrapper427.y + psp19.el_terminal;
  [...]
  elmod_system.p_at_nodes[436]  = 0;
  elmod_system.p_at_nodes[437]  = 0;
  elmod_system.p_at_nodes[438]  = nodeWrapper438.y + psp20.el_terminal;
  [...<further equations>...]

  sum_solarplant = (-1)*(0  + nodeWrapper3.sum_helper_solarplants.y
                            + [...<further equations>...] )
  sum_windplants = [...<further equations>...]
end Elmod_2014_biomass_and_run_of_river_psp_3LOP;
```

Listing A.4: Excerpt from the model code of scenario `3Lop`.

B

ACRONYMS AND GLOSSARY

ACRONYMS

AC Amplifying Current. 59, 258, 263
AMES Agent-based Modeling of Electricity Systems. 96
ANN Artificial Neural Network. 119, 179, 201, 202

BDEW Bundesverband der Energie- und Wasserwirtschaft (German Association of Energy and Water Industries). 10, 48, 52
BFGS Broyden–Fletcher–Goldfarb–Shanno. 196
BLT Block Lower Triangular. 108, 109
BMP Balance Management Process. xiii, xv, 5, 48, 51–55, 58, 60, 78, 116, 119, 153, 158, 159, 169, 174, 185, 192, 212, 224, 231
BMWi Bundesministerium für Wirtschaft und Energie (German Federal Ministry for Economic Affairs and Energy). 39
BNetzA Bundesnetzagentur (German Federal Network Agency). 31, 39, 56, 60, 61, 63, 78, 189
BRP Balance Responsible Party. 39, 47, 50, 51, 54, 56–60, 153, 158, 174, 212, 252, 253, 256, *Glossary:* Balance Responsible Party

CCGT Combined Cycle Gas Turbine. 94
CEN European Committee for Standardization. 46
CENELEC European Committee for Electrotechnical Standardization. 46
CHP Combined heat and power. 115
CIM Common Information Model. 116

DAE Differential Algebraic Equation. 101, 107–110, 112, 113, 145, 163, 168, 180, 241, 253, 254, 257, 262, *Glossary:* Differential Algebraic Equation
DASSL Differential Algebraic System Solver. 110, 112, 142, 168, 262, *Glossary:* Differential Algebraic System Solver
DC Direct Current. 203, 258
DER Decentralized Energy Resources. 254
DES Decentralized Energy System. 2, 9, 34, 43–46, 67
DIW Deutsches Institut für Wirtschaftsforschung (German Institute for Economic Research). 7, 187
DLR Deutsches Zentrum für Luft- und Raumfahrt (German Aerospace Center). 95, 123
DSO Distribution System Operator. 47, 72
DWD Deutscher Wetterdienst (German Meteorological Office). 7, 65, 156

ECST Energy Conversion and Storage Technologies. 1
EEG Erneuerbare-Energien-Gesetz (German Renewable Energy Sources Act). 39, 196
EEX European Energy Exchange. 61
ELMOD-DE Open Source Electricity Model for Germany. xvi, 6, 187–193, 195, 198, 199, 201–203, 207, 208
EMS Energy Management System. 43
ENTSO-E European Network of Transmission System Operators. 5, 7, 46, 53, 71, 94, 189
EnWG Energiewirtschaftsgesetz (German Energy Industry Act). 39
EPEX SPOT SE European Power Exchange SE. xiii, 31, 32, 56, 61
ESI Energy Service Interface. 43
ESMSF Energy System Modeling and Simulation Framework. 5, 23
ESS ENTSO-E Scheduling System. 40
ETSAP Energy Technology Systems Analysis Program. 85
ETSI European Telecommunications Standards Institute. 46

FIPA Foundation of Intelligent Physical Agents. 123

GAMS General Algebraic Modeling System. 188
GCT Gate Closure Time. 40, 41, 54, 56, 177, 178, 215, 257
GDP Gross Domestic Product. 189
GIS Global Information System. 130, 151
GPKE Geschäftsprozesse zur Kundenbelieferung mit Elektrizität (Business process for supplying customers with electricity). 52

HCDC High Voltage Direct Current. 66
HDAE Hybrid Differential Algebraic Equation. xiv, 107, 126, 164, 169, 174, 185, 241, 242, 262
HEMS Household Energy Management System. 47, 262
HVAC Heating, Ventilation, and Air Conditioning. 83

ICT Information and Communications Technology. 1, 7, 8, 46
IDE Integrated Development Environment. 111, 113, 259
IEA International Energy Agency. 85
ISR Imbalance Settlement Responsible. 59, 60

LCOE Levelized Cost of Electricity. 67, *Glossary:* Levelized Cost of Electricity
LEAP Long-range Energy Alternatives Planning System. 8
LSODAR Livermore Solver for Ordinary Differential Equations. 142, 168, 262, *Glossary:* Livermore Solver for Ordinary Differential Equations

MARKAL MARket ALlocation. xiii, 8, 11, 84–86, 252

MENT MOCES Entity. xiv, xv, 126, 127, 145, 151, 153, 154, 157–160, 162, 171, 175–179, 185, 187, 199, 202, 208, 209, 229, 231, 237, 239, 253, 255, 257, 260

MILP Mixed Integer Linear Programming. 92

MLS MODELICA Language Specification. 4, 102, 123–125, 134, 146, 147, 153, 158, 162, 164, 165, 185, 237

MMSP MODELICA Modeling and Simulation Process. 106, 113

MOCES Modeling of Complex Energy Systems. vii, viii, xiv–xvii, 4–8, 48, 52, 54, 64, 70, 73, 77, 81, 96, 99, 101, 102, 114, 116, 119, 121, 124–129, 131, 133, 134, 139, 140, 145–147, 150–154, 156–162, 164, 168–170, 173–175, 180–185, 187, 191–193, 197–201, 203, 204, 207–209, 225, 226, 231, 237–239, 251, 253, 255–258, 260, 261, 265, *Glossary:* MOCES

MPES Marktprozesse für erzeugende Marktlokationen (Strom) (Business process for accounting points producing electrical energy). 52

MS Modeling and Simulation. xiii, 15, 16, 78, 116, 123, 124, 139, 181, 185, 201, 213

MSL Modelica Standard Library. 103, 123, 181, *Glossary:* Modelica Standard Library

MTC Modelica Toolchain. 185

MTP Modelica Transformation Process. 106, 107, 111–113, 147, *Glossary:* Modelica Transformation Process

MV Market Value. xvi, 218, 219, 230, *Glossary:* Market Value

NARX Nonlinear Autoregressive Network with Exogenous Inputs. 179, 201

NEMS National Energy Modeling System. 89

NIST National Institute of Standards and Technology. xiii, 4, 41, 42, 46, 47, 259

NTC Net Transfer Capacity. 94

ODE Ordinary Differential Equation. xvii, 108, 110, 142, 163, 179, 252–254, 257, 262, 263, *Glossary:* Ordinary Differential Equation

OE Office of Electricity Delivery and Energy Reliability. 10, 11

OSeMOSYS Open Source Energy Modelling System. 86

OTC Over the Counter. 43, 61, 62

PCG Party Connected to the Grid. 49–51, 56–58, 70–72, 157, 253, *Glossary:* Party Connected to the Grid

PI Proportional–Integral. 205

POC Proof of Concept. 77, 116, 123, 132

PRIMES Price-Induced Market Equilibrium System. 87–90

PV Photovoltaic. xv, xvi, 1, 13, 27, 30, 43, 65, 67, 75, 151, 174, 175, 181, 184, 189, 193, 197, 199, 213, 230, 231

REMix Renewable Energy Mix. 95
RES Renewable Energy Source. 88, 115, *Glossary:* Renewable Energy Source
RMSE Root Mean Square Error. 197
rRMSE relative Root Mean Square Error. 197–199

SCR Self-Consumption Rate. xvi, 230–232, 235, *Glossary:* Self-Consumption Rate
SCUC Security-Constrained Unit Commitment. 90
SG-CG Smart Grid Coordination Group. 253, 255
SGAM Smart Grid Architecture Model. 253
SOP System Operator. 54, 56–60, 63, 64, 159, 174, 253, 259
SSR Self-Sufficiency Rate. xvi, 230, 231, 235, *Glossary:* Self-Sufficiency Rate
StromNZV Stromnetzzugangsverordnung (German Electricity Grid Access Ordinance). 39, 53

TIMES The Integrated MARKAL-EFOM System. 86
TSO Transmission Service Operator. 7, 39, 45, 47, 51, 60, 62, 63, 66, 71, 96, 116, 189, 196, 197

UCP Unit Commitment Problem. 57, 90–92, 94, 96, 97, 100, 116
URBS Urban Research Toolbox: Energy Systems. 65, 90, 92, 93, 95
UTC Coordinated Universal Time. 151

VAT Value Added Tax. 33, 69
VODE Variable-Coefficient ODE Solver. 142, *Glossary:* Variable-Coefficient ODE solver
VRE Variable Renewable Energy Source. xv, 30, 31, 34, 39, 56, 57, 64–68, 74, 75, 89, 92–95, 187, 189, 191–193, 198, 208, 209, 211, 213, 218, 229, 231, 235, 256, *Glossary:* Variable Renewable Energy Source

WACC Weighted Average Cost of Capital. 95

GLOSSARY

Accounting Point An accounting point for energy under the financial responsibility of exactly one BRP. 51, 60
Actor An abstract term defining a unit interacting with other units and the environment, cf. BUSINESS ACTOR and SYSTEM ACTOR. xiii, 10, 34, 40–47, 50–52, 54, 63, 72, 128, 169, 253, 254, 257, 261, 262
Agent An agent is a special type of ENTITY, only interacting using communication. 117, 120, 121, 126, 131, 147, 170, 172, 178, 180, 257

Balance Group A core domain of the ENTSO-E ROLE MODEL and a building block of liberalized energy markets. The BALANCE GROUP is an energy account under the responsibility of exactly one BRP. 50–52, 56–58, 158, 252, 253

Balance Responsible Party A PARTY that is responsible to balance its energy account, called BALANCE GROUP. It is therefore responsible for ensuring that the feed-in and consumption of all accounting points assigned to its BALANCE GROUP sum up to zero at each point in time. 39

Balance Supplier A PARTY that markets the difference between actual metered energy consumption or production and the energy bought respectively sold with firm energy contracts by the PCG [42]. A typical supplier is a vendor who has a supply contract with an end customer. 50, 54, 70, 71, 158, 212

Balancing Energy Energy used by the SOP to balance consumption and production. 61

Business Actor A type of ACTOR participating in business processes, such as a trader or customer. 34, 39, 41, 47, 169, 185, 252

Business Process Layer The second layer of the modeling framework MOCES used to model interactions between the business node part of the MENTS and their behavior, cf. PHYSICAL LAYER, INFORMATION LAYER. xvi, 5, 126, 127, 131, 132, 145, 152, 153, 157, 169, 170, 193, 202, 239, 260

Capacity Market A type of energy market in which the provision of capacity is remunerated, cf. ENERGY-ONLY MARKET. 61, 255

Causal Direction Determines which system gives the cause and which reacts to it with an effect. 18, 19, 21, 103, 131, 132

Conceptual Domain A CONCEPTUAL DOMAIN highlights the key areas of a conceptual model. Each domain groups elements of the conceptual model with a high level of interaction or based on other commonalities. E.g., in the context of SMART GRID ARCHITECTURE MODEL (SGAM), the SMART GRID COORDINATION GROUP (SG-CG) groups the domains from the point of view of responsibility. It groups (market) roles and their associated responsibilities present in the European electricity markets and the electricity system as a whole [39]. 4, 42, 51, 253

Continuous System A system that can be represented by a set of differential and algebraic equations, also known as a dynamic system, cf. DAE, ODE. 128, 254

Conventional Power Plant A technical unit producing electrical energy with a nominal power in the order of several MW up to a few GW by utilizing non-renewable ENERGY CARRIERS. The output of a conventional power plant is controllable. 31, 57, 58, 65, 101, 115

Demand Response Generic term for processes in which the load behavior of end customers is altered, e.g. by incentives to which the end customer reacts or by direct influence. 1, 54

Demand Sector A grouping of energy consumer groups. The following four main groups are typical: (1) transport, (2) industry, (3) household and (4) trade, commerce and services. 2, 25–29, 43, 71, 84, 87, 254

Differential Algebraic Equation A system of equations containing ODE and algebraic equations describing a CONTINUOUS SYSTEM: $\mathbf{0} = \mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, \mathbf{y}, t), \mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{y}, t)$. 101

Differential Algebraic System Solver An efficient and widely used SOLVER for DAES, see Petzold [174] for details. 110

Discrete-Time Variable A variable that changes its value only at discrete time steps. "[It] is a piecewise constant signal which changes its values only at event instants during simulation." ([120]). 134, 135, 137, 140

Distribution Grid The part of the electricity grid whose original function is to distribute the energy to the end consumers. Most DECENTRALIZED ENERGY RESOURCES (DER) are connected to the distribution grid. The distribution network operates with different voltage levels from 230.00 V up to 110.00 kV. 7, 34, 263

Dynamic Scheduling A method for handling SIMULATION TIME and scheduling agents' actions in a multi-agent simulation. 121, 122, 139, 162, 170

Dynamo A modeling language developed in the 1950s and used for the WORLD3 model. 115

Electricity Market 1. A market where the commodity electricity is bought, sold and traded. 2. An abstract term for the POWER SYSTEM from the point of view of ACTORS active in the two liberalized areas of the electric power industry: wholesale market and end-costumer sales. 10, 34, 51, 174, 258, 263, 264

Energy Carrier "A substance or phenomenon that can be used to produce mechanical work or heat or to operate chemical or physical processes" (ISO 13600:1997) This definition is extremely general and allows to define a donkey cart full of tensioned springs as an energy carrier. Typically, however, the term only addresses technically used carriers, such as petrol, electricity, and enthalpy (heat). Sometimes the term is synonymous with the term secondary energy (cf. PRIMARY ENERGY). 1, 9–11, 23, 26, 27, 29, 30, 84, 87–89, 218, 253, 254, 260

Energy Demand Sector cf. DEMAND SECTOR. 9, 11

Energy Sector An energy sector encompasses all ENTITIES directly connected to a specific ENERGY CARRIER. 11, 27, 28

Energy Service A service typically provided by the use of FINAL ENERGY, such as thermal comfort in a house or transportation of persons. The term *service* indicates that it does not necessarily have to be provided through the use of FINAL ENERGY. It is conceivable to fulfill the function through alternative approaches, for example, through better isolation, or technologies enabling work at home and online meetings. 2, 9, 41, 43, 255

Energy System Traditional definition: a "process chain [...] from the extraction of primary energy to the use of final energy to supply services and goods" (Pfenninger et al. [4]). Modern definition: "a system primarily designed to supply ENERGY SERVICES to end users" ([9]). xiii, 1, 2, 4, 5, 9–11, 15, 17, 23, 24, 26–29, 64, 76, 77, 81, 84–87, 89, 92, 112, 114–116, 132, 151, 255, 257, 260, 265

Energy System Model A model of an ENERGY SYSTEM. The term is mainly used for models that span nations and multiple energy carriers. xiii, 11, 85

Energy-Only Market A type of energy market in which only quantities of energy are remunerated. Typically, the term is used to describe the overall design of the electricity market and not to describe the characteristics of an individual market, cf. CAPACITY MARKET. 61, 67, 253

Energy-Related Service A service that is related to the ENERGY SYSTEM in some way. It differs from an ENERGY SERVICE in that its related task does not directly involve the provision of FINAL ENERGY. Possible examples are metering, billing, trading. 2, 41

Entity The smallest active unit of a system that interacts with other ENTITIES using communication, a physical coupling or a combination of both through interfaces, cf. MENT. 1, 4, 5, 11, 252, 254, 255

ENTSO-E Role Model A terminology that enables a dialogue between market participants from different nations. In particular, the goal is to support the development of IT systems. xiii, 5, 46–49, 51, 252, 255

European Conceptual Model A conceptual model of the SMART GRID developed by the SG-CG. It is based on the NIST CONCEPTUAL MODEL and aligned with the ENTSO-E ROLE MODEL. xiii, 46–48

Event An EVENT occurs, and indicates that something happens at a particular point in time. An EVENT has no duration. In the context of MOCES, *something happens* is a synonym for an EVENT CONDITION becoming true. An EVENT can, and typically does, trigger one or a set of actions linked to it, cf. STATE EVENT and TIME EVENT. 105, 110, 120–122, 162, 163, 165, 170, 255, 262

Event Condition An EVENT CONDITION is a condition that describes the triggering of an EVENT. Under certain circumstances, the points in time of occurrence can be determined a priori, cf. TIME EVENT. 130, 135–139, 163–166, 170, 255, 262

Experiment A methodical investigation to gather data or information, cf. SIMULATION. 117, 258

Experimental Frame "The set of experiments for which the model is valid" (Zeigler citet after Cellier [1]). 3, 6, 73, 75, 81

Final Energy The energy that is converted into USEFUL ENERGY by technical units. 2, 9, 25, 26, 255, 263

Flexibility The changes in consumption/injection of electrical power from/to the power system from their current/normal patterns in response to certain signals, either voluntarily or mandatory. [41]. 44, 46, 71

Full-Load Hours Measure for the use of power plants: full-load hours = annual work/ (nominal output · 8760.00 h). If used in the context of VRE-based power plants, it is a measure of their performance, mainly influenced by their location. xvi, 30, 31

Grid Connection Point The junction point from the private facility to the public grid. 43

Grid Operator "A party that operates one or more grids" ([42]). 7

Hybrid Model A MODEL of a HYBRID SYSTEM. 83, 113, 114

Hybrid System A system involving continuous as well as discrete event dynamics. This hybrid behavior can be found in continuous systems that include a phased operation, such as the famous bouncing ball. However, as most discrete event systems are human-made, a hybrid system is mostly a continuous process controlled by a digital controller. 8, 105, 107, 128, 129, 256

Imbalance Settlement Period "the time unit for which BRP imbalance is calculated" ([175]). In the European context, this period should be harmonized to 15.00 min. 40, 58, 261

Impure Function In the context of MOCES, an IMPURE FUNCTION is a subroutine called by a *Physical Node* that does *not* behave like a mathematical function. The same input passed to the function usually does not result in the same output. The *call* of the function does not affect the local or INTERNAL STATE of the linked *Business Node*. An example is a *true* random function. Cf. IMPURE SIDE EFFECT FUNCTION, PURE FUNCTION, and PURE SIDE EFFECT FUNCTION. 105, 164, 256, 260

Impure Side Effect Function In the context of MOCES, an IMPURE SIDE EFFECT FUNCTION is a subroutine called by a *Physical Node* that behaves like an IMPURE FUNCTION. In addition, its *call* may affect the local or INTERNAL STATE of the linked *Business Nodes* and therefore may influence the simulation result. An example is a function that increments the INTERNAL STATE by one and returns its value; cf. PURE FUNCTION, and PURE SIDE EFFECT FUNCTION. 165, 256, 260

Index "The number of differentiations needed to transform the DAE into an ODE" ([176]). 110, 112

Information Layer The INFORMATION LAYER is the third layer of MOCES. It provides a consistent environment to the MENTS. This layer is non-regenerative, meaning its behavior is not influenced by any MENT. 5, 126, 127, 131, 132, 145, 150, 152–154, 191, 195, 199, 253, 257, 260

Internal State Part of the state of an AGENT that can only be changed by the agent itself, cf. LOCAL STATE. 118, 119, 129, 164, 167, 170, 171, 256, 257, 260

Lead Time Generally the TIME INTERVAL between the initiation or fixation of an action and its execution. In particular the TIME INTERVAL between GCT and TIME OF DELIVERY. 29, 40, 56, 63, 64

Levelized Cost of Electricity "[An] economic assessment of the average total cost to build and operate a power-generating asset over its lifetime divided by the total energy output of the asset over that lifetime" ([177]). 67

Linear Optimization Model An optimization problem that has the following form: $\max \mathbf{c}^\mathsf{T}\mathbf{x}$ subject to $\mathbf{A}\mathbf{x} \leq \mathbf{b}$ and $\mathbf{x} \geq \mathbf{0}$, $\mathbf{A} \in \mathbb{R}^{m\times n}, \mathbf{b} \in \mathbb{R}^{m\times 1}, \mathbf{c} \in \mathbb{R}^{1\times n}, \mathbf{x} \in \mathbb{R}^{n\times 1}$. 17, 259, 261

Livermore Solver for Ordinary Differential Equations SOLVER for ODES written by A. C. Hindmarsh and L. R. Petzold to solve stiff or non-stiff system; it includes a root finding mechanism. See Hindmarsh and Petzold [178] for details. 142

Local Energy System An ENERGY SYSTEM that involves regions of maximally NUTS-3 level, but not higher, and has ACTORS who prefer to interact with ACTORS of the same local system in order to pursue their own or a common goal. 54

Local Optimizer A specific kind of MENT, developed to investigate the effect of components that aim to cover power consumption as locally as possible and to increase the self-consumption rate of renewable energy sources. xv, 187, 188, 192, 208–214, 216, 218, 226, 229–233, 235

Local State Part of the state of an AGENT that can be influenced by other AGENTS and the environment. It describes how an AGENT sees the environment, cf. INTERNAL STATE. 118, 122, 129, 131, 170–172, 257

Lotka–Volterra equations A system of two nonlinear coupled differential equations which describes the population of predators and prey. The system is often used as an example to demonstrate the feasibility of agent-based modeling of a dynamic system. 123

Marginal Cost Operational costs to produce one unit of electrical energy. Mainly determined by the fuel costs divided by the efficiency of the power plant. The fuel costs include the costs for the transport and CO_2 emissions. 28, 31, 32, 97, 258

Market Area "An area made up of several MARKET BALANCE AREAS interconnected through AC or DC links. Trade is allowed between different MARKET BALANCE AREAS with common MARKET RULES for trading across the interconnection" ([179]). 62

Market Balance Area "A geographic area consisting of one or more Metering Grid Areas with common MARKET RULES for which the settlement responsible party carries out a balance settlement and which has the same price for imbalance. A MARKET BALANCE AREA may also be defined due to bottlenecks" ([180]). 50, 51, 54, 195, 258

Market Rules The sum of all acts, regulations, and provisions on a legal or contractual basis that must be complied with by ELECTRICITY MARKET participants. With these legal specifications, the legislator wants to ensure a functioning market and thus a reliable electricity system. [181]. 37, 39, 40, 53, 56, 126, 258

Market Value Average energy source-specific market price for a period of time under consideration. 218, 219, 232

Merit Order The list of power plants sorted according to their marginal costs starting with the power plant with the lowest MARGINAL COSTS. 67, 218

Metamodel In the context of simulation-based optimization, a METAMODEL is a model that mimics the behavior of a more complex model. METAMODELS are mostly deterministic and cheap to calculate, cf. Barton and Meckesheimer [18]. 16, 17, 258

MOCES Modeling and simulation framework developed within this thesis. The framework involves: a modeling concept; an implementation in Modelica and C++; corresponding base model; and, guidelines how to use MOCES. 4

Model "A MODEL (M) for a SYSTEM (S) and an EXPERIMENT (E) is anything to which E can be applied in order to answer questions about S." (Marvin Minsky cited after Cellier [1]). 3, 11–15, 17, 102, 106, 117, 120, 126, 127, 138, 147, 151, 164, 256, 258, 259, 261, 262

Modelica "A freely available, object-oriented language for modeling of large, complex, and heterogeneous systems" ([120]). 115, 116, 123, 124

Modelica Standard Library Free, and a kind of reference library, from the Modelica Association providing base models for different domains, i.a., mechanical, electrical, thermal, and control systems, and hierarchical state machines [159]. 103

Modelica Transformation Process The fully automated process to transform a MODELICA MODEL into an executable or rather a SIMULATOR. 106, 163, 179

Modelica IDE An IDE that provides facilities to modelers for the development of MODELICA models and their simulation. Examples are DYMOLA and OPENMODELICA. 4, 7, 106, 110–112, 114, 145, 162, 170, 181, 184, 185

Modeling Language An artificial, human- and machine-readable language with which the dynamic behavior of a SYSTEM can be unambiguously described. It is only a modeling language if at least one interpreter (simulator) exists that can determine the behavior solely from the –potentially automatically transformed– MODEL. 102, 114, 124, 125, 238

N-1 Security A criterion that states that the power system must be operational even if any one element fails. 72

Networked Control System A control system in which the control loops are closed via a communication network. 8

Nist Conceptual Model A set of views (diagrams) and descriptions that are the basis for discussing the characteristics, uses, behavior, interfaces, requirements, and standards [...] for the SMART GRID developed by the NIST. (Adapted from National Institute of Standards and Technology (NIST) [37]). xiii, xvi, 5, 41–47, 255

NUTS Acronym from the French *Nomenclature des unités territoriales statistiques.* A geocode standard for referencing to subdivisions of countries developed by the European Union. In Germany, the NUTS-1 level refers to the states, the NUTS-2 level to the government regions, and the NUTS-3 level to districts. In total there are approximately 400 NUTS-3 districts in Germany. 5, 182, 257

Observer A person or a group of persons that observe a system and build a model for it, or that observe a run of an experiment or of a simulation and interpret the results. 151

Operating Reserve A load and generation capacity reserved and under control of the SOP to reconcile electrical generation with consumption. 59

Optimization Decision-making process with the aim of finding the best possible solution. 11, 81, 84

Optimization Model A model primarily designed to be used in a decision problem, cf. LINEAR OPTIMIZATION MODEL. 3, 6, 17, 187, 231, 259

Optimization Model Approach An approach using an OPTIMIZATION MODEL to answer a specific question. 6, 81, 82

Ordinary Differential Equation A system of ordinary differential equations: $\mathbf{0} = \mathbf{f}(\dot{\mathbf{x}}, \mathbf{x}, t)$. 108

Overlay Grid The vision of a continent-spanning grid enabling the transport of large amounts of electrical energy over long distances, also referred to as super grid. 27, 93

Party A legal entity, either a natural person or an organization. xiii, 2, 46, 47, 49–51, 56, 126, 128, 133, 212, 253, 260, 261, 263

Party Connected to the Grid A PARTY connected to the Grid. 49, 50

Perfect Market The model of a market where supply and demand are ideally matched. Assumptions made and conditions which have to be true include the following: the product is homogeneous, there are many suppliers and all market actors have the same perfect information. 39, 192, 201

Physical Layer The first and major layer of the modeling framework MOCES used to model interactions between the physical nodes of the MENTS and their behavior, cf. BUSINESS PROCESS LAYER, cf. INFORMATION LAYER. xvi, 4, 126, 127, 131, 145, 157, 192, 253

Power System The extract of all elements of the ENERGY SYSTEM directly linked to the ENERGY CARRIER electricity. xiii, 4, 5, 7, 8, 10, 15, 23, 28, 29, 34–36, 38, 41–47, 50, 52, 64, 66–68, 72–76, 81, 82, 88–90, 92, 93, 95, 96, 98, 114, 117–119, 124, 126, 139, 157, 169, 182, 187, 211, 229, 237, 254

Primary Energy An energy form found in nature that has not been subjected to any human-engineered conversion or transformation process [182]. It is either of type non-renewable or renewable. 9, 25, 254

Pure Function In the context of MOCES, a PURE FUNCTION is a subroutine called by a *Physical Node* that behaves like a mathematical function. The same input passed to the function results in the same output. Its *call* does not influence the simulation result. Cf. IMPURE FUNCTION, IMPURE SIDE EFFECT FUNCTION, and PURE SIDE EFFECT FUNCTION. 105, 139, 164, 167, 168, 256, 257, 260

Pure Side Effect Function In the context of MOCES, a PURE SIDE EFFECT FUNCTION is a subroutine called by a *Physical Node* that behaves like a mathematical function, cf. PURE FUNCTION, but its *call* may have side effects to the local or INTERNAL STATE of the linked *Business Nodes* and therefore may influence the simulation result. Cf. IMPURE FUNCTION and IMPURE SIDE EFFECT FUNCTION. 256, 257, 260

Random Walk A mathematical concept of a path through a space composed of random steps. In the 1-dimensional case with finite step length $N \in \mathbb{N}$, the following applies: $x_i = x_0 + \sum_{i \in N} z_i$, with z_i being a random number. 199, 200

Real-Time The ability to respond to an event within a defined deadline. Usually, this deadline is in the range of seconds. 57

Redispatch The temporary and planned intervention of the system operator in the market-driven feed-in behavior of power plants to prevent overloading of single grid assets and ensure grid stability. 51, 56

Renewable Energy Source An energy source which is continuously renewed on a human timescale, such as biomass and hydro power and further VARIABLE RENEWABLE ENERGY SOURCES like solar and wind. 88

Role The intended external behavior of a PARTY. A ROLE is atomic and can therefore not be shared between several PARTIES. xiii, 37, 42, 46–48, 51, 52, 54, 101, 158, 169, 212, 261

Schedule In general: a reference set of values that specify a time sequence of any quantity for a period. In particular: "A reference set of values representing the generation, consumption or exchange of electricity for a given time period" ([34]). 40, 54, 56–58, 62, 159, 263

Self-Consumption Rate The quota of produced energy which is used *locally*. 230

Self-Sufficiency Rate The quota of consumed energy which is produced *locally*. 230

Service A functionality governed by an ACTOR provided through a well-defined interface. 45, 51

Settlement Period See IMBALANCE SETTLEMENT PERIOD. 263

Signal A function that conveys information about the behavior or attributes of some phenomenon ([183]). 130

Simplex Method An algorithm to find an optimal solution for a LINEAR OPTIMIZATION MODEL. 17, 188

Simulation The activity of conducting an experiment on a model. 11, 13–15, 81, 84, 110, 120, 125, 128, 138, 168, 185, 256, 262

Simulation Clock The simulation clock provides the SIMULATION TIME within a SIMULATION MODEL. It can be seen as a model of a WALL CLOCK. It is common practice to initialize the simulation clock with 0. In contrast and compared to the wall time, the SIMULATION TIME can progress faster or slower and typically not uniformly. In MOCES, points in time defined by the SIMULATION CLOCK are always linked to a point in time defined by the WALL CLOCK. 151, 261

Simulation Model A mathematical MODEL used by a SIMULATOR to determine the temporal and spacial behavior of a SYSTEM under a given set of conditions (inspired by Lund et al. [90]). 3, 6, 17, 187, 261, 262

Simulation Time The TIME PERIOD measured by the SIMULATION CLOCK. 105, 120, 142, 151, 152, 165, 170, 180, 254, 261

Simulative Approach An approach using a SIMULATION MODEL to answer a specific question. 6, 81, 82

Simulator A SIMULATOR performs a SIMULATION. Typically, a SIMULATOR is a computer program that runs a SIMULATION, also called experiment, given a MODEL and returns a trajectory. If the MODEL is of type ODE, DAE or HDAE, the SIMULATOR usually uses an interchangeable SOLVER, such as DASSL or LSODAR. 15, 17, 106, 110, 120, 167, 168, 170, 174, 259, 261, 262

Smart Grid "Generally refers to a class of technology people are using to bring utility electricity delivery systems into the 21st century, using computer-based remote control and automation" ([10]). 10, 11, 46, 255, 259

Socio-technical System "SOCIO-TECHNICAL SYSTEMS, [...], consist of many technical artifacts (machines, factories, pipelines, wires, etc.) and social entities (individuals, companies, governments, organizations, institutions, etc.)" (Dam et al. [84]). 77, 127, 262

Solver An implemented algorithm capable to numerically solve an ODE or a DAE, cf. DASSL. 110, 112, 113, 141, 145, 163, 164, 167, 168, 180, 254, 257, 262, 263

Standard Load Profile A load profile used to describe the temporal behavior of consumers about which no time-resolved information based on measurements is available. Standard load profiles are largely unsuitable for describing individual consumers but are suitable for representing sums of consumers of the same type, for example, households. 70

State Event A kind of EVENT, whose occurrence is not known a priori as the EVENT CONDITION includes time-variant values whose behavior is unknown. xiv, 139, 142, 162–164, 167, 255

System "A regularly interacting or interdependent group of items forming a unified whole" ([13]). 11, 12, 14, 15, 17, 52, 102, 117, 125, 145, 258, 259, 261, 262

System Actor A SYSTEM ACTOR is an ACTOR participating in a physical process, such as a transmission line or a HEMS. xiii, 34–36, 47, 157, 185, 252, 262

System Boundary Demarcates the SYSTEM of interest from the environment. 9, 12

Time Delay System A system including delays, e.g., due to signal processing needing some time. 8

Time Event A special kind of an EVENT, whose occurrence is known a priori as its EVENT CONDITION only includes time-independent variables and the time itself. xiv, 139, 142, 162–164, 167, 255, 256

Time Interval A length of time, e.g., three months. [184]. 40, 62, 63, 92, 151, 257

Time of Delivery The agreed time at which a product is to be delivered. In particular, the time of delivery for electricity. The time of delivery is normally a TIME PERIOD called SETTLEMENT PERIOD. 40, 41, 53, 56, 57, 59, 71, 257

Time Period "An anchored duration of time" ([184]), e.g., 2017-01-01 00:00 until 2017-01-02 12:35. 40, 41, 53, 54, 59, 61, 63, 64, 91, 261, 263

Time Slice A finite or infinite amount of usually non-consecutive TIME PERIODS defined by a property of the TIME PERIODS, e.g. weekdays between 08:00 – 20:00. 85, 86, 95

Trader A PARTY that performs transactions at the different ELECTRICITY MARKETS. 50

Transaction Time Point in time when a fact, e.g. a SCHEDULE addressing a TIME PERIOD in the future or in the past in relation to transaction time, is published. This definition is inspired by the definition of Snodgrass [184] in the context of time-oriented databases. Cf. VALID TIME and TIME PERIOD. 40, 56

Transmission Grid Part of the grid that enables electrical power to be transported over long distances. Typically implemented as a 380.00 kV or 220.00 kV AC system and linked to several DISTRIBUTION GRIDS via electrical substations. 7, 34

Unix Time A system for describing a point in time. Put simply, it counts the seconds since January 1, 1970. 263

Useful Energy The energy actually required to satisfy a need such as heat, lighting or transportation. Part of the FINAL ENERGY after its conversion. 9, 26, 85, 87, 88, 256, 263

Useful Energy Demand Demand for USEFUL ENERGY. 11, 84, 86

Valid Time The point in time when a fact was true in the modeled reality [184]. 263

Variable Renewable Energy Source A fluctuating renewable energy source. The major sources are solar and wind energy. Power plants based on them are limited regarding their ability to be dispatched. In general, their output can only be limited and not fully controlled. 30, 261

Variable-Coefficient ODE solver SOLVER for ODES. 142

Voltage Level The different levels of voltage used in the electrical grid. Extra high voltage: nominal voltage above or equal to 220.00 kV; high voltage: between 45.00 kV and 150.00 kV, typically 110.00 kV; medium voltage: between 1.00 kV and 45.00 kV; low voltage: below or equal to 1.00 kV. Cited after [33, p. 56]. 34

Wall Clock The wall clock gives the present (not influenceable) point in time in relation to a specified system, e.g., the UNIX TIME, cf. WALL TIME. 130, 151, 261, 264

Wall Time The wall time is a time period measured by the WALL CLOCK. 140, 142, 151, 152, 180, 200, 263

Wholesale Market See ELECTRICITY MARKET. 31

World3 A dynamic simulation model that serves as the foundation for the report *The limits to growth: A report for the Club of Rome's project on the predicament of mankind* [134] . 115, 254

OWN PUBLICATIONS

The MOCES modeling framework is described in the following two publications:

- Exel and Frey: *Modeling and simulation of local flexibilities and their effect to the entire power system*, [146]
- Exel, Felgner, and Frey: *Multi-domain modeling of distributed energy systems - the MOCES approach*, [145]

Use cases of modeling and simulation in the context of power systems have been described by the author in the following joint works:

- Stüber, Exel, and Frey: *Using modelling and simulation as a service (MSaaS) for facilitating flexibility-based optimal operation of distribution grids*, [185]
- Felgner, Meiers, Exel, and Frey: *Design of distributed energy systems: Role and requirements of modeling and simulation*, [118]
- Exel and Frey: *Toward a decentralized forecast system for distributed power generation*, [186]

The designed architecture of MOCES is based on the experience with MODELICA and the use of the language to investigate components in the ENERGY SYSTEM:

- Exel, Frey, Wolf, and Oppelt: *Re-use of existing simulation models for DCS engineering via the functional mock-up interface*, [147]
- Felgner, Exel, Nesarajah, and Frey *Component-Oriented Modeling of Thermoelectric Devices for Energy System Design*, [187]
- Nesarajah, Exel, and Frey: *Modelica® library for dynamic simulation of thermoelectric generators*, [148]
- Felgner, Exel, and Frey: *Model-based design and validation of waste heat recovery systems*, [149]
- Felgner, Exel, and Frey: *Component-oriented ORC plant modeling for efficient system design and profitability prediction*, [150]

BIBLIOGRAPHY

[1] François E. Cellier. *Continuous System Modeling*. Springer, New York, NY, 1991. doi:10.1007/978-1-4757-3922-0.

[2] Bernard P. Zeigler. *Theory of Modeling and Simulation*. John Wiley, New York, NY, 1976.

[3] Leslie G. Fishbone and Harold Abilock. Markal, a linear programming model for energy systems analysis: Technical description of the bnl version. *International Journal of Energy Research*, 5(4):353–375, January 1981. doi:10.1002/er.4440050406.

[4] Stefan Pfenninger, Adam Hawkes, and James Keirstead. Energy systems modeling for twenty-first century energy challenges. *Renewable and Sustainable Energy Reviews*, 33:74–86, May 2014. doi:10.1016/j.rser.2014.02.003.

[5] National Institute of Standards and Technology (NIST). NIST framework and roadmap for smart grid interoperability standards — release 3.0, October 2014. doi:10.6028/nist.sp.1108r3.

[6] D. Connolly, H. Lund, B. V. Mathiesen, and M. Leahy. A review of computer tools for analysing the integration of renewable energy into various energy systems. *Applied Energy*, 87(4):1059–1082, April 2010. doi:10.1016/j.apenergy.2009.09.026.

[7] S. Jebaraj and S. Iniyan. A review of energy models. *Renewable and Sustainable Energy Reviews*, 10(4):281–311, August 2006. doi:10.1016/j.rser.2004.09.004.

[8] S. C. Mueller, H. Georg, J. J. Nutaro, E. Widl, Y. Deng, P. Palensky, M. U. Awais, M. Chenine, M. Kuch, M. Stifter, H. Lin, S. K. Shukla, C. Wietfeld, C. Rehtanz, C. Dufour, X. Wang, V. Dinavahi, M. O. Faruque, W. Meng, S. Liu, A. Monti, M. Ni, A. Davoudi, and A. Mehrizi-Sani. Interfacing power system and ICT simulators: Challenges, state-of-the-art, and case studies. *IEEE Transactions on Smart Grid*, 9(1):14–24, March 2016. doi:10.1109/TSG.2016.2542824.

[9] "Energy System", Wikipedia.org. `https://en.wikipedia.org/wiki/Energy_system`.

[10] "Smart Grid", Energy.gov. `https://energy.gov/oe/services/technology-development/smart-grid`.

[11] Volker Quasching. Sektorkopplung durch die Energiewende, June 2016. http://pvspeicher.htw-berlin.de/wp-content/uploads/2016/05/HTW-2016-Sektorkopplungsstudie.pdf, Last accessed: 23 May 2021.

[12] Henrik Lund, Poul Alberg Østergaard, David Connolly, and Brian Vad Mathiesen. Smart energy and smart energy systems. *Energy*, 137:556–565, October 2017. doi:10.1016/j.energy.2017.05.123.

[13] "System", Merriam-Webster.com. https://www.merriam-webster.com/dictionary/system.

[14] Peter Fritzson. *Principles of Object Oriented Modeling and Simulation with Modelica 3.3*. John Wiley & Sons, Inc., November 2014. doi:10.1002/9781118989166.

[15] Saarland University Archive of Chair of Automation. Picture of physical grid model at saarland university.

[16] Walter J. Karplus. The spectrum of mathematical modeling and systems simulation. *SIGSIM Simulation Digest*, 9(1):32–38, September 1977. doi:10.1145/1102505.1102522.

[17] François E. Cellier and Ernesto Kofman. *Continuous System Simulation*. Springer, Boston, MA, 2006. doi:10.1007/0-387-30260-3.

[18] Russell R. Barton and Martin Meckesheimer. Chapter 18 metamodel-based simulation optimization. *Handbooks in Operations Research and Management Science*, 13:535–574, 2006. doi:10.1016/S0927-0507(06)13018-2.

[19] Jack P. C. Kleijnen. A comment on blanning's "metamodel for sensitivity analysis: the regression metamodel in simulation". *Interfaces*, 5(3):21–23, May 1975. doi:10.1287/inte.5.3.21.

[20] Henry M. Paynter and Peter Briggs. *Analysis and design of engineering systems: class notes for M.I.T. course 2.751*. M.I.T. Press, 1961. http://hdl.handle.net/2027/mdp.39015064874921. Last accessed: 23 May 2021.

[21] Karl Johan Åström, Hilding Elmqvist, Sven Erik Mattsson, et al. Evolution of continuous-time modeling and simulation. In *Proceedings of the 12th European Simulation Multiconference, June 16–19, 1998, Manchester, UK*, pages 9–18, June 1998.

[22] Klaus Janschek. *Mechatronic Systems Design*. Springer, Berlin, Heidelberg, 2012. doi:10.1007/978-3-642-17531-2.

[23] Mario Schmidt. Der Einsatz von Sankey-Diagrammen im Stoffstrommanagement, December 2006. https://www.econstor.eu/handle/10419/97580, Last accessed: 23 May 2021.

[24] AG Energiebilanzen. Auswertungstabellen zur Energiebilanz für die Bundesrepublik Deutschland 1990 bis 2018, August 2019. https://ag-energiebilanzen.de/index.php?article_id=29&fileName=awt_2018_deutsch.xlsx, Last accessed: 23 May 2021.

[25] Federal Ministry for the Environment, Nature Conservation and Nuclear Safety (BMU). Climate action in figures, 2017. https://www.bmu.de/fileadmin/Daten_BMU/Pools/Broschueren/klimaschutz_in_zahlen_2017_en_bf.pdf, Last accessed: 21 March 2021.

[26] Federal Ministry for Economic Affairs and Energy (BMWi). Energy data: Complete edition, May 2021. https://www.bmwi.de/Redaktion/EN/Artikel/Energy/energy-data.html, Last accessed: 23 May 2021.

[27] Deutsche Energie-Agentur GmbH (dena). Kurzanalyse der Kraftwerksplanung in Deutschland bis 2020 (Aktualisierung), February 2010. https://effizienzgebaeude.dena.de/fileadmin/dena/Dokumente/Pdf/9108_Studie_Kurzanalyse_der_Kraftwerksplanung_in_D_bis_2020.pdf, Last accessed: 23 May 2021.

[28] German Federal Network Agency. Smard Strommarktdaten (Electricity Market data), April 2018. https://www.smard.de/blueprint/servlet/page/en/5790, Last accessed: 23 May 2021.

[29] Agora Energiewende. Die Energiewende im Stromsektor: Stand der Dinge 2015, February 2016. https://www.agora-energiewende.de/fileadmin/Projekte/2016/Jahresauswertung_2016/Agora_Jahresauswertung_2015_Slides_web_DE.pdf, Last accessed: 23 May 2021.

[30] German Federal Network Agency. Monitoringbericht 2017, December 2017. https://www.bundesnetzagentur.de/SharedDocs/Mediathek/Monitoringberichte/Monitoringbericht2017.pdf?__blob=publicationFile&v=4, Last accessed: 23 May 2021.

[31] German Federal Network Agency. EEG in Zahlen 2016, 2017. https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/Energie/Unternehmen_Institutionen/ErneuerbareEnergien/ZahlenDatenInformationen/EEGinZahlen_2016_BF.pdf?__blob=publicationFile&v=3, Last accessed: 23 May 2021.

[32] Valentin Crastan. *Elektrische Energieversorgung 2*. Springer Vieweg, Berlin, Heidelberg, 2017. doi:10.1007/978-3-662-48965-9.

[33] Valentin Crastan. *Elektrische Energieversorgung 1*. Springer Vieweg, Berlin, Heidelberg, 2015. doi:10.1007/978-3-662-45985-0.

[34] "Schedule", ENTSO-E. https://www.entsoe.eu/data/data-portal/glossary/Pages/home.aspx. accessed: 19 March 2018.

[35] Nomination of schedules in germany using the ENTSO-E scheduling system (ESS) — version: 2, release: 1, 2010. https://www.transnetbw.de/downloads/strommarkt/bilanzkreismanagement/ess_vorgaben_in_deutschland_v2r1_englisch.pdf, Last accessed: 23 May 2021.

[36] European Transmission System Operators ETSO. ETSO scheduling system (ESS) implementation guide — version: 2, release: 3, April 2003. https://www.entsoe.eu/Documents/EDI/Library/cim_based/archive/ESS%20V4R1%20and%20older.zip, Last accessed: 23 May 2021.

[37] National Institute of Standards and Technology (NIST). NIST framework and roadmap for smart grid interoperability standards — release 1.0, January 2010. doi:10.6028/NIST.sp.1108.

[38] European Commission, Derectorate-General for Energy. Smart grid mandate – standardization mandate to european standardisation organisations (esos) to support european smart grid deployment. http://ec.europa.eu/growth/tools-databases/mandates/index.cfm?fuseaction=search.detail&id=475, March 2011.

[39] Smart Grid Coordination Group (SG-CG). Overview of SG-CG methodologies, 2014. https://ftp.cencenelec.eu/EN/EuropeanStandardization/HotTopics/SmartGrids/SGCG_Methodology_Overview.pdf, Last accessed: 23 May 2021.

[40] European Network of Transmission System Operators for Electricity ENTSO-E. The harmonised electricity market role model, January 2015. https://eepublicdownloads.entsoe.eu/clean-documents/EDI/Library/HRM/2015-September-Harmonised-role-model-2015-01.pdf, Last accessed: 23 May 2021.

[41] Smart Grid Coordination Group (SG-CG). Flexibility management: Overview of the main concepts of flexibility management, 2014. https://ftp.cencenelec.eu/EN/EuropeanStandardization/HotTopics/SmartGrids/SGCG_Methodology_FlexibilityManagement.pdf, Last accessed: 23 May 2021.

[42] European Network of Transmission System Operators for Electricity ENTSO-E. The harmonised electricity market role model — version: 2017-01 approved, January 2017. https://eepublicdownloads.entsoe.eu/clean-documents/EDI/Library/HRM/Harmonised_Role_Model_2017-01.pdf, Last accessed: 23 May 2021.

[43] Bundesverband der Energie– und Wasserwirtschaft (BDEW) (German Association of Energy and Water Industries). Das deutsche Marktrollen-Modell, 2014.

[44] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Festlegung zur Regelung der Erbringung von Sekundärregelleistung und Minutenreserve durch Letztverbraucher in Stromlieferverträgen, 2017. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2017/BK6-17-046/BK6-17-046_Beschluss%20vom%2014_09_2017_download.pdf?__blob=publicationFile&v=3, Last accessed: 11 March 2021.

[45] Bundesverband der Energie– und Wasserwirtschaft (BDEW) (German Association of Energy and Water Industries). BDEW-Roadmap Daten und Marktkommunikation, January 2017. https://www.bdew.de/media/documents/Stn_20170123_Roadmap-Marktkommunikation.pdf.

[46] European Network of Transmission System Operators for Electricity ENTSO-E. The ENTSO-E scheduling system (ESS) implementation guide, 2012.

[47] ENTSO-E. Continental europe operation handbook, 01 2000. https://www.entsoe.eu/publications/system-operations-reports/operation-handbook/Pages/default.aspx.

[48] European Network of Transmission System Operators for Electricity ENTSO-E. The ENTSO-E settlement process (ESP) implementation guide, 2010.

[49] Verband der Netzbetreiber VDN – e.V. beim VDEW. Transmissioncode 2007, network and system rules of the german transmission system operators, 2007. https://www.vde.com/resource/blob/937766/bfe325518ace878935966b6efbc493e4/transmissioncode-2007--network-and-system-rules-of-the-german-transmission-system-operators-data.pdf.

[50] German Federal Network Agency. Modell zur Berechnung des regelzonenübergreifenden einheitlichen Bilanzausgleichsenergiepreises (reBAP) unter Beachtung des Beschlusses BK6-12-024

der Bundesnetzagentur vom 25.10.2012, 2012. https://www.regelleistung.net/ext/static/rebap.

[51] Tobias Weißbach. *Verbesserung des Kraftwerks- und Netzregelverhaltens bezüglich handelsseitiger Fahrplanänderungen*. PhD thesis, Universität Stuttgart, 2009. doi:10.18419/opus-1823.

[52] Frieder Borggrefe and Karsten Neuhoff. Balancing and intraday market design: Options for wind integration. *SSRN Electronic Journal*, 2011. doi:10.2139/ssrn.1945724.

[53] Cherrelle Eid, L. Andrew Bollinger, Binod Koirala, Daniel Scholten, Emanuele Facchinetti, Johan Lilliestam, and Rudi Hakvoort. Market integration of local energy systems: Is local energy management compatible with european regulation for retail competition? *Energy*, 114:913–922, November 2016. doi:10.1016/j.energy.2016.08.072.

[54] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Beschluss in dem Verwaltungsverfahren wegen der Standardisierung vertraglicher Rahmenbedingungen für Eingriffsmöglichkeiten der Übertragungsnetzbetreiber in die Fahrweise von Erzeugungsanlagen, 2011. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2011/BK6-11-098/BK6_11_098_Beschluss.pdf?__blob=publicationFile&v=2, Last accessed: 11 March 2021.

[55] Matthias P. Nowak, Rüdiger Schultz, and Markus Westphalen. A stochastic integer programming model for incorporating day-ahead trading of electricity into hydro-thermal unit commitment. *Optimization and Engineering*, 6(2):163–176, June 2005. doi:10.1007/s11081-005-6794-0.

[56] EPEX SPOT. EPEX SPOT operational rules, May 2021. https://www.epexspot.com/sites/default/files/download_center_files/EPEX%20SPOT%20Market%20Rules_7.zip, Last accessed: 23 May 2021.

[57] Tobias Weissbach and Ernst Welfonder. High frequency deviations within the european power system: Origins and proposals for improvement. In *2009 IEEE/PES Power Systems Conference and Exposition*. IEEE, March 2009. doi:10.1109/psce.2009.4840180.

[58] Lion Hirth and Inka Ziegenhagen. Balancing power and variable renewables: Three links. *Renewable and Sustainable Energy Reviews*, 50:1035–1051, October 2015. doi:10.1016/j.rser.2015.04.180.

[59] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Verwaltungsverfahren wegen der Weiterentwicklung des Ausgleichsernergiepreis-Abrechnungssystems, 2012. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2012/BK6-12-024/BK6-12-024_Beschluss_2012_10_25_BF.pdf?__blob=publicationFile&v=6, Last accessed: 11 March 2021.

[60] European Energy Exchange (EEX). Contract specifications — release 0078b, May 2021. https://www.eex.com/fileadmin/EEX/Downloads/Rules/Contract_Specifications/20210506_EEX_Contract_Spezifications_0078b_E_D.zip.

[61] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Festlegung zu Verfahren zur Ausschreibung von Regelenergie in Gestalt der Primärregelung, insbesondere zu Mindestangebotsgrößen, Ausschreibungszeiträumen und Ausschreibungszeitscheiben, zum technisch notwendigen Anteil nach § 6 Abs. 2 StromNZV und zu den einheitlichen Bedingungen, die Anbieter von Regelenergie erfüllen müssen, sowie zu den zu veröffentlichenden Daten nach § 27 Abs. 2 StromNZV, 2010. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2010/BK6-10-097bis-099/BK6-10-097_Beschluss_2011_04_12.pdf?__blob=publicationFile&v=3, Last accessed: 11 March 2021.

[62] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Festlegung zu Verfahren zur Ausschreibung von Regelenergie in Gestalt der Sekundärregelung, insbesondere zu Mindestangebotsgrößen, Ausschreibungszeiträumen und Ausschreibungszeitscheiben, zum technisch notwendigen Anteil nach § 6 Abs. 2 StromNZV und zu den einheitlichen Bedingungen, die Anbieter von Regelenergie erfüllen müssen, sowie zu den zu veröffentlichenden Daten nach § 27 Abs. 2 StromNZV, 2010. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2010/BK6-10-098/BK6-10-098_Beschluss_Download.pdf?__blob=publicationFile&v=3, Last accessed: 11 March 2021.

[63] Ruling Chamber 6 of the Bundesnetzagentur (Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway). Verwaltungsverfahren wegen der Festlegung zu Verfahren zur Ausschreibung von Regelenergie in Gestalt der Minutenreserve, insbesondere zu Mindestangebotsgrößen, Ausschreibungszeiträumen und Ausschreibungszeitscheiben, zum technisch not-

wendigen Anteil nach § 6 Abs. 2 StromNZV, zu den einheitliche Bedingungen, die Anbieter von Regelenergie erfüllen müssen sowie zu den zu veröffentlichenden Daten nach § 27 Abs. 2 StromNZV und zum Einsatz von Regelenergie nach 27 Abs. 1 Nr. 3 StromNZV, 2010. https://www.bundesnetzagentur.de/DE/Beschlusskammern/1_GZ/BK6-GZ/2010/BK6-10-099/BK6-10-099_Beschluss_Download.pdf?__blob=publicationFile&v=3, Last accessed: 11 March 2021.

[64] Agora Energiewende. 12 Thesen zur Energiewende, November 2012. https://www.agora-energiewende.de/fileadmin/Projekte/2012/12-Thesen/Agora_12_Thesen_Langfassung_2.Auflage_web.pdf, Last accessed: 23 May 2021.

[65] German Meteorological Office. Press Conference: Wetterbedingte Risiken der Stromproduktion aus erneuerbaren Energien durch kombinierten Einsatz von Windkraft und Photovoltaik reduzieren, March 2018. https://www.dwd.de/DE/presse/pressekonferenzen/DE/2018/PK_06_03_2018/pressekonferenz.html?nn=509470, Last accessed: 23 May 2021.

[66] Katrin Schaber. *Integration of Variable Renewable Energies in the European power system: a model-based analysis of transmission grid extensions and energy sector coupling*. PhD thesis, TU München, 2014. http://hdl.handle.net/11858/00-001M-0000-0024-B6DB-5.

[67] Jed J. Cohen, Johannes Reichl, and Michael Schmidthaler. Refocussing research efforts on the public acceptance of energy infrastructure: A critical review. *Energy*, 76:4–9, 2014. doi:10.1016/j.energy.2013.12.056.

[68] Agora Energiewende. Energiewende und Dezentralität, February 2017. https://static.agora-energiewende.de/fileadmin/Projekte/2016/Dezentralitaet/Agora_Dezentralitaet_WEB.pdf, Last accessed: 24 May 2021.

[69] Paul L. Joskow. Capacity payments in imperfect electricity markets: Need and design. *Utilities Policy*, 16(3):159–170, September 2008. doi:10.1016/j.jup.2007.10.003.

[70] Fraunhofer ISE. Stromgestehungskosten Erneuerbare Energien, March 2018. https://www.ise.fraunhofer.de/content/dam/ise/de/documents/publications/studies/DE2018_ISE_Studie_Stromgestehungskosten_Erneuerbare_Energien.pdf, Last accessed: 24 May 2021.

[71] German Federal Network Agency. Genehmigung des Szenariorahmens 2021-2035, December 2020. https://www.netzentwicklungs

plan.de/sites/default/files/paragraphs-files/NEP_2030_2017_Bestaetigung.pdf.

[72] DIW (German Institute for Economic Research). Potentiale für Ökostrom in Deutschland- Verbraucherpräferenzen und Investitionsverhalten der Energieversorger, January 2012. https://diw-econ.de/en/wp-content/uploads/sites/2/2014/03/DIWecon_HSE_Oekostrom.pdf.

[73] Joachim Bagemihl, Frank Boesner, Jens Riesinger, Michael Künzli, Gwendolin Wilke, Gabriela Binder, Holger Wache, Daniel Laager, Jürgen Breit, Michael Wurzinger, Juliana Zapata, Silvia Ulli-Beer, Vincent Layec, Thomas Stadler, and Franz Stabauer. A market-based smart grid approach to increasing power grid capacity without physical grid expansion. *Computer Science - Research and Development*, September 2017. doi:10.1007/s00450-017-0356-5.

[74] B. P. Heard, B. W. Brook, T. M. L. Wigley, and C. J. A. Bradshaw. Burden of proof: A comprehensive review of the feasibility of 100% renewable-electricity systems. *Renewable and Sustainable Energy Reviews*, 76:1122–1133, September 2017. doi:10.1016/j.rser.2017.03.114.

[75] George E. P. Box. Science and statistics. *Journal of the American Statistical Association*, 71(356):791–799, December 1976. doi:10.1080/01621459.1976.10480949.

[76] T. W. Brown, T. Bischof-Niemz, K. Blok, C. Breyer, H. Lund, and B. V. Mathiesen. Response to 'burden of proof: A comprehensive review of the feasibility of 100% renewable-electricity systems'. *Renewable and Sustainable Energy Reviews*, 92:834–847, September 2018. doi:10.1016/j.rser.2018.04.113.

[77] Bernhard Ernst. Analysis of wind power ancillary services characteristics with german 250-mw wind data, December 1999. doi:10.2172/752651.

[78] J. P. Deane, G. Drayton, and B. P. Ó Gallachóir. The impact of sub-hourly modelling in power systems with significant levels of renewable generation. *Applied Energy*, 113:152–158, January 2014. doi:10.1016/j.apenergy.2013.07.027.

[79] Armen Der Kiureghian and Ove Ditlevsen. Aleatory or epistemic? does it matter? *Structural Safety*, 31(2):105–112, March 2009. doi:10.1016/j.strusafe.2008.06.020.

[80] Jan Abrell and Friedrich Kunz. Integrating intermittent renewable wind generation - a stochastic multi-market electricity model for

the european electricity market. *Networks and Spatial Economics*, 15 (1):117–147, December 2014. doi:10.1007/s11067-014-9272-4.

[81] B. Palmintier, E. Hale, T. M. Hansen, W. Jones, D. Biagioni, H. Sorensen, H. Wu, and B. M. Hodge. Igms: An integrated iso-to-appliance scale grid modeling system. *IEEE Transactions on Smart Grid*, 8(3):1525–1534, May 2017. doi:10.1109/TSG.2016.2604239.

[82] E. A. M. Klaassen, C. B. A. Kobus, J. Frunt, and J. G. Slootweg. Responsiveness of residential electricity demand to dynamic tariffs: Experiences from a large field test in the netherlands. *Applied Energy*, 183:1065–1074, December 2016. doi:10.1016/j.apenergy.2016.09.051.

[83] Murray Goulden, Ben Bedwell, Stefan Rennick-Egglestone, Tom Rodden, and Alexa Spence. Smart grids, smart users? the role of the user in demand side management. *Energy Research & Social Science*, 2:21–29, June 2014. doi:10.1016/j.erss.2014.04.008.

[84] Koen H. Dam, Igor Nikolic, and Zofia Lukszo, editors. *Agent-Based Modelling of Socio-Technical Systems*. Springer Netherlands, 2013. doi:10.1007/978-94-007-4933-7.

[85] Koen H. Dam. *Capturing socio-technical systems with agent-based modelling*. PhD thesis, Technische Universiteit Delft, 2009. `http://resolver.tudelft.nl/uuid:1b36bc02-c7fe-4773-9ab2-3daa3b891754`. Last accessed: 25 May 2021.

[86] Stefan Pfenninger, Lion Hirth, Ingmar Schlecht, Eva Schmid, Frauke Wiese, Tom Brown, Chris Davis, Matthew Gidden, Heidi Heinrichs, Clara Heuberger, Simon Hilpert, Uwe Krien, Carsten Matke, Arjuna Nebel, Robbie Morrison, Berit Müller, Guido Pleßmann, Matthias Reeg, Jörn C. Richstein, Abhishek Shivakumar, Iain Staffell, Tim Tröndle, and Clemens Wingenbach. Opening the black box of energy modelling: Strategies and lessons learned. *Energy Strategy Reviews*, 19:63–71, January 2018. doi:10.1016/j.esr.2017.12.002.

[87] Stefan Pfenninger, Joseph DeCarolis, Lion Hirth, Sylvain Quoilin, and Iain Staffell. The importance of open data and software: Is energy research lagging behind? *Energy Policy*, 101:211–215, February 2017. doi:10.1016/j.enpol.2016.11.046.

[88] Frauke Wiese. *renpass Renewable Energy Pathways Simulation System - Open Source as an approach to meet challenges in energy modeling*. PhD thesis, University of Flensburg, 2015. `https://backend.orbit.dtu.dk/ws/portalfiles/portal/127260304/Wiese2015_PhdThesis.pdf`. Last accessed: 23 May 2021.

[89] Markus Haller. *CO2 mitigation and power system integration of fluctuating renewable energy sources: A multi-scale modeling approach*. PhD thesis, Technische Universität München, 2012. doi:10.14279/depositonce-3138.

[90] Henrik Lund, Finn Arler, Poul Østergaard, Frede Hvelplund, David Connolly, Brian Mathiesen, and Peter Karnøe. Simulation versus optimisation: Theoretical positions in energy system modelling. *Energies*, 10(7):840, June 2017. doi:10.3390/en10070840.

[91] I. van Beuzekom, M. Gibescu, and J. G. Slootweg. A review of multi-energy system planning and optimization tools for sustainable urban development. In *2015 IEEE Eindhoven PowerTech*, pages 1–7, June 2015. doi:10.1109/PTC.2015.7232360.

[92] Frauke Wiese, Simon Hilpert, Cord Kaldemeyer, and Guido Pleßmann. A qualitative evaluation approach for energy system modelling frameworks. *Energy, Sustainability and Society*, 8(1), April 2018. doi:10.1186/s13705-018-0154-3.

[93] Per-Anders Bergendahl and Clas Bergström. Long-term oil substitution—the iea-markal model and some simulation results for sweden. In Lars Matthiessen, editor, *The Impact of Rising Oil Prices on the World Economy*, pages 97–112. Palgrave Macmillan UK, London, 1982. doi:10.1007/978-1-349-06361-1_7.

[94] Ad J. Seebregts, Gary A. Goldstein, and Koen Smekens. Energy/environmental modeling with the markal family of models. In Peter Chamoni, Rainer Leisten, Alexander Martin, Joachim Minnemann, and Hartmut Stadtler, editors, *Operations Research Proceedings 2001: Selected Papers of the International Conference on Operations Research (OR 2001) Duisburg, September 3–5, 2001*, pages 75–82. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. doi:10.1007/978-3-642-50282-8_10.

[95] International Energy Agency. A comparison of the times and markal models. `http://iea-etsap.org/tools/TIMESVsMARKAL.pdf`, Last accessed: 23 May 2021.

[96] E3MLab. Primes model version 6, 2016–2017, detailed model description, 2016. `http://www.e3mlab.eu/e3mlab/PRIMES%20Manual/The%20PRIMES%20MODEL%202016-7.pdf`, Last accessed: 23 May 2021.

[97] Max Jacob Steinberg and Smith Theodore Hunter Smith. *Economy Loading of Power Plants and Electric Systems*. John Wiley & Sons, 1943.

[98] Georg R. Davison. Dividing load between units. *Electrical World*, 80(26):1385–1387, December 1922.

[99] Allen J. Wood, Bruce F. Wollenberg, and Gerald B. Sheblé. *Power generation, operation, and control, 3rd Edition*. John Wiley & Sons, 2013. ISBN 978-0-471-79055-6.

[100] M. Carrion and J. M. Arroyo. A computationally efficient mixed-integer linear formulation for the thermal unit commitment problem. *IEEE Transactions on Power Systems*, 21(3):1371–1378, August 2006. doi:10.1109/TPWRS.2006.876672.

[101] Karl-Kiên Cao, Ambros Gleixner, and Matthias Miltenberger. Methoden zur Reduktion der Rechenzeit linearer Optimierungsmodelle in der Energiewirtschaft – Eine Performance-Analyse. In *14. Symposium Energieinnovation*, February 2016, Last accessed: 23 May 2021.

[102] Johannes Dorfner. *Open Source Modelling and Optimisation of Energy Infrastructure at Urban Scale*. PhD thesis, Technische Universität München, München, 2016. `https://mediatum.ub.tum.de/doc/1285570/1285570.pdf`. Last accessed: 23 May 2021.

[103] P. Ahlhaus and P. Stursberg. Transmission capacity expansion: An improved transport model. In *IEEE PES ISGT Europe 2013*, pages 1–5, October 2013. doi:10.1109/ISGTEurope.2013.6695322.

[104] Hans Christian Gils, Yvonne Scholz, Thomas Pregger, Diego Luca de Tena, and Dominik Heide. Integrated modelling of variable renewable energy-based power supply in europe. *Energy*, 123 (Supplement C):173–188, 2017. doi:10.1016/j.energy.2017.01.115.

[105] Yvonne Scholz. *Renewable energy based electricity supply at low costs: development of the REMix model and application for Europe*. PhD thesis, Universität Stuttgart, 2012. doi:10.18419/opus-2015.

[106] Eric Guerci, Mohammad Ali Rastegar, and Silvano Cincotti. Agent-based modeling and simulation of competitivewholesale electricity markets. In Steffen Rebennack, editor, *Handbook of Power Systems II*, chapter 2, pages 241–286. Springer Berlin Heidelberg, Heidelberg, 2010. doi:10.1007/978-3-642-12686-4.

[107] Anke Weidlich and Daniel Veit. A critical survey of agent-based wholesale electricity market models. *Energy Economics*, 30(4):1728 – 1759, 2008. doi:10.1016/j.eneco.2008.01.003.

[108] Philipp Ringler, Dogan Keles, and Wolf Fichtner. Agent-based modelling and simulation of smart electricity grids and markets

– a literature review. *Renewable and Sustainable Energy Reviews*, 57 (Supplement C):205–215, 2016. doi:10.1016/j.rser.2015.12.169.

[109] Junjie Sun and Leigh Tesfatsion. Dynamic testing of wholesale power market designs: An open-source agent-based framework. *Computational Economics*, 30(3):291–327, October 2007. doi:10.1007/s10614-007-9095-1.

[110] Daniel J. Veit, Anke Weidlich, and Jacob A. Krafft. An agent-based analysis of the german electricity market with transmission capacity constraints. *Energy Policy*, 37(10):4132–4144, 2009. doi:10.1016/j.enpol.2009.05.023.

[111] Ido Erev and Alvin E. Roth. Predicting how people play games: Reinforcement learning in experimental games with unique, mixed strategy equilibria. *The American Economic Review*, 88(4):848–881, 1998.

[112] Bundeskartellamt (German Federal Cartel Office). Sektoruntersuchung Stromerzeugung und -großhandel, Abschlussbericht gemäß § 32e GWB, January 2011. `https://www.bundeskartellamt.de/SharedDocs/Publikation/DE/Sektoruntersuchungen/Sektoruntersuchung%20Stromerzeugung%20Stromgrosshandel%20-%20Abschlussbericht.pdf?__blob=publicationFile&v=3`, Last accessed: 23 May 2021.

[113] G. Conzelmann, G. Boyd, V. Koritarov, and T. Veselka. Multi-agent power market simulation using EMCAS. In *IEEE Power Engineering Society General Meeting, 2005*. IEEE, 2005. doi:10.1109/pes.2005.1489271.

[114] David P. Chassin, Jason C. Fuller, and Ned Djilali. GridLAB-d: An agent-based simulation framework for smart grids. *Journal of Applied Mathematics*, 2014:1–12, 2014. doi:10.1155/2014/492320.

[115] Göran Andersson. Power system analysis — power flow analysis, fault analysis, power system dynamics and stability, September 2012. `http://research.iaun.ac.ir/pd/bahador.fani/pdfs/UploadFile_6990.pdf`, Last accessed: 23 May 2021.

[116] Richard Marenbach, Dieter Nelles, and Christian Tuttas. *Elektrische Energietechnik*. Springer Fachmedien Wiesbaden, 2013. doi:10.1007/978-3-8348-2190-4.

[117] "Modeling Language", Wikipedia.org. `https://en.wikipedia.org/wiki/Modeling_language`. accessed: 07 Mai 2018.

[118] Felix Felgner, Josef Meiers, Lukas Exel, and G. Frey. Design of distributed energy systems: Role and requirements of modeling and simulation. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*. IEEE, September 2014. doi:10.1109/etfa.2014.7005305.

[119] Atiyah Elsheikh, Edmund Widl, and Peter Palensky. Simulating complex energy systems with modelica: A primary evaluation. In *2012 6th IEEE International Conference on Digital Ecosystems and Technologies (DEST)*. IEEE, June 2012. doi:10.1109/dest.2012.6227921.

[120] Modelica Association. Modelica - A Unified Object-Oriented Language for Systems Modeling, Language Specification Version 3.4, 2017. `https://www.modelica.org/documents/ModelicaSpec34.pdf`, Last accessed: 14 March 2021.

[121] Liu Liu. *Object-oriented Modeling and Efficient Simulation of C^3-Systems*. PhD thesis, Universität des Saarlandes, 2014.

[122] Jens Frenkel. *Entwicklung eines Modelica Compiler BackEnds für große Modelle*. PhD thesis, Technische Universität Dresden, 2013. `https://nbn-resolving.org/urn:nbn:de:bsz:14-qucosa-132032`.

[123] Willi Braun, Francesco Casella, and Bernhard Bachmann. Solving large-scale modelica models: new approaches and experimental results using OpenModelica. In *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17*. Linköping University Electronic Press, July 2017. doi:10.3384/ecp17132557.

[124] Guiyou Mao and Linda R. Petzold. Efficient integration over discontinuities for differential-algebraic systems. *Computers & Mathematics with Applications*, 43(1):65–79, 2002.

[125] I. S. Duff and J. K. Reid. An implementation of tarjan's algorithm for the block triangularization of a matrix. *ACM Trans. Math. Softw.*, 4(2):137–147, June 1978. doi:10.1145/355780.355785.

[126] Constantinos C. Pantelides. The consistent initialization of differential-algebraic systems. *SIAM Journal on Scientific and Statistical Computing*, 9(2):213–231, March 1988. doi:10.1137/0909014.

[127] Jens Frenkel, Christian Schubert, Günter Kunze, Peter Fritzson, Martin Sjölund, and Adrian Pop. Towards a benchmark suite for modelica compilers: Large models. In *Proceedings from the 8th International Modelica Conference, Dresden, Germany*. Linköping University Electronic Press, June 2011. doi:10.3384/ecp11063143.

[128] Francesco Casella. Simulation of large-scale models in modelica: State of the art and future perspectives. In *Proceedings of the 11th International Modelica Conference, Versailles, France, September 21-23*. Linköping University Electronic Press, September 2015. doi:10.3384/ecp15118459.

[129] P. Palensky, E. Widl, and A. Elsheikh. Simulating cyber-physical energy systems: Challenges, tools and methods. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(3):318–326, March 2014. doi:10.1109/TSMCC.2013.2265739.

[130] Per Sahlin and Pavel Grozman. IDA simulation environment - a tool for modelica based end-user application deployment. In *3-rd International Modelica Conference, Linköping, Sweden, November 3–4, 2003*, pages 105–114, 2003.

[131] Dirk Zimmer. Module-preserving compilation of modelica models. In *Proceedings of the 7th International Modelica Conference, Como, Italy*. Linköping University Electronic Press, October 2009. doi:10.3384/ecp09430028.

[132] Francesco Casella, Alberto Leva, and Andrea Bartolini. Simulation of large grids in OpenModelica: reflections and perspectives. In *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17*. Linköping University Electronic Press, July 2017. doi:10.3384/ecp17132227.

[133] François E. Cellier. World3 in modelica: Creating system dynamics models in the modelica framework. In *Proceedings of the 6th International Modelica Conference, Bielefeld, Germany, March 3-4*, pages 393–400, March 2008.

[134] Donella H. Meadows, Dennis L. Meadows, Jørgen Randers, and William W. Behrens III. *The limits to growth: A report for the Club of Rome's project on the predicament of mankind*. Univers Books, New York, 1972.

[135] Matthis Thorade, Jörg Rädler, Peter Remmen, Tobias Maile, Reinhard Wimmer, Jun Cao, Moritz Lauster, Christoph Nytsch-Geusen, Dirk Müller, and Christoph van Treeck. An open toolchain for generating modelica code from building information models. In *Proceedings of the 11th International Modelica Conference, Versailles, France, September 21-23*. Linköping University Electronic Press, September 2015. doi:10.3384/ecp15118383.

[136] Technische Universität Hamburg-Harburg. Website of TransientEE project. `https://www.tuhh.de/transient-ee/`, Last accessed: 23 May 2021.

[137] Pascal Dubucq and Günter Ackermann. Optimal use of energy storage potentials in a renewable energy system with district heating. *Energy Procedia*, 135:158–171, October 2017. doi:10.1016/j.egypro.2017.09.499.

[138] F. J. Gómez, L. Vanfretti, and S. H. Olsen. Binding cim and modelica for consistent power system dynamic model exchange and simulation. In *2015 IEEE Power Energy Society General Meeting*, pages 1–5, July 2015. doi:10.1109/pesgm.2015.7286434.

[139] I. Nikolic and J. Kasmire. Theory. In Koen H. Dam, Igor Nikolic, and Zofia Lukszo, editors, *Agent-Based Modelling of Socio-Technical Systems*, chapter 2, pages 11–72. Springer Netherlands, 2013.

[140] Volker Grimm and Steven F. Railsback. *Individual-based Modeling and Ecology*. Princeton University Press, 2005. doi:10.1515/9781400850624.

[141] Graeme D. Ruxton. Effects of the spatial and temporal ordering of events on the behaviour of a simple cellular automaton. *Ecological Modelling*, 84(1-3):311–314, January 1996. doi:10.1016/0304-3800(94)00145-6.

[142] Victorino Sanz, Federico Bergero, and Alfonso Urquia. An approach to agent-based modeling with modelica. *Simulation Modelling Practice and Theory*, 83:65–74, April 2018. doi:10.1016/j.simpat.2017.12.012.

[143] Ana Constantin, Artur Löwen, Ferdinanda Ponci, Kristian Huchtemann, and Dirk Müller. Dymola-JADE co-simulation for agent-based control in office spaces. In *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17*. Linköping University Electronic Press, July 2017. doi:10.3384/ecp17132345.

[144] Felix Bünning, Roozbeh Sangi, and Dirk Müller. A modelica library for the agent-based control of building energy systems. *Applied Energy*, 193:52–59, May 2017. doi:10.1016/j.apenergy.2017.01.053.

[145] Lukas Exel, Felix Felgner, and Georg Frey. Multi-domain modeling of distributed energy systems - the MOCES approach. In *2015 IEEE International Conference on Smart Grid Communications (SmartGridComm)*. IEEE, November 2015. doi:10.1109/smartgridcomm.2015.7436395.

[146] Lukas Exel and Georg Frey. Modeling and simulation of local flexibilities and their effect to the entire power system. *Computer*

Science - Research and Development, 33(1-2):49–60, September 2017. doi:10.1007/s00450-017-0346-7.

[147] Lukas Exel, Georg Frey, Gerrit Wolf, and Mathias Oppelt. Re-use of existing simulation models for DCS engineering via the functional mock-up interface. In *Proceedings of the 2014 IEEE Emerging Technology and Factory Automation (ETFA)*. IEEE, September 2014. doi:10.1109/etfa.2014.7005261.

[148] Marco Nesarajah, Lukas Exel, and Georg Frey. Modelica® library for dynamic simulation of thermoelectric generators. In *Proceedings of the 11th European Conference on Thermoelectrics*, pages 213–217. Springer International Publishing, 2014. doi:10.1007/978-3-319-07332-3_25.

[149] Felix Felgner, Lukas Exel, and Georg Frey. Model-based design and validation of waste heat recovery systems. In *2012 IEEE International Energy Conference and Exhibition (ENERGYCON)*. IEEE, September 2012. doi:10.1109/energycon.2012.6347764.

[150] Felix Felgner, Lukas Exel, and Georg Frey. Component-oriented ORC plant modeling for efficient system design and profitability prediction. In *2011 International Conference on Clean Electrical Power (ICCEP)*. IEEE, June 2011. doi:10.1109/iccep.2011.6036274.

[151] Liu Liu, F. Felgner, and G. Frey. Comparison of 4 numerical solvers for stiff and hybrid systems simulation. In *Proceedings of the 2010 IEEE Emerging Technology and Factory Automation (ETFA)*, pages 1–8, September 2010. doi:10.1109/ETFA.2010.5641330.

[152] “Time”. `https://en.wikipedia.org/wiki/time`. accessed: 16 March 2018.

[153] Ibrahim Reda and Afshin Andreas. Solar position algorithm for solar radiation applications, January 2008. doi:10.2172/15003974.

[154] Destatis (Federal Statistical Office of Germany). Data in energy price trends – long-time series to march 2021, March 2021. `https://www.destatis.de/EN/Themes/Economy/Prices/Publications/Downloads-Energy-Price-Trends/energy-price-trends-pdf-5619002.html`, Last accessed: 23 May 2021.

[155] ENTSO-E. Data portal. `https://www.entsoe.eu/data/data-portal/`, Last accessed: 23 May 2021.

[156] German Transmission Service Operators. Information plattform of the german transmission service operators. `https://www.netztransparenz.de`. Last accessed: 27 June 2016.

[157] Håkan Lundvall, Peter Fritzson, and Bernhard Bachmann. Event handling in the openmodelica compiler and runtime system. *Technical reports in Computer and Information Science*, (2):21, September 2008. Last accessed: 26 May 2021.

[158] Michel Berkelaar, Kjell Eikland, and Peter Notebaert. lpsolve, open source (mixed-integer) linear programming system, 2004. `https://sourceforge.net/projects/lpsolve/`, Last accessed: 25 May 2021.

[159] Modelica standard library, 2021. `https://github.com/modelica/ModelicaStandardLibrary`, Last accessed: 14 March 2021.

[160] Andreas Pfeiffer. Optimization library for interactive multi-criteria optimization tasks. In *Proceedings of the 9th International Modelica Conference, Munich, Germany, September 3-5*, pages 669–679. Linköping University Electronic Press, 2012.

[161] Jonas Egerer. Open source electricity model for germany (elmod-de), 3 2016. `http://www.diw.de/elmod`, Last accessed: 26 May 2021.

[162] Jonas Egerer, Clemens Gerbaulet, Richard Ihlenburg, Friedrich Kunz, Benjamin Reinhard, Christian von Hirschhausen, Alexander Weber, and Jens Weibezahn. Data documentation 72: Data documentation and applications to the german and european electricity markets, 2014. `https://www.diw.de/documents/publikationen/73/diw_01.c.528927.de/diw_datadoc_2016-083.pdf`, Last accessed: 26 May 2021.

[163] Jonas Egerer and Wolf-Peter Schill. Power system transformation toward renewables: Investment scenarios for germany. *Economics of Energy & Environmental Policy*, 3(2), April 2014. doi:10.5547/2160-5890.3.2.jege.

[164] Jonas Egerer, Jens Weibezahn, and Hauke Hermann. Two price zones for the german electricity market — market implications and distributional effects. *Energy Economics*, 59:365–381, September 2016. doi:10.1016/j.eneco.2016.08.002.

[165] Enercon product overview, 2015. `https://www.enercon.de/fileadmin/Redakteur/Medien-Portal/broschueren/pdf/en/ENERCON_Produkt_en_06_2015.pdf`, Last accessed: 26 May 2021.

[166] Gaël Guennebaud, Benoît Jacob, et al. Eigen v3, 2018. `https://eigen.tuxfamily.org`, Last accessed: 26 May 2021.

[167] Christoph Nytsch-Geusen, Alexander Inderfurth, Werne Kaul, Katharina Mucha, Jörg Rädler, Matthis Thorade, and Carles Ribas Tugores. Template based code generation of modelica building energy simulation models. In *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic, May 15-17*. Linköping University Electronic Press, July 2017. doi:10.3384/ecp17132199.

[168] Frank Sensfuß, Mario Ragwitz, and Massimo Genoese. The merit-order effect: A detailed analysis of the price effect of renewable electricity generation on spot market prices in germany. *Energy Policy*, 36(8):3086–3094, 2008. doi:10.1016/j.enpol.2008.03.035.

[169] Lion Hirth. The market value of variable renewables: The effect of solar wind power variability on their relative price. *Energy Economics*, 38:218–236, 2013. doi:10.1016/j.eneco.2013.02.004.

[170] Xianda Deng, Hongyu Li, Wenpeng Yu, Wang Weikang, and Yilu Liu. Frequency observations and statistic analysis of worldwide main power grids using FNET/GridEye. In *2019 IEEE Power & Energy Society General Meeting (PESGM)*. IEEE, 8 2019. doi:10.1109/pesgm40551.2019.8973560.

[171] Johannes Weniger, Tjarko Tjaden, and Volker Quaschning. Sizing of residential PV battery systems. *Energy Procedia*, 46:78–87, 2014. doi:10.1016/j.egypro.2014.01.160.

[172] Dirk Zimmer. *Equation-based modeling of variable-structure systems*. PhD thesis, Eidgenössische Technische Hochschule (ETH) Zürich, 2010. doi:10.3929/ethz-a-006053740.

[173] Modelica Association. Modelica - A Unified Object-Oriented Language for Systems Modeling, Language Specification Version 3.5, 2021. `https://specification.modelica.org/maint/3.5/MLS.pdf`, Last accessed: 14 March 2021.

[174] Linda R. Petzold. A description of dassl: A differential/algebraic system solver, September 1982. `https://www.osti.gov/servlets/purl/5882821/`, Last accessed: 23 May 2021.

[175] "Imbalance Settlement Period", COMMISSION REGULATION (EU) 2017/2195. `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32017R2195`. accessed: 6 April 2018.

[176] Volker Mehrmann. Index concepts for differential-algebraic equations, 2012. `ftp://ftp.math.tu-berlin.de/pub/Preprints/numerik/Meh11.pdf`, Last accessed: 21 March 2021.

[177] Igor L. Pioro and R. Panchal. Appendix a7 - supplementary tables. In Igor L. Pioro, editor, *Handbook of Generation IV Nuclear Reactors*, Woodhead Publishing Series in Energy, pages 859–869. Woodhead Publishing, 2016. doi:10.1016/B978-0-08-100149-3.15007-4.

[178] Alan. C. Hindmarsh and Linda R. Petzold. Lsodar, ordinary differential equation solver for stiff or non-stiff system with root-finding. http://www.oecd-nea.org/tools/abstract/detail/uscd1228, Last accessed: 26 May 2021.

[179] "Market Area", ENTSO-E. https://www.entsoe.eu/data/data-portal/glossary/Pages/home.aspx. accessed: 3 April 2018.

[180] "Market Balance Area", ENTSO-E. https://www.entsoe.eu/data/data-portal/glossary/Pages/home.aspx. accessed: 28 March 2018.

[181] "market rules", E-Control. https://www.e-control.at/marktteilnehmer/strom/marktregeln. accessed: 21 May 2018.

[182] "Primary Energy", Wikipedia.org. https://en.wikipedia.org/wiki/Primary_energy. accessed: 19 March 2018.

[183] Roland Priemer. *Introductory Signal Processing*. World Scientific, November 1990. doi:10.1142/0864.

[184] Richard T. Snodgrass. *Developing time-oriented database applications in SQL*. Morgan Kaufmann Publishers, 2000. https://www2.cs.arizona.edu/~rts/tdbbook.pdf.

[185] Moritz Stüber, Lukas Exel, and Georg Frey. Using modelling and simulation as a service (MSaaS) for facilitating flexibility-based optimal operation of distribution grids. In *Proceedings of the 15th International Conference on Informatics in Control, Automation and Robotics*. SCITEPRESS - Science and Technology Publications, 2018. doi:10.5220/0006899106230630.

[186] Lukas Exel and Georg Frey. Toward a decentralized forecast system for distributed power generation. In *2014 IEEE International Energy Conference (ENERGYCON)*. IEEE, May 2014. doi:10.1109/energycon.2014.6850577.

[187] Felix Felgner, Lukas Exel, Marco Nesarajah, and Georg Frey. Component-oriented modeling of thermoelectric devices for energy system design. *IEEE Transactions on Industrial Electronics*, 61 (3):1301–1310, March 2014. doi:10.1109/tie.2013.2261037.