Université Fédérale
Toulouse Midi-Pyrénées

THÈSE

En vue de l'obtention du
DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE
Délivré par l'Université Toulouse 3 - Paul Sabatier

Présentée et soutenue par

Antoine BOCHE

Le 21 juillet 2023

Étude de la durabilité des micro-réseaux par un modèle systémique basé sur la théorie de la modélisation et de la simulation

---

Study of the sustainability of microgrids using systemic models based on the theory of modeling and simulation

---

Ecole doctorale : **GEETS - Génie Electrique Electronique,Télécommunications et Santé : du système au nanosystème**

Spécialité : **Génie Electrique**

Unité de recherche :
LAAS - Laboratoire d'Analyse et d'Architecture des Systèmes

Thèse dirigée par
Luiz LAVADO- VILLA et Clément FOUCHER

Jury

Mme Delphine RIU, Rapporteure
M. Thomas HAMACHER, Rapporteur
M. Simon MEUNIER, Examinateur
M. Luiz LAVADO- VILLA, Directeur de thèse

Contents

Résumé 5

Abstract 6

List of Figures 9

List of Tables 10

General Introduction 11

1 The Source System: The basis of modeling and microgrids issues 13
1.1 M&S theory: The basics . . . 14
1.1.1 What is a source system? . . . 14
1.1.2 Basic system modeling concepts . . . 14
1.2 Microgrid: Problem statement and thesis objectives . . . 16
1.2.1 Thesis context: Energy access and microgrids . . . 16
1.2.2 Thesis object of study: microgrid and its sustainability . . . 18
1.2.3 Problem statement . . . 19
1.2.4 Thesis motivation: Contributing to sustainable electricity access in rural areas 19
1.2.5 Thesis methodology: Systemic study, model and simulator . . . 19

2 Experimental frame: Describing microgrid systems and their sustainability 21
2.1 M&S Theory: Framing the system . . . 22
2.1.1 What is an experimental frame? . . . 22
2.1.2 Systems in the vision of MS theory . . . 22
2.1.3 Relations between models and experimental frames . . . 24
2.1.4 M&S theory in this chapter . . . 25
2.2 Microgrids: Systemic analysis . . . 26
2.2.1 Towards a more comprehensive understanding of microgrid systems . . . 26
2.2.2 Using the state of the art as joint experimental frame . . . 27
2.2.3 Energy Field . . . 27
2.2.4 Information Field . . . 30
2.2.5 Financial Field . . . 35
2.2.6 Social Field . . . 38
2.2.7 Framing sustainability in microgrids . . . 41
2.2.8 Conclusions from the proposed experimental framework . . . 42

3 Model: Formalisms and their microgrid applications 45
3.1 M&S Theory: Modelling formalism . . . 46
3.1.1 What are models? . . . 46
3.1.2 More on time and trajectories . . . 46
3.1.3 More on System specifications - from data to system . . . 46
3.1.4 More on System specifications – component coupling . . . 47
3.1.5 Modeling formalisms . . . 49
3.1.6 The DEVS formalism . . . 50

3.1.7 The PDEVS atomic model . . . 50
3.1.8 The PDEVS coupled model . . . 51
3.1.9 Examples of DEVS-based models . . . 52
3.2 Microgrid: From the structure to the I/O functions . . . 57
3.2.1 Structural model: Assembling fields . . . 57
3.2.2 Generative Model: Assembling components . . . 72
3.2.3 Replicative model: Assembling I/O functions and states into a co-simulation 102

4 Simulator: Formalism and their application for the proposed models 117
4.1 M&S Theory: Simulator formalism . . . 118
4.1.1 What is a simulator? . . . 118
4.1.2 General simulator concepts . . . 118
4.1.3 Spreadsheet-based simulator . . . 119
4.1.4 PDEVS-based simulator: The choice of PyPDEVS . . . 122
4.1.5 DEVS bus . . . 126
4.2 Microgrid: Simulating the proposed models . . . 127
4.2.1 First simulator . . . 127
4.2.2 Component Based Simulator . . . 137
4.2.3 Co-simulation based simulator . . . 141

Conclusion and Future Work 147
Conclusion . . . 147
Future work . . . 151

Résumé

Les nations unies estiment qu'environ 789 millions de personnes vivent sans accès à l'électricité, et qu'en l'absence d'une action plus déterminée, 650 millions de personnes resteront sans accès à l'énergie en 2030. Cependant, les technologies basées sur les énergies renouvelables, qui permettent de fournir une électricité propre et sûre, sont devenues plus accessibles et moins coûteuses que jamais.

Malgré ces progrès prometteurs, le défi de l'électrification rurale durable et abordable reste complexe et risqué. L'étude des micro-réseaux en tant que moyen de surmonter le manque d'accès à l'énergie suscite un intérêt grandissant parmi les chercheurs. Les micro-réseaux peuvent constituer la clé pour assurer l'accès à l'énergie à l'échelle mondiale en raison de leur flexibilité et efficacité. Toutefois, la mise en œuvre durable et résiliente de ces micro-réseaux reste difficile sans un consensus clair sur les causes de ces échecs.

Nous suggérons donc d'aborder ce problème de manière différente en adoptant une approche plus formelle pour la représentation des micro-réseaux. Au fil de ce manuscrit, nous adopterons la méthodologie établie par la théorie de la modélisation et de la simulation de Ziegler pour élaborer un modèle de micro-réseau capable de représenter les défis de la durabilité de ses systèmes. Les termes de modélisation et de simulation, souvent définis de manière imprécise dans les pratiques courantes, ont une signification précise dans le cadre de ce travail. Afin de comprendre les fondements de ces deux termes, ce travail guidera le lecteur dans le développement et la compréhension de la théorie de la modélisation et de la simulation tout en les appliquant à la compréhension de la durabilité des micro-réseaux.

Une des motivations principales de cette thèse est de contribuer à la compréhension de la durabilité des micro-réseaux. Cette contribution est basée sur l'exploration de différentes méthodes pour élaborer des modèles qui intègrent les aspects énergétiques et communautaires des micro-réseaux. La littérature manque de modèles compatibles entre ces deux aspects, ce qui devrait apporter des voies de corrections par la contribution de notre travail à la communauté scientifique.

Afin d'atteindre cet objectif, nous avons précisé notre système source pour montrer la complexité et la diversité des micro-réseaux. En combinant les problèmes identifiés dans les micro-réseaux à la définition formelle d'un système source, nous proposons l'élaboration d'un formalisme systémique pour l'étude de ces systèmes. Le manuscrit présente un cadre expérimental qui combine les perspectives des différentes approches de la littérature sur les micro-réseaux dans un seul morphisme, étudiant les domaines clés des micro-réseaux : l'énergie, l'information, le financement et les aspects sociaux. À partir de ce cadre, trois modèles différents seront proposés, formalisés et simulés de manière itérative tout au long du manuscrit afin d'étudier les caractéristiques internes des modèles et d'approfondir à chaque étape la formalisation de la durabilité des micro-réseaux.

MOTS CLEFS : micro-réseau, Théorie de la modélisation et de la simulation, modélisation, simulation, approche systémique, durabilité, résilience

Abstract

The United Nations reports that approximately 789 million people live without access to electricity and that without more determined action, 650 million people will remain without access to energy by 2030. However, renewable energy technologies, which can provide clean and safe electricity, have become more accessible and are cheaper than ever.

Despite these promising advances, the challenge of sustainable and affordable rural electrification remains complex and risky. The study of microgrids as a means of overcoming the lack of access to energy is attracting growing interest among researchers. Microgrids hold the key to achieving global energy access due to their flexibility, efficiency, and reliability. However, sustainable and resilient implementation of these microgrids is only possible with a clear consensus on the causes of these failures.

We therefore suggest a different approach to this problem by adopting a more formal approach to the representation of microgrids. Throughout this manuscript, we will adopt the methodology established by Ziegler's modeling and simulation theory to develop a microgrid model capable of representing the sustainability challenges of its systems. The terms Modeling and Simulation, often loosely defined in common practice, have a precise meaning in this work. In order to understand the foundations of these two terms, this work will guide the reader through the development and understanding of modeling and simulation theory while applying them to the understanding of microgrid sustainability.

One of the main motivations of this thesis is to contribute to understanding microgrids' sustainability. This contribution is based on exploring different methods to develop models that integrate microgrids' energy and community aspects. The literature lacks models compatible with these two aspects, which should provide avenues for correction through the contribution of our work to the scientific community.

In order to achieve this objective, we have specified our source system to show the complexity and diversity of microgrids. By combining the problems identified in microgrids with the formal definition of a source system, we propose the development of a systemic formalism for studying these systems. The manuscript presents an experimental framework that combines the perspectives of different approaches in the microgrid literature into a single morphism, studying the critical domains of microgrids: energy, information, financing, and social aspects. Based on this framework, three different models will be proposed, formalized, and simulated iteratively throughout the manuscript to study the models' internal characteristics and deepen the formalization of microgrid sustainability at each step.

KEYWORDS: microgrid, modeling and simulation theory, modeling, simulation, systemic approach, sustainability, resilience

List of Figures

1 Core entities in modeling and simulation and their relationship related to the manuscript chapters . . . 11

1.1 Scope of the Chapter 1 . . . 13
1.2 The energy microgrid . . . 16
1.3 Evolution of usage and architecture with the grid power adapted from [1] . . . 17
1.4 The community microgrid . . . 17
1.5 microgrid-space: the conditions where microgrids are competitives in comparison with national grid extension and solar home systems [2] . . . 18
1.6 Evolution of Sustainability and Resilient microgrid publications in the literature . . . 18

2.1 Scope of the Chapter 2 . . . 21
2.2 A basic I/O system . . . 22
2.3 The experimental frames related to the intersection of *energy* and *community* microgrids . . . 26
2.4 Sustainability and Resilience microgrid publications related to the field they study 27
2.5 Representation of the different energy architectures . . . 28
2.6 Relationship between data and control for different layers . . . 31
2.7 Representation of the different information architectures . . . 31
2.8 Classification of communication technology for microgrid inspired from [3] . . . 35
2.9 Representation of the different financial architectures . . . 36
2.10 Representation of the different social architectures . . . 39
2.11 Structure of the review . . . 41

3.1 Scope of the Chapter 3 . . . 45
3.2 Model transformation layers within this part . . . 46
3.3 Detailed representation of a system with the contribution from each level of specification . . . 47
3.4 A multi-component system . . . 48
3.5 A network of systems . . . 48
3.6 The different time formalism . . . 49
3.7 A DEVS-based model . . . 50
3.8 Graphical representation of PDEVS . . . 51
3.9 Chronogram of the behaviour of the *State_variable* value . . . 52
3.10 A graphical representation of DEVS model state dynamic . . . 53
3.11 Coupling of the three example models . . . 54
3.12 Chronogram of the behaviour of the *State_variable* of each example model . . . 55
3.13 Diagram sequential . . . 56
3.14 Structural model step 1 - Laying out the fields . . . 57
3.15 Structural model step 2 - Energy and information exchanges . . . 58
3.16 Structural model step 3 - Energy and financial exchanges . . . 58
3.17 Structural model step 4 - Information and financial exchanges . . . 59
3.18 Structural model step 5 - Social-to-X exchanges . . . 59
3.19 Full structural model with all the atomic components represented . . . 61
3.20 View of the fluxes with its inputs, outputs and state . . . 63

3.21 View of the exchanges with its inputs, outputs and state . . . 64
3.22 Graphical representation of the flux model states and transition functions . . . 64
3.23 Chronogram of the behaviour of *SoH* in the flow model . . . 65
3.24 Graphical representation of the exchange model states and transition functions . . . 67
3.25 Chronogram of the behaviour of *SoH* in the exchange model . . . 67
3.26 The sequential diagram of the full generative model . . . 71
3.27 Components of the generative model . . . 72
3.28 Component model representation example . . . 73
3.29 Components of the generative model with their I/O connected together with only the flows . . . 73
3.30 Components of the generative model with their I/O connected together adding the exchanges . . . 74
3.31 Graphical representation of the power source S1 model states and dynamics . . . 75
3.32 Chronogram of the behaviour of *Prod* in the generator model . . . 75
3.33 Graphical representation of the load L1 model states and dynamics . . . 77
3.34 Chronogram of the behaviour of *Cons* in the load model . . . 77
3.35 Graphical representation of the battery B1 model states and dynamics . . . 78
3.36 Chronogram of the behaviour of *Capa* in the battery model . . . 79
3.37 Graphical representation of the power converter C1 model states and dynamics . . . 80
3.38 Chronogram of the behaviour of $Ctrl_cons$ in the power converter model . . . 81
3.39 Graphical representation of the wallet W1 model states and dynamics . . . 83
3.40 Chronogram of the behaviour of $Account_1$ in the wallet 1 model . . . 83
3.41 Graphical representation of the wallet W2 model states and dynamics . . . 84
3.42 Chronogram of the behaviour of $Account_2$ and *Payment* in the wallet 2 model . . . 85
3.43 Graphical representation of the bank B1 model states and dynamics . . . 86
3.44 Chronogram of the behaviour of *Bill* and *Pay* in the financial institution model . . . 87
3.45 Graphical representation of the measurement M1 model states and dynamics . . . 88
3.46 Chronogram of the behaviour of *Meas_cons* in the measure 1 model . . . 89
3.47 Graphical representation of the measure M2 model states and dynamics . . . 90
3.48 Chronogram of the behaviour of *Meas_wallet* in the measure 2 model . . . 90
3.49 Graphical representation of the electronic device D1 model states and dynamics . . . 91
3.50 Chronogram of the behaviour of *Command_bill* in the electronic device 1 model . . . 92
3.51 Graphical representation of the electronic device D2 model states and dynamics . . . 93
3.52 Chronogram of the behaviour of *Command_D2* in the electronic device 2 model . . . 94
3.53 Graphical representation of the Acceptance A1 model states and dynamics . . . 95
3.54 Chronogram of the behaviour of *Acce* and *Last_state* in the acceptance 1 model . . . 96
3.55 Graphical representation of the Acceptance A2 model states and dynamics . . . 97
3.56 Chronogram of the behaviour of *Acce* in the acceptance 2 model . . . 98
3.57 The sequential diagram of the full generative model . . . 100
3.58 Desired co-simulation systemic microgrid model using the four main fields . . . 102
3.59 Modeled co-simulation systemic microgrid model during this work . . . 103
3.60 PyPSA bus model . . . 105
3.61 Representation of the diversity of agent within the microgrid and their modeling carateristics . . . 106
3.62 Representation of the generic agent in the financial model . . . 106
3.63 Induced Bayesian network of the example . . . 107
3.64 Bayesian network inspired from the TAM model . . . 108
3.65 Graphical representation of the energy model states. . . . 110
3.66 Graphical representation of the social model states. . . . 112
3.67 Graphical representation of the social model states. . . . 113

4.1 Scope of the Chapter 4 . . . 117
4.2 Model transformation layers within the microgrid modeling process . . . 118
4.3 Evolution of the modelling approach during the PhD . . . 119
4.4 Graphical representation of the dependency diagram generated by Excel for one step of simulation . . . 122

4.5 A PDEVS-based systems . . . 122
4.6 The basic PDEVS model sequence . . . 123
4.7 DEVS class structure from PyPDEVS . . . 125
4.8 DEVS class structure from PyPDEVS . . . 126
4.9 Representation of the dependency tree that is generated by the spreadsheet . . . 128
4.10 Result of the simulation of CREDA microgrid . . . 129
4.11 Result of the simulation of DESI microgrid . . . 130
4.12 Result of the simulation of GETP microgrid . . . 131
4.13 Result of the simulation of EDH microgrid . . . 132
4.14 Result of the simulation of HPS microgrid . . . 133
4.15 Result of the simulation of OREDA microgrid . . . 134
4.16 A PV plant and its cells of WBREDA microgrid . . . 135
4.17 Graphical representation of the component-based microgrid model . . . 137
4.18 Representation of the dependency tree that is generated by the spreadsheet . . . 138
4.19 Components evolution during the simulation . . . 139
4.20 Architecture of the microgrid co-simulation executable model . . . 141
4.21 Archtitecture of the microgrid co-simulation . . . 142
4.22 Scheduling a simulation step of the executable model . . . 142
4.23 Proposed co-simulation model . . . 143
4.24 Simulation of a simple microgrid between two user . . . 144
4.25 Simulation of a More complex microgrid with 7 users . . . 145
4.26 The two visions of microgrid in the literature . . . 147
4.27 The experimental frames related to the intersection of *energy* and *community* microgrids . . . 148
4.28 Microgrid systemic homomorphic review . . . 148
4.29 The structural graphical model . . . 149
4.30 The generative graphical model . . . 149
4.31 The replicative graphical model . . . 150
4.32 Evolution of the modelling approach during the PhD . . . 150
4.33 Systemic co-simulation tool adapted from [4] . . . 151

List of Tables

1.1 Levels of system knowledge [5] . . . 14
1.2 Fundamental system problems [5] . . . 15

2.1 System specification hierarchy [5] . . . 23
2.2 System morphisms [5] . . . 24
2.3 Model to experimental frame validity . . . 24
2.4 Classification of microgrid power architecture . . . 29
2.5 Information component architectures and their layers adapted from [6] . . . 32
2.6 The information architectures and their relationship with control issues . . . 33
2.7 Summary of the different control issues in microgrids . . . 34
2.8 The seven V's management in the different data layers . . . 34
2.9 Classification of microgrid business model architecture . . . 37
2.10 Cross-analysis between social elements, social architecture and their acceptance[7] 39
2.11 Different strategies for social acceptance based on the elements acceptance . . . 40
2.12 List of the issues of microgrid sustainability . . . 42
2.13 Recent literature review on microgrid relating to the topics addressed in this review. 43

3.1 Detailed system specification hierarchy [5] . . . 47
3.2 System specification formalisms . . . 49
3.3 Description of the exchanges. . . . 60
3.4 Evaluation of the different component stock level . . . 62
3.5 Evaluation of the different flows and exchanges. . . . 63
3.6 Details of components of the generative model . . . 72
3.7 Open source python based power flow simulator review adapted from [8] . . . 104
3.8 Discrete distribution of the rain probability . . . 108
3.9 Probability of the Bayesian network of the umbrella taken or not . . . 108
3.10 Discrete distribution of the availability of the energy . . . 109
3.11 Discrete distribution of the affordability of the energy . . . 109
3.12 Discrete distribution of the agent's experience level . . . 109
3.13 Probability of the Bayesian network of the agent's perceived ease of use . . . 109
3.14 Probability of the Bayesian network of the agent's perceived usefulness . . . 109
3.15 Probability of the Bayesian network of the agent's intention to use . . . 109
3.16 Bayesian network probability of the agent's actual use . . . 109

General Introduction

This research work seeks to answer a simple question: "Why do microgrids fail?" Behind these four words lie all the work proposed in this thesis, its ideas, models, bibliographic revisions and questions. What is "fail"? What are "microgrids"?

Through four chapters, this work will dive into these questions and take the reader through a microgrid literature analysis underpinned by Modeling and Simulation (M&S) theory. The main contribution of this thesis revolves around the idea of exploring a multi-disciplinary problem (microgrids) through a powerful and formal theory (M&S). The hope is that a formal modeling approach will help identifying which internal mechanics are at stake in microgrids, and contribute toward their success or failure. To the best of our knowledge, this combination of microgrids and MS has never been explored to this extent with this multi-disciplinary touch before.

To kick-off this work, Figure 1 shows the overall structure of this document. This structure is at the core of M&S theory, and we will follow its practical application to the letter.

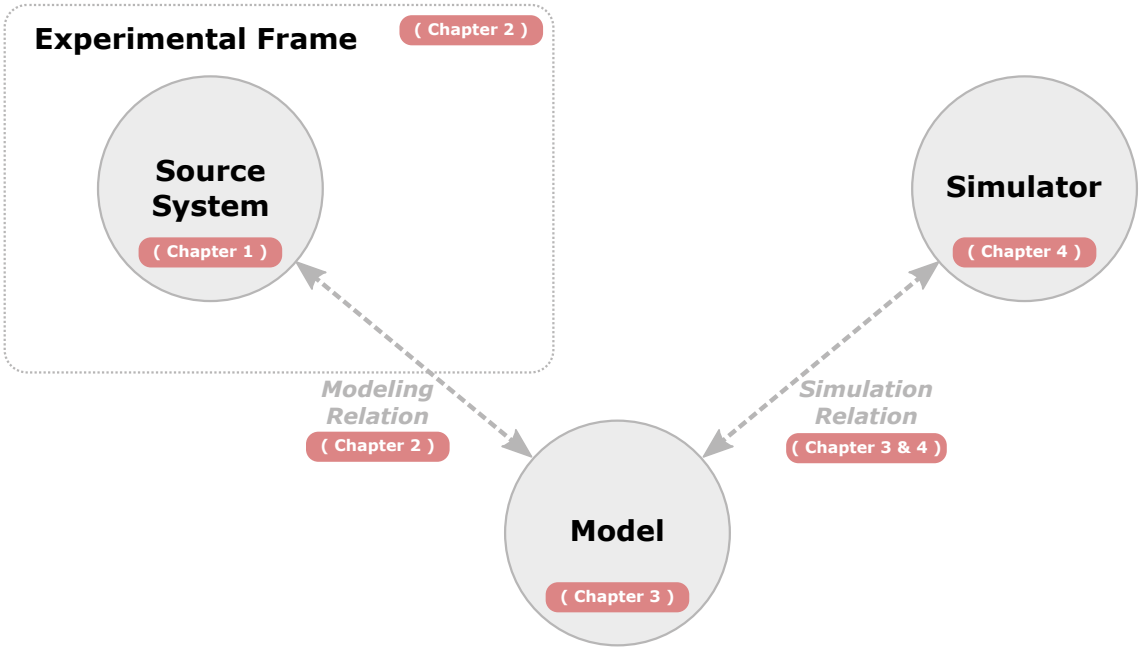


Figure 1: Core entities in modeling and simulation and their relationship related to the manuscript chapters

The words "Modeling" and "Simulation" are often loosely defined in current practice but have a very precise meaning in the framework used in this thesis [5]. This theory proposes four important steps in the process of creating models and simulating them, namely, choosing a source system, defining an experimental framework, building a model and creating a simulator. At each chapter, this work will also take the description of the M&S theory further to better provide the reader with the theoretical concepts necessary to understand the process.

The source system, composed of either real or artificial sources of data, must be defined. In this thesis these are microgrids. More specifically, microgrid sustainability. microgrids and their

shortcomings will be explored in Chapter 1. This chapter will also present the levels of system specification, which provides formal means of describing systems in M&S.

The experimental framework specifies the conditions under which the system is observed or experienced with. It defines the objectives of the modeling and simulation project, as models can only provide correct results when used within the experimental frame from which they are built. In our case, due to the complexity of microgrids and their sustainability, the main source of "observation" will be a thorough literature review. This review will take place in Chapter 2 and be built in the light of M&S theory, in order to provide the basis for a novel microgrid model.

Models are made of instructions, rules, equations or constraints designed for generating an input/output behavior. The word "model" is loosely defined in many different fields, which makes essential to provide a system theory backed by a sound mathematical foundation for its semantics. microgrid modeling requires a multi-disciplinary approach, interfacing different disciplines and, hence, different types of models. Chapter 3 will present all the details of modeling formalisms, and apply them to present three different microgrid models going from less to more detailed.

The simulator is the entity that interpreters the model to generate data from it. It can be seen as an agent that follows a rigorous and clear set of instructions, that generates data from the model. Chapter 4 will present how the simulator proposed in this work handles elements such as time, events, collisions and many more, making its construction an important step in this work. Three different attempts at building a simulator will be presented, with the main effort and focus being on the final one.

Much of this work makes a clear and direct reference to Ziegler et al's book "Theory of Modeling and Simulation" [5]. All of the M&S theory concepts embedded in each chapter is a summary made for the purpose of helping those new to the theory as the modeling effort becomes more complex. Any misrepresentations and/or error on this part is solely this authors' fault. If the reader wishes to dive deeper in the theory, or is not satisfied with the explanations, we can only advise to refer to the source.

Chapter 1

The Source System: The basis of modeling and microgrids issues

This chapter introduces the notion of source system and how it applies to microgrids. The main objective is to outline the problem that will underpin this thesis and the methodology adopted to address it. The scope of this chapter in M&S theory is illustrated by figure 1.1.

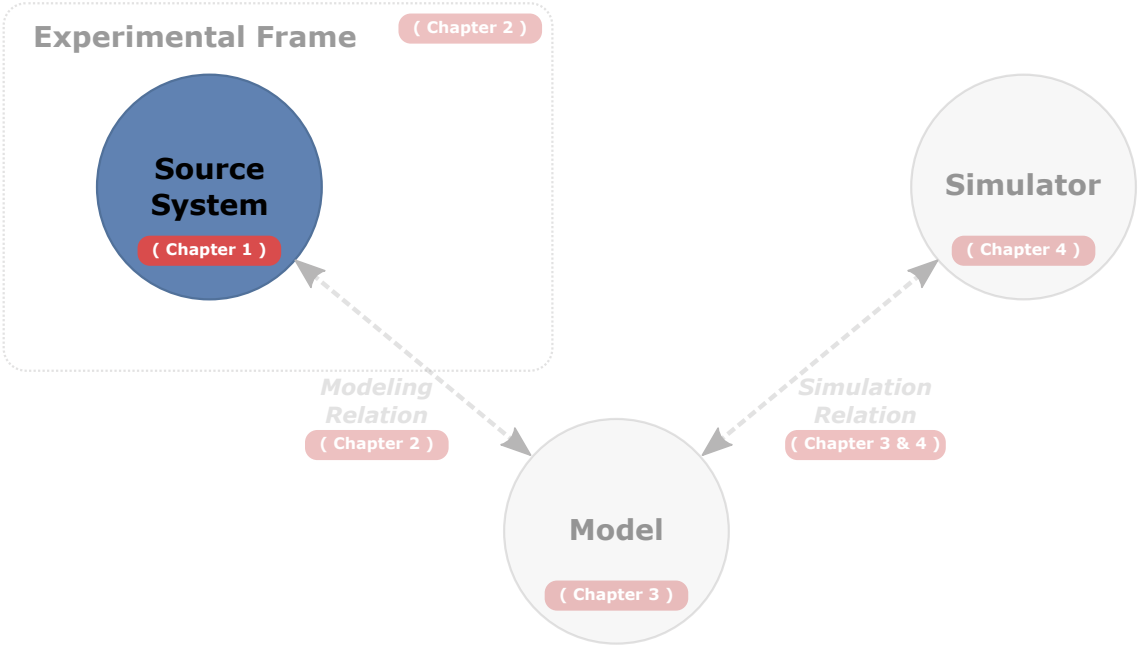


Figure 1.1: Scope of the Chapter 1

1.1 M&S theory: The basics

System modeling is at the core of scientific activity. As a consequence of such a wide use, the words "knowledge", "model", "data" often have slightly different meanings within different fields of scientific investigation. The M&S theory provides an formal framework of concepts which will be used in this work to guide the modeling process of microgrids. It is thus necessary to provide the reader with an overview of the primary system modeling concepts that guide this work.

This section provides the main concepts that will be most useful in this chapter and the concepts around source systems and modeling itself.

1.1.1 What is a source system?

The source system is the system under study, which will be the focus of the model engineering effort. It is the *source of observable data* in the form of time-indexed samples of variables [5]. The data extracted from a source system can be gathered into a *system behavioral database*.

Systems are not the same regarding the data available to populate their associated databases. *Data rich* source systems have a wealth of data at the disposal of the modeler, or this data can be generated with little effort. *Data poor* source systems, in contrast, provide either little data to work with or the data is of low quality.

The first primary challenge of modeling any system is to clearly define if they are data rich or poor. This will fundamentally define the experimental frame best suited to extract the data and decide how much data is "enough". As will be seen in this chapter, microgrids are a particular type of source system with some crucial *data blind spots*.

1.1.2 Basic system modeling concepts

What is modeling? What is a model? Are there different methods for creating models? Can these methods be regrouped or categorized? What should be known in advance before starting the modeling process?

This section will build the ideas of system modeling from the basic concepts of system knowledge up through system problems and finish with a hierarchy of system specifications.

Levels of system knowledge

George Klir [9] proposed a system knowledge framework that is a handy starting point. Summarized in Table 1.1, this framework goes from level 0, i.e., the source, up to level 3, i.e., the system's structure.

Table 1.1: Levels of system knowledge [5]

Level	Name	What we know at this level
0	Source	What variables to measure and how to observe them
1	Data	Data collected from the source system
2	Generative	Means to generate data in a data system
3	Structure	Components coupled together to form a generative system

The most basic level of knowledge is the Source level, which represents the knowledge of the system and its relevant observable characteristics. The Data level that follows is obtained by actually acquiring data from it, typically collected via deliberate observations of the source system to constitute a database. Modelers can then derive from this database an equation or some other

generative tool with the aim of producing similar data to reach the Generative level. When scientists talk about models, they usually refer to the Generative level of system knowledge. Typical systems require a chain of equations or components passing information one to the another. Each will contribute to generate the data observed on different parts constituting the whole source system. This is the Structure level, when the internal organization is represented within the model of the system. Scientists tend to talk about systems when operating at this level.

Typical system related problems

The levels of system knowledge naturally show that moving between levels means acquiring a new understanding of the system. This *quest for knowledge* encapsulates the three types of problems related to systems: *analysis*, *inference* and *design*. These different types of problems are summarized in the Table 1.2.

Table 1.2: Fundamental system problems [5]

Problem	**Does the source system exist?**	**What are we trying to learn about the system?**	**Which level transition is involved?**	**Example**
Analysis	Yes OR No	Behavioural data from components	From higher to lower levels	Creating data from equations
Inference	Yes	Components from behavioural data	From lower to higher levels	Fitting equations from data
Design	No	New system creation	From lower to higher levels	Coupling known components to generate expected data

This framework creates new knowledge when moving up the levels (in Klir's sense). Moving down makes explicit (creating the data) what was implicit in the higher levels. When studying a given source system, a modeler may need to iterate this *quest for knowledge* to the satisfaction of the objectives of the study.

From this basic epistemological framework, a more formal hierarchy of system specification can be drawn. This hierarchy is beneficial from practical modeling and simulation perspective.

In this work we focus on the analysis of microgrid systems, with the objective of understanding what make them fail. This means in practice that this research work will follow the path from higher to lower levels, where behavioural data is generated from components.

1.2 Microgrid: Problem statement and thesis objectives

microgrids sustainability is the *source system* of interest in this thesis. This section frames the main driving questions of this research work and addresses the issues regarding microgrid systems. The objective is to clearly define what kind of source system it is and provide the basis for its description. This section will begin by stating the motivations for this work. From these motivations, the main scientific questions of interest will be formulated. Finally, an overview of the source system will be given, providing the basis for the next chapter.

1.2.1 Thesis context: Energy access and microgrids

Energy access is fundamental to lifting people out of poverty [10], emancipating women [11], and creating local-driven development through meaningful, productive use cases [12]. According to the United Nations, 789 million people are living with no access to electricity [10], and without a more engaged action, 650 million people will still remain without energy access in 2030 [11]. Renewable energy-based technologies can provide safe and clean electricity at lower costs [13]. However, sustainable and affordable rural electrification remains a very complex and risky endeavour [14].

To provide electricity to a new location, there are essentially two solutions: extending the main grid or creating a local installation. Existing definitions of what this local installation may be can be broadly regrouped into two categories that will be used throughout this work, namely *energy microgrid* and *community microgrid*.

The *energy microgrid* can be defined as distribution networks comprising various *distributed generators*, *storage devices* and *controllable loads* that can operate either *interconnected* or *isolated* from the main distribution grid as a *controlled* entity [15]. Figure 1.2 shows these *elements* centered around a *control* with the possibility of *connection*. *Energy* is what flows between the elements and *information* is used by the control to regulate this flow.

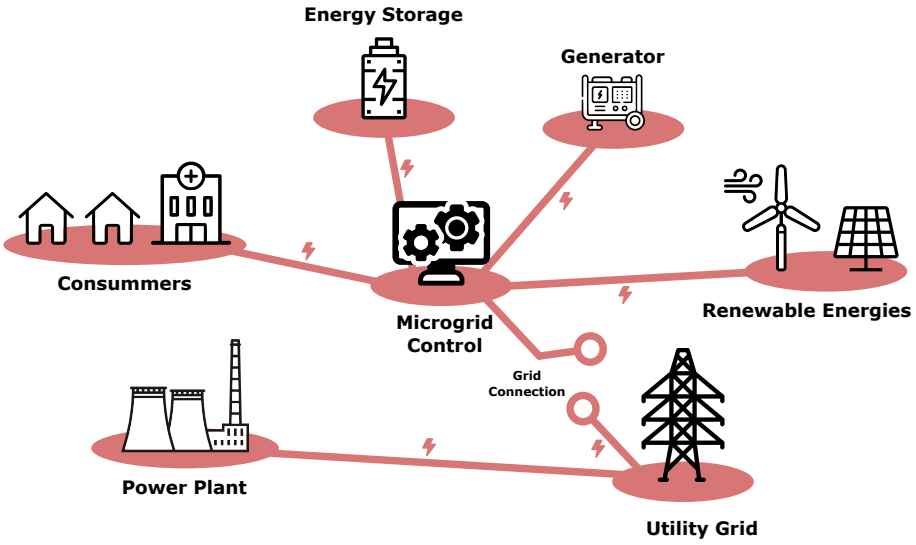


Figure 1.2: The energy microgrid

Energy microgrids vary in complexity, going from simple individual *stand-alone systems* to bigger *microgrids*. Figure 1.3 shows different energy usage levels and their grid equivalent size.

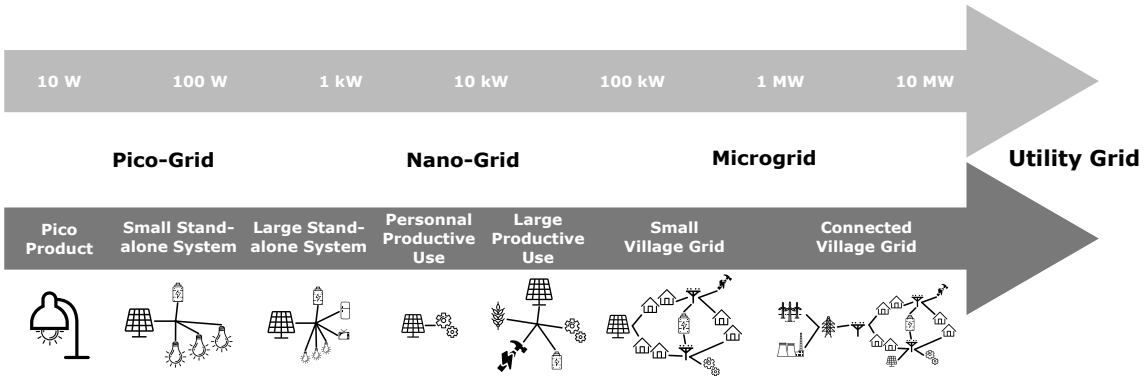


Figure 1.3: Evolution of usage and architecture with the grid power adapted from [1]

Community microgrids can be defined as an electricity supply and distribution with an *ownership structure* and objectives of achieving *economic*, *social*, and *environmental benefits* [16]. Figure 1.4 illustrates the different *stakeholders* which compose the ownership structure, and are centered around a *socio-economic network* representing the community. *Money* is what flows between these stakeholders, and *social acceptance* is what measures the level of appreciation of the community regarding the objectives of the microgrid.

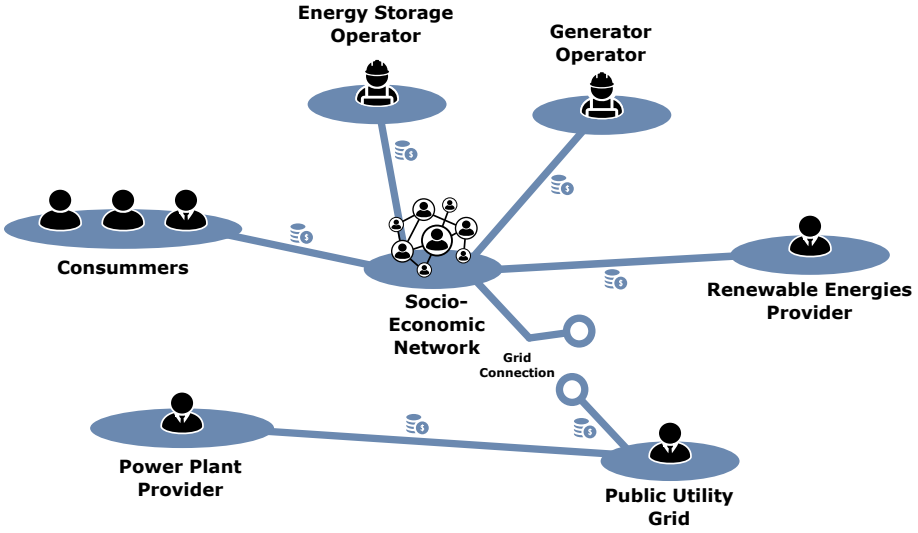


Figure 1.4: The community microgrid

These solutions and their cost evolution are illustrated in figure 1.5. This representation indicates the advantage of implementing microgrids in particular areas as an intermediate solution between the extension of the national grid and the solar home system.

The cost of main grid extension rises steeply with distance being driven by the terrain [17]. microgrid costs also rise, but more slowly, essentially due to logistics of installation and maintenance [11]. Stand-alone systems have more fixed costs, essentially driven by their production volume [2].

microgrids and stand-alone systems have an essential role in rural electrification due to their high versatility and lower cost [18]. However, despite all the qualities of microgrids, the uptake of this type of system in rural communities remains extremely low. Technology used on pico-grids are usually not designed to operate in microgrid or utility grid level, which represents a challenge from an investment perspective as to the issue of locking users in certain power level traps due to financial constraints of investing at an infrastructure whose power level is not adapted to the evolution of the local needs [1].

microgrids are thus seen by the literature from two different points of view. These points of view also condition the definition of sustainability, which is the core object of study in this work.

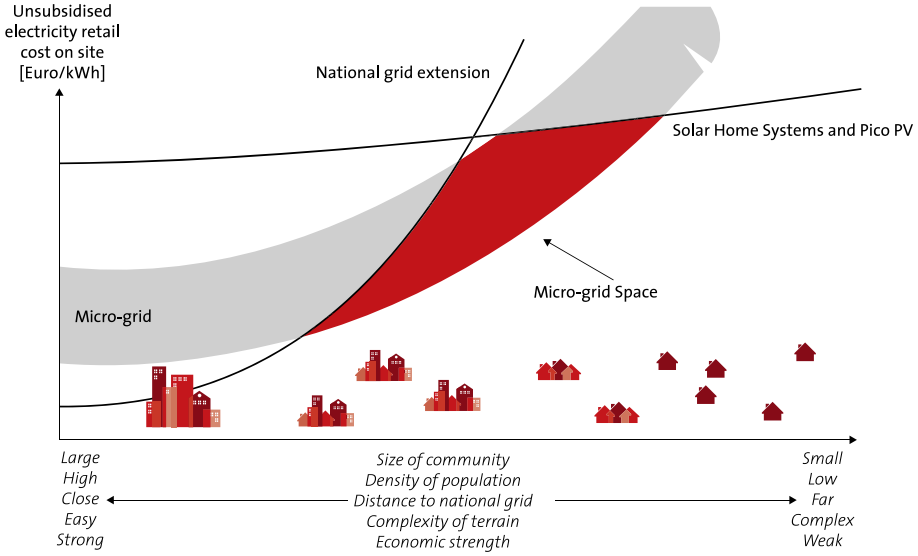


Figure 1.5: microgrid-space: the conditions where microgrids are competitives in comparison with national grid extension and solar home systems [2]

1.2.2 Thesis object of study: microgrid and its sustainability

microgrids and their sustainability have been studied extensively in the literature, with a growing interest in the past ten years, as shown in figure 1.6.

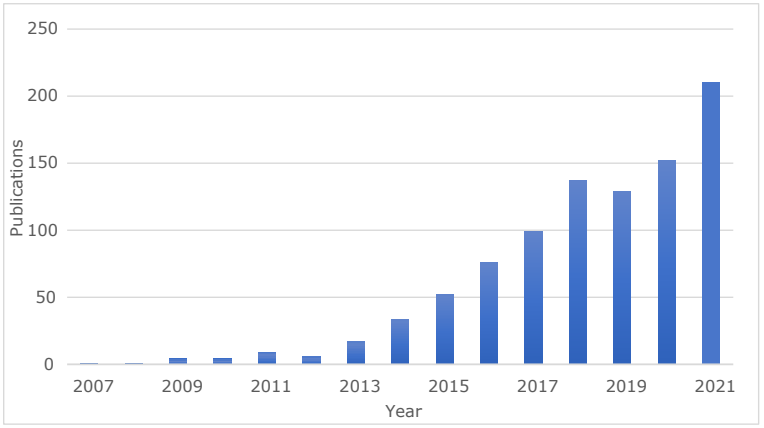


Figure 1.6: Evolution of Sustainability and Resilient microgrid publications in the literature

This figure shows the annual publications related to sustainability or resilience for microgrids since 2007. The data is extracted from the search engine webofscience for the publishers IEEE, Elsevier, and MDPI. As a result, 837 publications have been identified as relevant because they include the notions of sustainability and resilience in their study. The interest in these notions of microgrid has dramatically increased, showing the interest in finding solutions for the weaknesses of microgrid systems. The concept of sustainable energy development has evolved to become really central in energy and microgrid creation and operation and is now a key factor for its study [19].

This literature tends to focus on identifying limiting actions or promising practices to be followed to optimize the microgrid [20, 21, 22]. However, it tends to overlook non-technical actions, which are also crucial to guarantee long-term sustainability to the microgrid [23]. The notion of virtuous circle [24] illustrates the impact of non-technical elements such as fair tariffs, proper maintenance,

and availability of well-trained human resources [25] on microgrid sustainability. For rural areas, microgrid sustainability is paramount because these populations cannot afford a system failure [26].

The obstacles to the implementation of microgrids are the lack of economic and business consideration [27], the gap in strategic planning approach [28], the neglect of social issues [29] and the deficiency of an apparent global consideration on overall microgrid sustainability [14]. Furthermore, as the population grows, its needs will change, and so will the way it consumes energy, often creating the need for more energy [30].

Put together, the dual vision of energy and community microgrids, with their differences in the definition of sustainability set the conditions to the emergence of the main problem this work will focus on.

1.2.3 Problem statement

microgrid sustainability is a multi-disciplinary problem split into two visions, namely *energy* and *community* microgrids. As a consequence, microgrid studies are done by specialists from different fields that use different modeling techniques to describe their vision of a microgrid. These methods yield models that cannot operate together, meaning they are incapable of studying microgrid sustainability in a comprehensive way. This incompatibility of different models in the literature is the core problem treated in this thesis.

1.2.4 Thesis motivation: Contributing to sustainable electricity access in rural areas

This thesis one main motivation is to contribute to the understanding of microgrid *sustainability*. This contribution is based on exploring different methods of creating models that integrate *energy* and *community* microgrid aspects. This model compatibility is mostly absent from the literature and it is expected to help shape future sustainability studies in microgrids from the scientific community.

1.2.5 Thesis methodology: Systemic study, model and simulator

The M&S theory will guide our exploration process of different types of models and simulators which will become more sophisticated as we explore different *degrees of knowledge*.

From a modeling perspective the main problem of microgrid sustainability is that it is a *source system* with varying levels of knowledge. While *energy microgrids* literature is *data rich*, it remains comparatively *data poor* in *community microgrids* topics. Furthermore, technical and non-technical models are not structurally connected (or "connectable"), hampering the study or their joint (or "cross") contribution to the sustainability of the microgrid as a whole.

In M&S theory terms, the core problem is how to structurally connect *energy* and *community* microgrid based models built from analyzing heterogeneous levels of knowledge of the same *source system*. The thesis will be structured in three main chapters.

Chapter 2 will focus on exploring the literature through a systemic study with the objective of creating an unique framework for describing microgrids which is an important contribution of this thesis.

Chapter 3 will explore different models based on the conclusions that there is a clear need to link sustainability in technical and social terms. Since this work deploys a formal approach to modeling, **it was decided to gather all the models in a single chapter, while the simulation results will be introduced in the next chapter**.

Chapter 4 will explore the different simulators that were developed during the thesis to test the modeling approaches. While several simulators for microgrids exist in the literature, they do not consider multi-disciplinary models. Each simulator will be described and its inner mechanics explored in detail. Simulation results will be discussed for each implemented simulator.

The core contribution of this work is to formally explore models which can encapsulate the complexity of microgrids as a multi-disciplinary technology and its associated sustainability issues.

Chapter 2

Experimental frame: Describing microgrid systems and their sustainability

In the previous chapter, we characterized the two main visions related to microgrids in the literature: *energy microgrid* and *community microgrid*. Energy microgrids are described from a technical perspective, integrating aspects related to energy flow and information flow. Community microgrids are described from a social perspective, integrating aspects related to financial issues and social acceptance. This duality was shown to be at the origin of the main problem this work focuses on: the incompatibility of the models yielded from these two visions.

This chapter focuses on bridging the gap between these models by creating a unified *experimental frame* around energy and community microgrids. From the M&S theory, *experimental frames* describe the methods through which the modeler will acquire data on the system to be modeled. If a joint model is to be created between energy and community microgrids, a joint experimental frame is its natural starting point. The scope of this chapter is given in figure 2.1.

This chapter will describe sustainability from the perspective of this joint experimental frame and lay the foundations to the creation of the models and simulators of the next chapters.

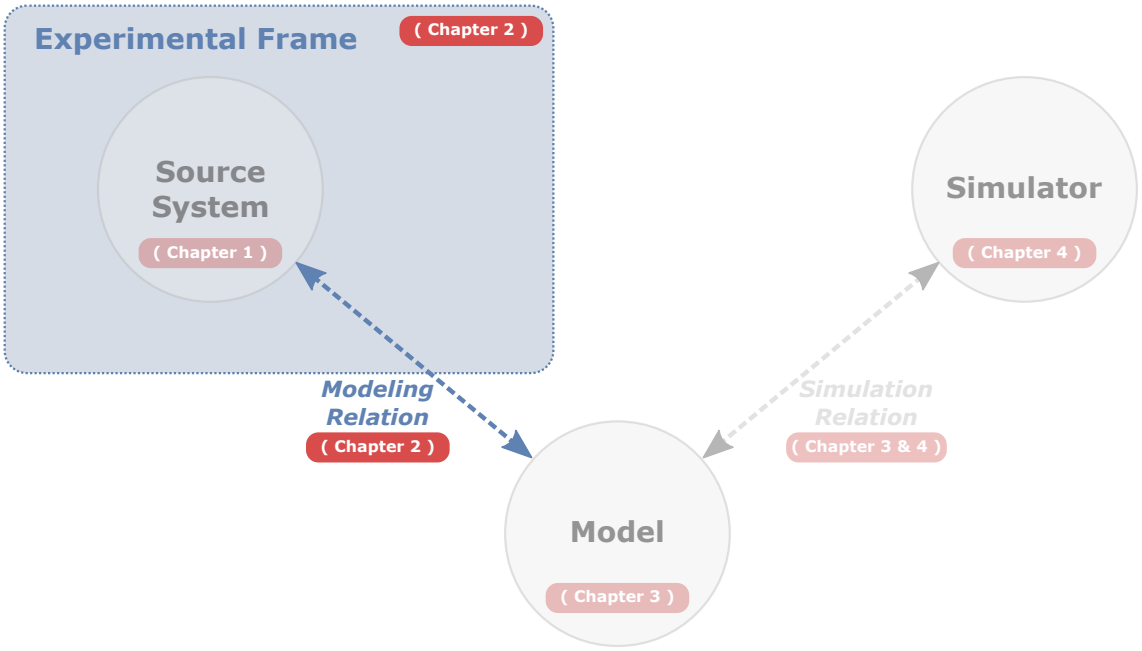


Figure 2.1: Scope of the Chapter 2

2.1 M&S Theory: Framing the system

2.1.1 What is an experimental frame?

In the previous chapter, we explained how a *source system* provides data from which a *generative structure* can be built to reproduce this data. Thus, the choice of *which data to extract from the source system* is central to its modeling. This choice is called an *experimental frame*.

There are two equally valid but different views as to what is an experimental frame. One is to define what kind of data elements should go into the database. The other is that a frame is a system that interacts with the source system to obtain data under specific conditions.

Data rich source systems for which a wealth of quality data is already available would put its modeler in the situation of cherry-picking through the data and possibly filling the gaps, if any exist. *Data poor* source systems for which a small amount of dubious data is available (or no data at all) would put its modeler in the situation of considering the necessary data elements to minimize the efforts of building the system to measure it. Notice that even if the system is non-measurable (typically in social-economic models), both views can be applied to what is known as the literature review or documentary research.

An *experimental frame* sets the objectives that motivate the modeling and simulation project [5]. These objectives will focus the development of the model on particular issues and establish which data is best to be extracted from the source system to represent these issues. Thus, the experimental frames can be developed only once the objectives are known.

In this thesis the objective is to create a joint experimental frame between *energy* and *community* microgrid.

2.1.2 Systems in the vision of MS theory

Simulation of a system deals with *dynamics* [5]. Thus, simulating systems means understanding their input/output behavior evolution over time. This practical system description is given in figure 2.2.

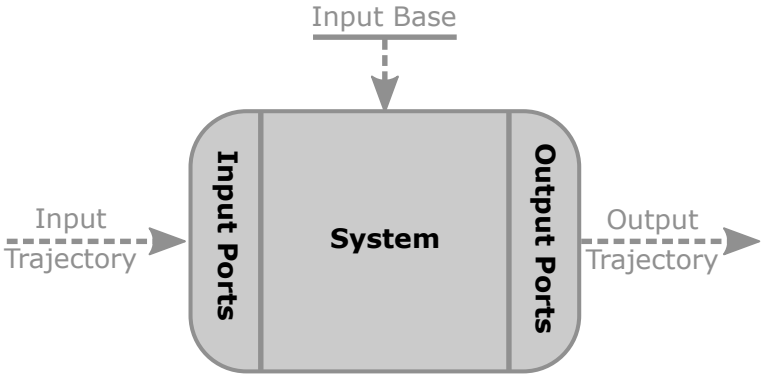


Figure 2.2: A basic I/O system

A system has *Ports*, *Trajectories* and a *Time Base*. *Ports* represent the way of interacting with the system. It can be either by stimulating it (inputs) or by observing it (outputs). *Trajectories* are built from time indexing these stimuli (input trajectories) and observations (output trajectories). The *time base* provides the time indexes needed by the trajectories.

Hierarchy of system specifications

System specification levels are built on these basic concepts while adding a few more to explain what is inside the system itself. These levels are summarized in table 2.1.

Table 2.1: System specification hierarchy [5]

Level	Specification name	Equivalent knowledge level	What is known at this level?
0	Observation Frame	Source	How to stimulate the system. What variables to measure. How to observe them over a time base.
1	I/O Behavior	Data	Time indexed data. Input/Output pairs of points.
2	I/O Function	Data	Knowledge of initial **state**. Each input stimulus produces an unique output.
3	State Transition	Generative	How states are affected by inputs. What is the state after input is over. What output **event** is generated by a state.
4	Coupled Component	Structure	Components and how they are coupled together. Component to component coupling. Component aggregation and structure hierarchy.

A total of five system specification levels are identified in the M&S theory proposed by Ziegler. To summarize the difference, two key concepts are shown in bold in table 2.1, **state** and **event**.

The notion of *state* relates to the uniqueness between input and output. Given the initial state of the system, an input trajectory can be mapped to a single and unique output trajectory. In practice, it is through the internal state that functions can be mapped to data. Note that when the model is at Structure level, the system state is itself composed of multiple states, one for each component in the model.

As the system receives inputs, its states change. The state can also evolve spontaneously as a result of time evolution. State transitions register these changes. As the system state changes, it triggers *events* on its output over time.

When dealing with a system, a modeler will be confronted with a series of issues from data to component coupling. Handling these issues means using one of the possible modeling formalisms described below.

In this thesis, these issues will be handled progressively, starting with the problems of microgrid systems in general and leading to the source system of this work: *microgrid sustainability*.

Morphism and association

Two different systems can be related to one another via *morphisms*. These relations can be set at each level of their specification. Table 2.2 details how the morphisms relate to each level.

Table 2.2: System morphisms [5]

Level	Specification name	Equivalent knowledge level	Systems are morphic if...
0	Observation Frame	Source	Their input, output and time bases can be put into correspondence
1	I/O Behavior	Data	Time-indexed I/O pairs match in one-to-one fashion.
2	I/O Function	Data	Their initial states can be put into correspondence. Their I/O functions with corresponding states are the same.
3	State Transition	Generative	A sequence of state transitions in one model has a correspondence in the other, creating a homomorphism
4	Coupled Component	Structure	Components can be put into correspondence and their couplings are equal.

It is important to note that source systems may be described simultaneously at different levels. *Association mapping* is the act of defining the links between higher and lower levels. It is not unusual to map several higher-level components to a handful of lower-level ones. This top-down mapping is much easier than its bottom-up counterpart, as it is an exercise of deduction rather than inference.

In this work, this top-down approach is the method that was chosen to explore the creation of models that integrate energy and community microgrid elements.

2.1.3 Relations between models and experimental frames

The last theoretical concept that will be explored in this chapter it the relations between models, simulations and their experimental frames.

The *validity* of a model is the degree to which a model faithfully represents the source system. This is achievable only to the extent demanded by the objectives of the modeling study. Ideally, it should be impossible to distinguish a model from its associated source system within the scope of the experimental frame. *Validity* can be *replicative*, *predictive* and/or *structural*. The description of these concepts and their correspondence in terms of system specification level is shown in table 2.3.

Table 2.3: Model to experimental frame validity

Type of validity	System Specification Level	Description
Replicative	I/O Behavior	For all experiments possible within the experimental frame, the behavior of the model and the system agree within an accepted tolerance
Predictive	I/O Function	The ability to predict an yet unseen system behavior.
Structural	State transition and coupled component	The model mimics in step-by-step, component-by-component fashion, the way the system does its transitions

The model building process can thus be seen as a way to validate different levels of system specification. Successful modeling can be seen as valid simplification [5].

Detailed models can be simplified into *lumped models*. Demanding experimental frames yield more complex and complete data, leading to detailed models. Lumped models, by contrast, require simpler experimental frames.

The M&S theory states clearly that the modeler has the freedom to experimentally frame the source system according to a clear set of objectives. This frame can be made as simple or as demanding as possible, with the direct consequence of yielding lumped or detailed models. In the process of modeling a system, a modeler can evolve models and experimental frames. This is particularly useful when modeling a complex system such as a microgrid.

2.1.4 M&S theory in this chapter

The objective of this chapter is to create a joint experimental frame for energy and community microgrids. This experimental frame must implement a homomorphism capable of rendering the model from these two visions compatible with one another. This experimental frame must also allow the exploration of modeling through a top-down approach, where the models will go from structural to replicative and eventually predictive.

2.2 Microgrids: Systemic analysis

The previous chapter has introduced the issue of sustainability in microgrids as the *source system* of this thesis. This system comprises heterogeneous levels of knowledge between two main structural blocks, namely the technical and non-technical structures. This structural mismatch was shown to be the core problem that this thesis will seek to study.

The M&S theory shows that models are built within the scope of *experimental frames*. These experimental frames set the objective of the modeling effort and provide the base data which allows going up the system specification ladder. This chapter will build the experimental frame and set the objectives which will guide the modeling of sustainability in microgrid systems, our source system.

2.2.1 Towards a more comprehensive understanding of microgrid systems

The previous chapter has highlighted the difference between the *energy* and *community* visions of microgrid systems. An essential objective of any experimental frame associated with the sustainability of microgrids is to avoid this divide as much as possible. Thus, the description of the *energy* and *community* visions should be *as structurally similar as possible*.

This chapter seeks to create an experimental frame capable of providing data from both types of microgrids in estimating the sustainability of a microgrid [31]. Figure 2.3 depicts the existing experimental frames related to these microgrids in blue and red. The gray areas show experimental frames that can capture some data from their intersection. The purple zone represents the experimental frame which is the objective of this chapter.

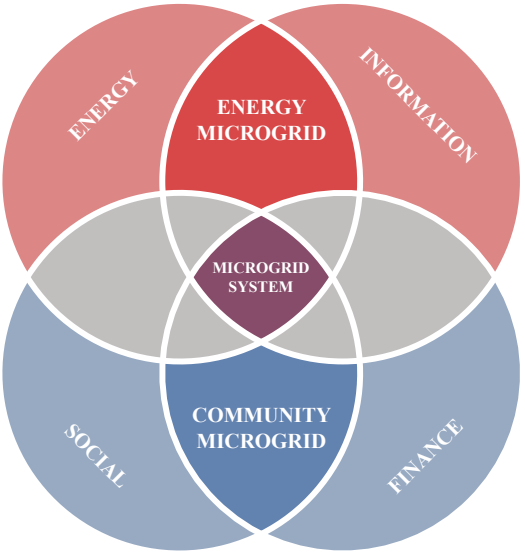


Figure 2.3: The experimental frames related to the intersection of *energy* and *community* microgrids

Finding data sources in the intersections shown in figure 2.3 is not trivial.

The data-rich *energy* microgrid category mostly provides data about behavior and function through analytic studies, leading to a bottom-up system inference specification that is fundamentally replicative in nature [32]. The objective of these models is to size the technical system from a series of hypothesis, typically on the load curve of the target community. The quality of these hypothesis is beyond of the scope of these models and represent their main weakness.

The data-poor *community* microgrids mostly provide deductions of the structure, components, and their associated states through systemic studies, leading to a top-down system analysis specification which is fundamentally structural in nature [33]. The objective of these models is to provide decision makers with the means to estimate the risk represented by the investment on the

microgrid or study its impact in the perception of the local community. The hypothesis regarding the robustness of the energy system, the availability of the energy and the funding necessary for maintaining the system are the weak points of these models.

The joint experimental frame in this chapter will seek to use homomorphism as a mean to guarantee a description of the energy and community microgrids which can yield interoperable model components.

2.2.2 Using the state of the art as joint experimental frame

From the M&S theory, one of the visions of the *experimental frame* is the act of choosing which information to gather in order to build the *data base* of the *source system*. This chapter sets out to do exactly that by conducting an in-depth bibliographical review on the sustainability of microgrid systems.

Figure 2.4 shows the distribution from the 900 publications according to the field(s) they study, highlighting how they are positioned in terms of the intersections shown in Figure 2.3. Most of these articles focus on analytical work and technical descriptions but often lack the systemic vision. Purely financial and social articles are less common. Most financial and social analyses are done in parallel with energy and information studies.

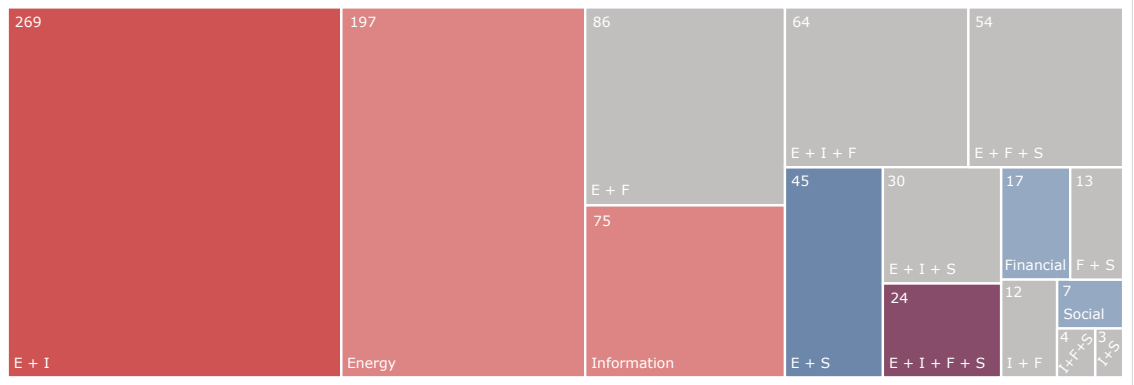


Figure 2.4: Sustainability and Resilience microgrid publications related to the field they study

microgrid reviews usually consider multiple parameters for a multi-disciplinary study of microgrids. These microgrid reviews make a clear and complete state of the art of the microgrid operation [34], describe sustainable business model solutions for the development of a microgrid [35], show all the difficulties which can appear in microgrid development [33], provide global performance indicators for the proper functioning of a microgrid [36, 37] make a systemic diagnosis of the microgrid state of health [38] or make a complete criticism of the limits of the microgrids [39].

The *experimental frame* proposed in this chapter pays strong attention to the *morphism* between the *energy* and *community* microgrids. This morphism tries to address the core issue of this thesis: the lack of inter-operability between the fields of the *energy* and *community* microgrids. Figure 2.11 shows the proposed frame.

The first important *morphism* in our experimental frame is the description of both types of microgrids with two *fields*. *Energy* microgrids have the *energy* and *information* field. *Community* microgrid have the *financial* and *social* fields. The knowledge from these fields can be described into three categories: elements, architectures, and issues. Each field has an associated *unit*, meaning energy is expressed in Watts, information in Bits, financial in Currency, and social in Acceptance.

The next sections will detail the elements, architectures and issues of each field.

2.2.3 Energy Field

The energy field represents all electrical and energy aspects of a microgrid and their interconnections. In other words, it represents all the energy fluxes of the microgrid. The assembly of energy

elements is done through energy architectures which are designed to answer energy issues. In electrical terms, microgrids are mainly seen as the assembly of electrical blocks that can be operated in island mode, grid-connected mode, or both [40].

Energy Elements

In this work, the energy component is composed of four types of elements: sources, storage systems, loads, and power electronics converters [41].

There is a wide variety of sources that can be connected to microgrids [42], which are mainly separated into two main types: conventional generators (diesel generator, gas turbine) and non-conventional generators (fuel cells, photovoltaic, hydro turbine biomass, wind, geothermal and solar thermal) [43]. Most microgrids work with renewable energy-based sources such as photovoltaics and wind turbines, usually associated with diesel generators. Diesel generators are particularly popular in microgrid applications due to their flexibility, low-cost, and ease of implementation, despite being known for their poor stability and low inertia [44].

Storage systems provide stability allowing to handle energy production intermittency and downtime, both typical of distributed generation [45]. Storage systems can be divided in: electrical storage (magnetic storage, supercapacitor), mechanical storage (flywheel, potential storage), electrochemical batteries, and fuel cells [46].

All the energy produced and stored is to be consumed by the users through loads commonly categorized into two types: fixed and variable loads [47]. A fixed load requires a constant flow of energy, while a variable load turns on and off according to a random control signal. Taken together, these types of loads make up a load profile used to design and manage modern *energy* microgrids [48].

Power electronics converters are used as an interface between sources, storage, and loads [49]. Power converters have two main functions, a power flow management function that controls the current and/or the voltage ratings between two elements and a passive filter function that eliminates harmonic contents created by the active fields whose switching is responsible for the power flow management [50].

Energy Architectures

Energy architectures assemble energy elements together in ways that suit the specific need of the microgrid. There are three main architectures available in the literature, represented in Figure 2.5 and summarized in Table 2.4: the AC architecture, the DC architecture, and the hybrid architecture [51, 52].

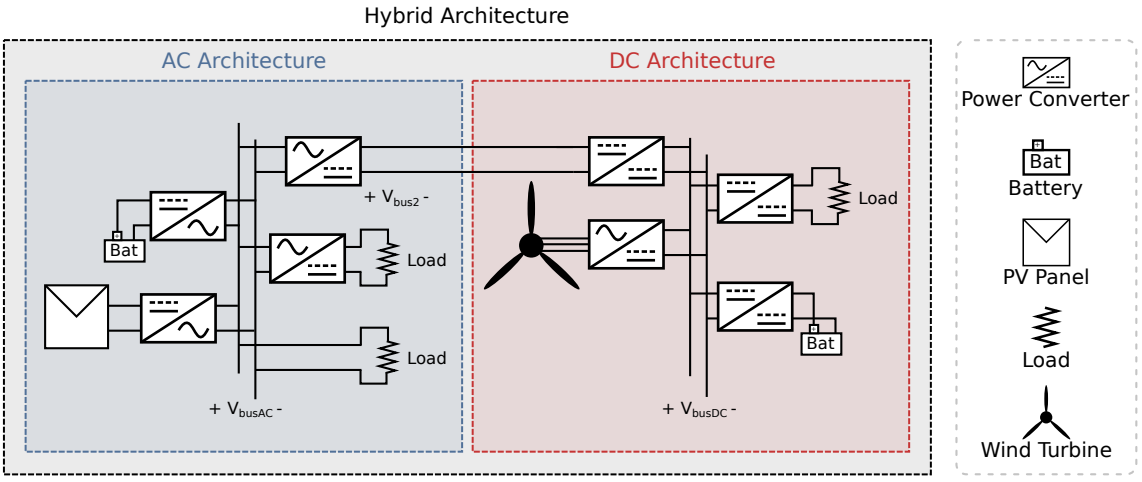


Figure 2.5: Representation of the different energy architectures

The AC architecture has been widely adopted for energy distribution by utilities due to its capability for transformation between varying voltages, thus leading to a reduction in energy loss during long-distance transportation [53].

The DC architecture is the most used architecture in microgrids because of its high concentration of DC sources and loads, resulting in easier management of the microgrid energy [54].

The hybrid architecture can be seen in many different microgrids and has more variety than classic AC or DC architectures. The classical way to build this architecture is to connect DC sources and loads to DC buses and AC sources and loads to AC buses [55]. This architecture can be optimized if all the sources and loads are connected to an AC bus and all the storage to a DC bus to centralize their management [56]. The DC zonal architecture connects all the loads to DC buses and uses AC buses to connect all the DC buses together [57]. The Solid State Transformer based microgrid centralizes all the buses in high-frequency transformers that can manage AC and DC feeders as well as power flow between the main grid and the microgrid [58, 59].

Between these three main types of architectures, there are many different solutions that are more subtle to solve a specific need with a mix of their benefits and downsides [60].

In the literature, classic microgrids are mainly seen as the assembly of electrical blocks that can be operated in island mode, grid-connected mode, or both [40]. A clustered microgrid can also be seen as an assembly of smaller grids that are interconnected to form a much bigger entity [61]. Swarm microgrids are a recent concept where the microgrid evolves in an "organic" way resulting in a swarm architecture capable of stability due to energy production decentralization [62].

In the rural electrification context, DC architectures are the most used since energy sources are easy to implement (solar PV and batteries) and basic loads such as light or electronic devices recharge can easily accommodate DC [63].

Table 2.4: Classification of microgrid power architecture

Architecture	Type	Advantages and Drawbacks	Articles
AC	AC microgrid	AC microgrid is really easy to implement and reconfigure, but requires a complex power electronic interface and a generally poor quality energy	[64, 53]
DC	DC microgrid	DC microgrids are relatively simple to control with a relatively good quality of energy but tend to be limited in terms of expansion and lack reliability with a distribution grid connection	[54, 65]
Hybrid	AC-DC microgrid	Combines the advantages of AC and DC architecture but cannot be suited for all applications	[55]
	AC microgrid with DC storage	More reliable storage devices and has similar performance to the hybrid AC-DC but the energy storage must be centralized	[56]
	DC-zonal microgrid	Allows different busses voltage and management technique but increase the complexity of the control	[57]
	Solid state transformer based microgrid	Very high quality of energy and high compatibility with AC or DC devices but the entire grid is dependent on the solid state transformer	[58, 59]
	Swarm architecture	Easy development and high overall reliability and flexibility but require a complex power electronic interface	[62]

Energy Issues

Based on the literature, this work considers the energy field to have three main issues: *appropriate system sizing*, *quality of the components*, and *protection* [66, 67, 68, 69].

Appropriate system sizing refers to the conformity of the microgrid to the needs and earnings of its final users [70, 71, 72]. The microgrid sizing must strike a balance between being large enough to avoid any outages and small enough to remain as affordable as possible for final users [73, 74]. This sizing problem can be broken down into a multitude of critical factors ranging from the optimization of the sizing of storage systems [75], to the understanding of local energy needs [76, 77], or the balance of different energy sources [78] but also by addressing more specific issues such as the optimization of the integration of hydrogen fuel cells [79].

Quality of components describes the microgrid equipment in terms of robustness and maintainability [80]. Current literature points out that higher quality of the electrical grid components as a whole avoids blackouts and dysfunctions [81], which is a crucial element, especially in rural areas [82]. For this, significant knowledge of the technologies available for implementing microgrids is required [42, 83, 84]. However, it is essential to keep in mind that these systems change and evolve rapidly and the grid quality also requires regular upgrades and maintenance [68, 85] which have been summarized in the literature by robustness, resourcefulness against disasters, rapid recovery and adaptability [86].

Protection of the microgrid refers to the equipment and techniques used to protect the elements of the energy system from any malfunction, internal or external to the element [87]. Extensive reviews have been made to present all the protection possibilities in AC or DC grids and different standards [65, 64, 88].

In the energy field, *sustainability translates as defining clear thresholds to the issues of sizing, quality, and protection.* Sizing should provide a compromise between power production capacity and foreseeable load. Quality is the cost-benefit analysis of the equipment cost used to build the microgrid. Protection is the number of protection elements for different kinds of faults and the overall cost linked to their presence or absence from the microgrid. These thresholds will then be used to monitor the evolution of the microgrid over time to detect any issues in the energy field.

2.2.4 Information Field

The information field is built on two different aspects of a microgrid: data and control. Control is used to regulate the power flow within the microgrid, handling voltage stability, power quality, and other issues related to keeping the electricity flowing. Data is related to the generation, transport, and aggregation of data from sensors and other sources spread throughout the microgrid [89, 90, 91].

Modern microgrids deploy a complex link between control and data [92]. The scientific community crystallize this link through the concept of smart grid [93]. In this work, the complex link between data and control is the cornerstone for analyzing the information field.

Information Elements

The information layer is made of two main elements: the information nodes and communication links.

An information node is composed of a control part and a data part. These parts vary significantly depending on the microgrid, its information architecture, and how the information issues are handled.

Communication links connect the different information nodes of the architecture together [94]. The literature on communication links divides them into different types of networks, namely, Wide-area network (WAN), Field Area Network (FAN), Neighborhood Area Network (NAN), Building Area Network (BAN), Industrial Area Network (IAN) and Home Area Network (HAN) [22, 95].

Information Architecture

The information field architecture is composed of information nodes that are connected together through communication links forming layers. These layers define the relationship between control and data on given information architecture. In this work, all microgrids are considered to have

three layers [6, 96, 97]. Figure 2.6 shows the relationship between information layers, control and data.

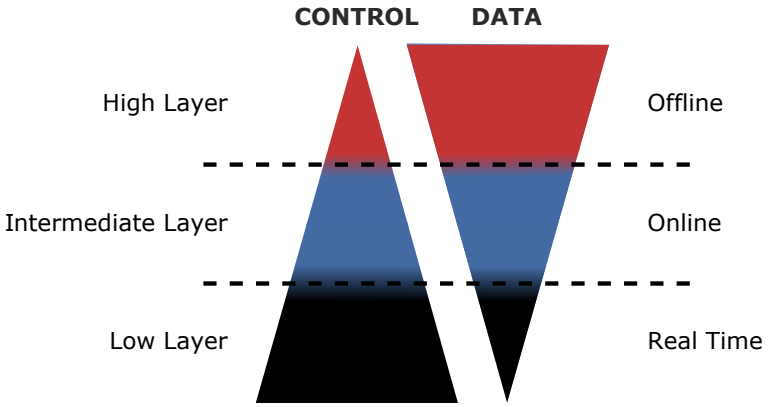


Figure 2.6: Relationship between data and control for different layers

Lower layers are more control intensive, usually operating in real-time or with stringent timing constraints. Their data intensity is low, requiring small number of measurements to operate. Their communication links tend to be very fast, embedded into devices and very close to power flows.

Intermediary layers are less control intensive, usually setting the reference points or tracking reference to the lower layers. They are more data-intense, requiring averaged and more robust data to calculate these operating points. Their communication links are slower, with more communication overhead, and use means such as wires or the air to exchange information.

Higher layers have very little control intensity, usually calculating parameters in asynchronous or very slow frequency. Their data intensity is very high, aggregating averaged values and many different data sources together. Their communication links are usually the slowest of all and with more significant overheads. These communication links tend to use already existing networks [6].

The architecture of the information field connects nodes together through communication links organized in layers depending on the type of power control, data management, and communication links that are needed by the microgrid. A literature review shows that control, data, and communication have the same types of architecture, namely centralized, hierarchical, distributed, and decentralized [70, 98, 96].

Depending on the architecture, the power control, data management, and communication approaches change, and so does the layer deployment, as shown in Table 2.5. Figure 2.7 shows how the nodes connect together to form the different architectures.

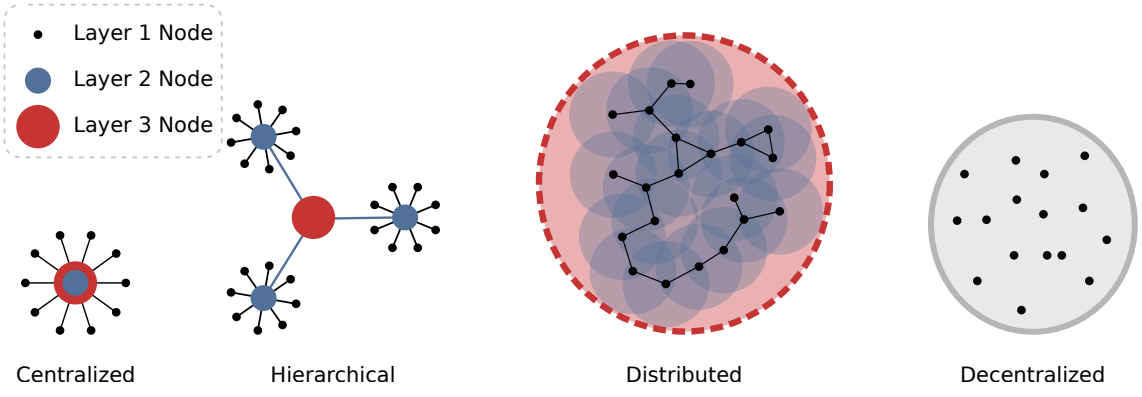


Figure 2.7: Representation of the different information architectures

Information Issues

To be sustainable, the information field must simultaneously address the issues from control, data, and communication. This section explains these issues in detail and analyzes them within the framework of the above elements and architectures.

Table 2.5: Information component architectures and their layers adapted from [6]

Layer	Node part	**Centralized** [15]	**Hierarchical** [101, 97]	**Distributed** [102, 103]	**Decentralized** [104]
		All nodes connected to a central node	Nodes are connected in a pyramid	Nodes are connected peer-to-peer	Nodes are not connected to one another
Low Layer	**Control**	Source/Load controller	Source/Load controller	Source/Load controller	Source/Load controller
	Data	Source/Load measurement	Source/Load measurement	Source/Load measurement	Source/Load measurement
	Communication	HAN/IAN	HAN/IAN	HAN/IAN	HAN/IAN
Intermediate Layer	**Control**	Source/Load balancing	Local Source/Load balancing	Source/Load droop control	Source/Load droop control
	Data	All lower layer measurements	Area-wide measurements	Only connected lower layer nodes measurements	No data from other nodes
	Communication	FAN/NAN/BAN/WAN	FAN/NAN/BAN	FAN/NAN/BAN	HAN/IAN
High Layer	**Control**	Long-term dispatch for all sources	Long-term dipatch for area-wide sources	Long-term dispatch for local source	Long-term disptach for local source
	Data	All lower layer measurements and outside measurements	All intermediary layer measurements and outside measurements	All lower layer connected nodes measurements and outside measurements	Only the local data available
	Communication	FAN/NAN/BAN/WAN	WAN	FAN/NAN/BAN	-
Advantages		Easy to implement	Compromise between implementation and expansion	Easy to expand	Easy to expand
Drawbacks		Hard to expand	Complex communication system	Communication overhead	No communication

Control Issues A review of microgrid control literature has highlighted six issues that must be addressed for a sustainable, reliable, and stable microgrid operation. These are voltage and frequency regulation, power quality, transition between two modes of operation, microgrid protection, power flow management, and optimisation [6, 99, 100]. These issues are related to the layers and architectures presented previously, as shown in Table 2.6.

Voltage and frequency regulation and power quality are focused on electric regulation through a limited time frame and location linked to the electricity flow control, being near and fast. Transition operations and protection are focused on operation and protection through varying time frames and can operate on various geographic ranges. They can fit the not-so-fast and not-so-near middle range. Power flow management and optimization focus on energy management at a larger scale and time frame with the purpose of correction, anticipation, and improvement of real-time control. They fit the far and slow category.

Table 2.6: The information architectures and their relationship with control issues

Control architectures	Control Layers	V/I regulation	Power Quality	Transitions	Protection	Power Flow	Optimization
Centralized	Low	X	X		X		
	Intermediate		X	X	X	X	
	High			X	X	X	X
Hierarchical	Low	X	X	X	X		
	Intermediate			X	X	X	
	High				X	X	X
Decentralized	Low	X	X	X	X		
	Intermediate				X	X	X
	High						
Distributed	Low	X	X	X	X		
	Intermediate			X	X	X	
	High					X	X

Table 2.7 shows the control issues from a system dynamics (fast vs slow) and geographic perspective [100].

While these *control issues* must be thoroughly taken into account in the process of designing the microgrid, they constitute only a third of the information field issues.

Table 2.7: Summary of the different control issues in microgrids

Type of issue	Control issues	Description	Implementation
Fast and Near	**V/I regulation**	Regulation on a small time frame and a specific location [105]	Droop, PID, model predictive, fuzzy, neuro-fuzzy, learning, Virtual generator [106]
	Power Quality	Secondary loop regulation of the voltage and frequency [107]	P/Q control, Parallel BIC operation and harmonic mitigation [108]
Mid-Fast and Mid-Near	**Transitions**	Operation of the microgrid through disturbances [99]	Islanding detection, grid sync and the BIC management [109, 110]
	Protection	Operation of the microgrid through failure, notably short circuits [88]	Over/under I/V, fault detection, ground leakage, black start [64], cybersecurity and other [45, 111]
Slow and Far	**Power Flow**	Coordination and improvement of the energy fluxes [112]	Storage coordination, V/f improvement and demand response [113]
	Optimization	Entire microgrid improvement [114]	Economic dispatch, optimal load dispatch or prediction and forecasting algorithms [115, 116, 117, 118]

Data Issues A review of the literature on microgrid data has yielded two main issues, namely collecting and processing data [119, 120]. Data collection is broken down into smaller issues: acquiring, managing and storing, analyzing and disclosing the data [121, 122]. Data processing seeks to aggregate value to the collected data and is composed of six smaller issues: Volume, Variety, Velocity, Validity, Veracity and Volatility [123]. Table 2.8 summarizes the cross-analysis between these smaller issues treated in the literature.

Table 2.8: The seven V's management in the different data layers

Collection \ Processing Issues	**Volume**	**Variety**	**Validity**	**Volatility**	**Veracity**	**Velocity**	**Value**
Description	All the data	Diversity of sources	Accuracy of the raw data	Data storage	Quality of the data	Acquisition speed	Final aggregation
Acquisition [124]	X	X	X				
Management & storage [49, 125]				X	X	X	
Analysis [126]					X	X	
Disclosing [127, 128]							X
Protection [129, 130]	X	X	X	X	X	X	

Data issues are important to the sustainability of the microgrid as data can be used to guarantee the payment of the consumed energy [131], for a better understanding of the microgrid community [132], to make the enforce legislation and rules within the microgrid [133], to improve the power conversion within the microgrid [134], and many other applications [119].

All of these smaller issues must be considered when designing and operating a microgrid. Providing objective thresholds for their performance is paramount for achieving information sustainability within the microgrid from the data perspective.

Communication Issues The literature on communication links highlights three issues, namely data rate, geographic coverage area, and latency [97, 92]. The data rate is the capacity of a communication link to send out data quickly. The coverage area is the area that can be covered using a particular communication link with an acceptable cost. Latency is the time necessary for the communication link to transmit all the required data.

Figure 2.8 shows the communication links according to these issues. Numerous reviews on the subject have made it possible to represent a maximum of the links used with their own characteristics [135, 136, 137, 138]. It can be seen that communication link provide the lowest latency on the lowest layers, typically at the information node scale [139]. As more nodes are connected, the coverage area becomes more significant, thus sacrificing latency. Notable exceptions are fiber optics, the 5G and NB-PLC, all expensive and difficult to deploy.

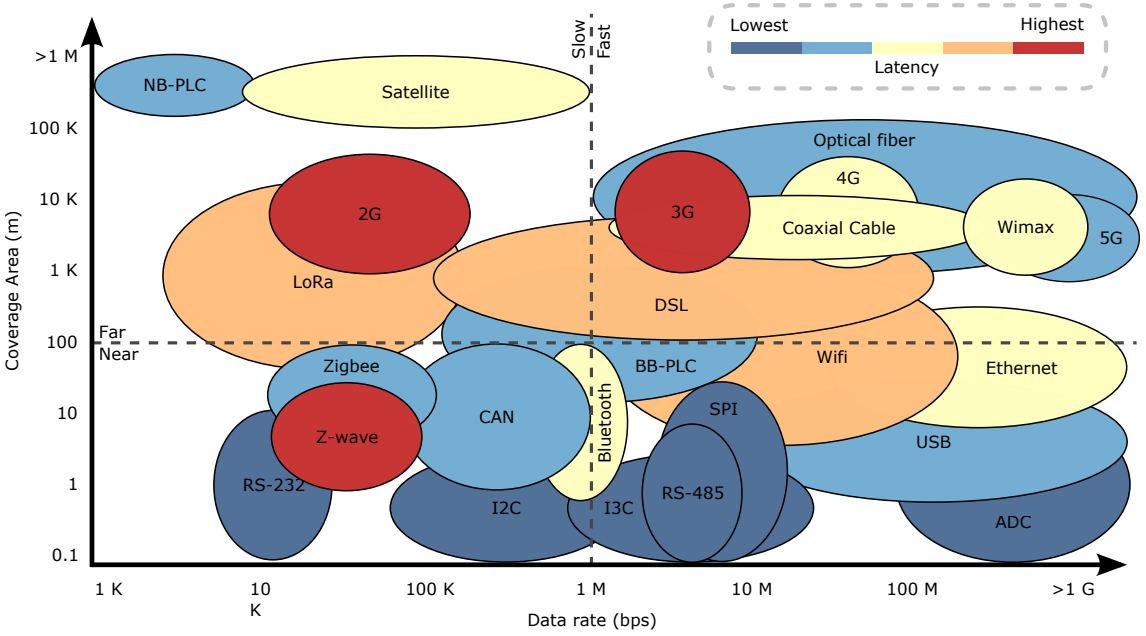


Figure 2.8: Classification of communication technology for microgrid inspired from [3]

A microgrid will invariably deploy a mix of different communication links. These links must be chosen wisely and objective thresholds must be set for their operation in order to track the sustainability of the information field from the the communication perspective.

2.2.5 Financial Field

The financial field of a microgrid is built on two aspects: governance and ownership [16]. Together, they provide the business model of the microgrid [140] and materialize its strategic objectives [141]. A business model can incur in a more or less centralized ownership [142], and the strategy can be implemented via a more or less centralized governance [143]. As a consequence, elements, architectures and issues are represented around these two aspects.

Financial Elements

microgrids are complex economic systems that integrate multiple types of assets. These assets can have one or several owners, including the community, utilities, private stakeholders or public companies. All these owners have different roles in the microgrid, various sorts of economic interactions with each other and diverse roles in funding the operation and maintenance of the microgrid on the long term [144].

The financial field elements regroup the different microgrid owners into four groups: consumers, producers, maintainers, and prosumers. Consumers simply use the microgrid, producers own energy assets, maintainers operate the microgrid, and prosumers own energy assets, use and maintain a part of the microgrid.

Each has a different ownership and governance role according to the business model of the microgrid.

Financial Architecture

In this work, the business model of a microgrid is the equivalent of its financial architecture. It is the tool through which the choices in terms of ownership and governance are made and the strategic objectives of the microgrid are set. From the literature, three primary microgrid business models or financial architectures can be pointed out: the Gathered business model, the Federated business model and the Networked business model [145, 146]. These architectures are illustrated in Figure 2.9 and detailed in Table 2.9.

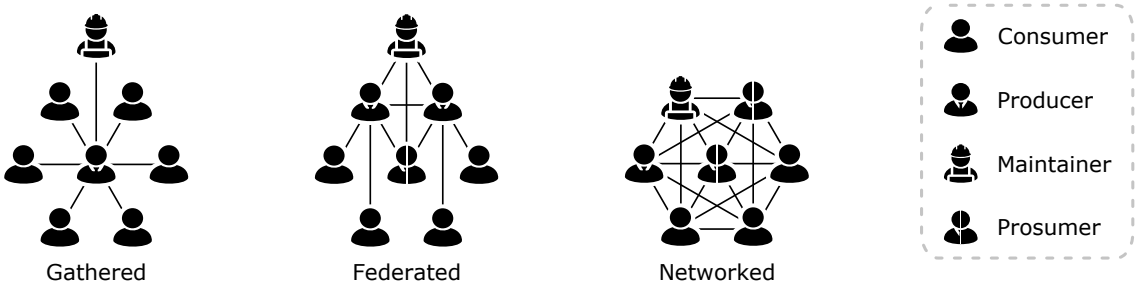


Figure 2.9: Representation of the different financial architectures

It is essential to notice some similarities between financial and information architectures. The level of centralization of the control, the data and the communication of the microgrid information nodes will have an impact on or be driven by the level of centralization of governance and ownership built into the financial architecture.

An example from the literature, a microgrid with a hierarchical information architecture will have a federated business model with different owners in different parts of the information pyramid [147, 148].

Financial Issues

The literature points to three key financial issues that must be addressed to ensure the economic sustainability of a microgrid: *techno-economic design, management, and long-term planning* [149, 28, 24, 150].

Techno-economic design is based on the calculation of Levelized Cost of Energy (LCOE) of the microgrid, expressed in equation 2.1.

$$LCOE = \frac{\sum_{t=1}^{n} \frac{CAPEX_t + OPEX_t}{(1+r)^t}}{\sum_{t=1}^{n} \frac{E_{prod_t}}{(1+r)^t}} \tag{2.1}$$

Where CAPEX_t is the Capital Expenditure in year t, OPEX_t is Maintenance and Operational Expenditure in year t, E_{prod_t} is the produced energy in year t, r is the discount rate and t is the time step going from 1 to n with n the lifetime of the system. The numerator of equation 2.1 represents the sum of the microgrid's costs and its denominator represents all the produced energy [151].

The challenge of techno-economic design is that there are assumptions regarding inflation, discount rates, downtime, and consumer payment default, among others which can totally alter the analysis of the feasibility of a microgrid. Many studies address the problem of rightly estimating the LCOE of a microgrid by highlighting the importance of a preliminary feasibility study of the microgrid [152, 153], providing a rigorous framework to understand better the local context and user needs [154, 155, 156]. The same goes for considering good practices that have been identified in different case studies to make better assumptions when estimating the chances of success of new microgrid projects [157, 158]. Finally, some studies go a little deeper by providing complete

Table 2.9: Classification of microgrid business model architecture

Architecture	Type	Description	Articles
Gathered - One main owner and maintainer of the microgrid [16]	Government utility ownership	The microgrid is owned by a public entity which takes care of its entire operation	[150]
	Third party ownership	The microgrid is owned by a private entity which takes care of its operation	[164]
	Anchor costumer business model	The owner cooperates with a large infrastructure to ensure a minimum energy consumption, such as a telecommunication tower	[173]
	Productive use of energy	The microgrid is linked with small businesses to ensure minim revenues	[174]
	Pay as you go	Solar Home Systems that are configured to deliver energy only if the consumer pays in advance	[175]
Federated - Multiple owners and maintainers of the microgrid [142]	Mixed ownership	Community own a part of the microgrid	[176]
	Community owned microgrid	The community owns the microgrid and takes care of its operation, but an entity exploits it	[177]
	Distributed model	A centralized entity owns the microgrid but shares the governance with the community	[178]
	Energy as a service	A centralized entity owns the microgrid and sells the energy as a service	[179, 180, 181]
	ICES	Integrated community energy system where the community participates in optimizing production by providing usage data	[182]
	Virtual power plant	Prosumers can be assembled in a bigger entity that faciliate the operation of the microgrid	[183, 184]
Networked - Community owns and operates the microgrid [185]	Standalone systems	Solar Home systems that are installed at the point of consumption	[78, 186]
	Decentralized prosumers	Prosumers are connected together in a decentralized way and each actor takes care of its own sources exchanging energy wit ha peer-to-peer market	[147, 148, 61]
	Blockchain based energy market	A decentralized peer-to-peer market where the price of energy fluctuates depending on the demand with some blockchains applications	[187, 61, 188, 189]
	Energy internet model	The community is a self-regulated market connected through a decentralized online tool	[190]

models for microgrid deployment [159], systemic methods for microgrid design [160, 151] or techno-economic optimization frameworks [161, 162].

Management issues handle the day-to-day cash flow of the microgrid [163, 164, 165, 166] and it can be summarized as maintenance cost and payment collection [167, 168]. Maintenance costs money but is imperative for the operation of the microgrid. It requires trained and qualified personnel together with the proper equipment [168, 28]. Collecting payments provides money to the microgrid, but it requires a particular organization and staff to ensure its correct execution [25, 169, 170]. More intelligent microgrids have automated and optimized payments, which greatly facilitate its collection such as pay-as-you-go [171, 172].

Long-term planning is at the same time forward-looking and backward-looking in nature [191]. Forward-looking planning tries to anticipate smooth transitions, especially if the acquisition of new equipment and expansion is involved [192, 193]. Backward-looking planning inventories recurring problems that must be addressed when new investments become necessary [194, 18, 195]. Several methodologies have been proposed in the literature to guarantee the integrity and the execution of the long-term planning process [196, 197].

The financial issues of a microgrid change over time and can be seen through a before, during and after deployment framework. Before deployment, a techno-economic study must address the most relevant aspects of ownership and governance to minimize the LCOE of the microgrid and maximize its likelihood of success. During deployment, management must strike a balance between minimizing maintenance costs, providing exemplary service and maximizing payment collection. After deployment denotes the long-term planning that must anticipate investments to replace equipment and expand services while keeping the evolution of the LCOE as low as possible.

2.2.6 Social Field

The social field of a microgrid relates to the complex socio-cultural aspects around its implementation [198]. The literature points to the notion of social acceptance as a simple way to express the shaping of a community to a new technology [199] despite a lack of consensus on its definition [200, 201]. In this work, social acceptance will be defined as "*not simply a set of static attitudes of individuals; instead, it refers more broadly to social relationships and organizations, and it is dynamic as it is shaped in learning processes.*" [202]. It will be used to determine the sustainability of the microgrid on the social field [203].

Social Elements

The most fundamental social element used in this work is social groups composed of individuals who share common interests [204]. The literature highlights three major social groups within the community of the microgrid [205, 206, 207, 208, 209]: individuals that use the microgrid, individuals that own and/or maintain the microgrid, and policymakers and/or representative. Some individuals will belong to several of these groups simultaneously and see their acceptance influenced through a multi-group perspective. The group containing all individuals who relate directly or indirectly to a microgrid will be called, in this work, the community of the microgrid.

Social Architecture

In order to better understand the acceptance of the microgrid community, it is necessary to establish an architecture linking, in a simple but realistic way the acceptance of all the groups that compose it. From the literature, this social architecture can be represented by three layers [210, 211, 7] which are the macro level, the meso level and the micro level. Figure 2.10 shows this architecture.

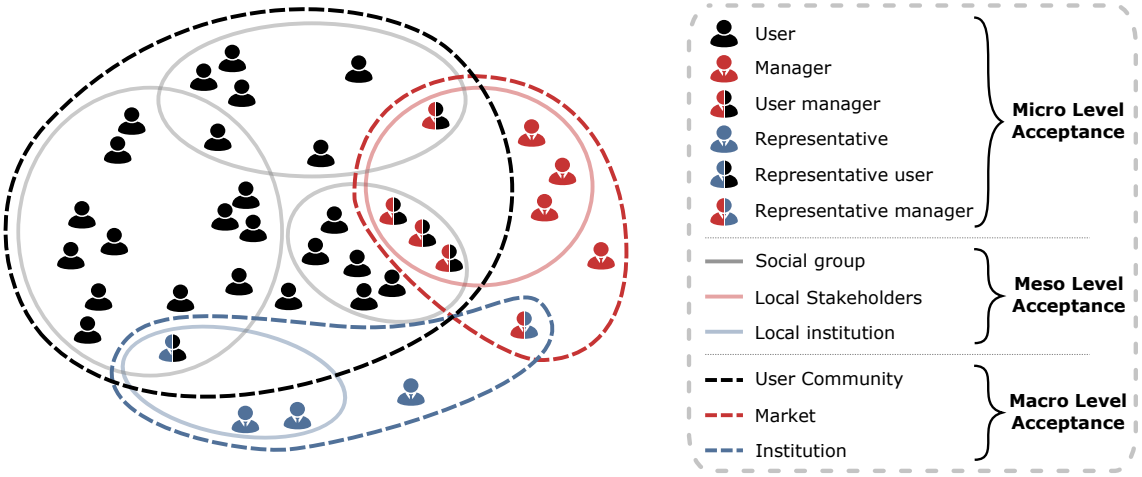


Figure 2.10: Representation of the different social architectures

The macro level refers to the entire community of the microgrid and represents its global acceptance. The meso level represents the different social groups described by the three social elements. The micro level represents the individuals themselves.

The microgrid's social acceptance will differ depending on the type of group and its layer [7]. At an individual level, social acceptance of the microgrid will be influenced by the social acceptance of the microgrid of the groups these individuals belong to. In turn, these different groups will shape their social acceptance of the microgrid based on the overall community acceptance of the microgrid [212]. These interactions are bi-directional, meaning they can go from a lower to a higher layer and vice-versa [213]. Table 2.10 describes the acceptance at each layer and for each social group.

Table 2.10: Cross-analysis between social elements, social architecture and their acceptance[7]

		Social architecture layers		
		Macro layer	**Meso layer**	**Micro layer**
Social elements	**User**	**Public acceptance:** all the groups containing end-users [214]	**Local public acceptance:** groups and organizations of end-users [215]	**End-user acceptance**: households and individual end-users [216]
	Owner / Manager	**Market acceptance:** all the groups containing owners/managers [217, 218]	**Local stakeholder acceptance:** groups and organizations of owners/managers [219]	**Owner/manager acceptance** local companies and/or individual owners/managers [210]
	Policy / Decision maker	**Socio-Political acceptance:** all the groups containing policy/decision makers [220, 221]	**Local political acceptance** groups and organizations of policy/decision makers [222]	**Policy/Decision maker acceptance:** local government bodies and/or individual policy/decision makers

Social Issues

The core social issues of a microgrid are the communication and the social interactions necessary to shape its social acceptance. The literature points to three defining factors in shaping social acceptance of microgrids: the knowledge about the microgrid, the rules that govern the microgrid and the perception of the microgrid [223, 224, 225, 7].

The knowledge of the community is about the level of social awareness on the technology of microgrids and the level of education of the individuals relating to the use or production of energy [226]. Communication shaped around the comprehension of the technology is vital for its social acceptance because it allows the community to understand and correctly use the technology. Microgrids are an excellent tool to contribute to the development of remote communities by unlocking and teaching new skills to local people [158]. The quantification and propagation of this knowledge are highly beneficial for the development and operation of microgrids [227].

The perception of the microgrid relates to how the technology changes the narrative of the individuals and the groups [228, 229]. This community perception will be strongly related to how well the microgrid improves or does not improve the local quality of life [224] but also the general confidence of the community in the entire microgrid system [202]. Communication about the benefits of the microgrid [230] and transparency on its nuisances requires a highly qualified management team which takes into account the vision of the community and frame the technology within it [231].

The rules that govern the microgrid should contain good planning for a clear vision of the project, transparent and sustainable ownership, appropriate pricing, and respect for community territoriality and culture in its governance [39]. The local or national policy is often a barrier to microgrid development and policy regulations must address social needs and feedback [232]. Communicating about these rules allows the community to integrate the technology more easily into their daily habits, raising their acceptance [233].

It must be stressed that this communication effort is permanent. The management team of the microgrid must integrate into their work the effort of addressing the knowledge, the rules, and the perception of the different groups within the microgrid at their different layers. This requires different strategies, some of which are summarized in table 2.11.

Table 2.11: Different strategies for social acceptance based on the elements acceptance

		Social acceptance issues		
		Knowledge	**Rules**	**Perception**
Social acceptance elements	**User community**	Educational events, door-to-door campaigns, usage tutorials, community maintenance engagement [234, 235]	Preliminary community survey, asking community feedback framework on rules acceptance [236]	Local stakeholder/ manager presence, consumption monitoring system, community ownership integration [237, 238, 239]
	Market	Training sessions, microgrid knowledge sharing locally or on the web [234, 235]	Detailed business model documentation, microgrid design co-construction with the user community [240, 241]	Detailed planning of the microgrid project, Supervision of rules compliance,financial feasibility study [242, 243]
	Institution	microgrid policy training, energy educational programs, practical experience report [33]	Impact workshop design, detailed ownership and governance definition [244]	Lobbyist or association campaign, presence of representatives in the field [63]

These strategies require ressources and can be seen as a social maintenance cost of the technology. Their deployment must be taken into account with the other issues of the microgrid to ensure its sustainability.

2.2.7 Framing sustainability in microgrids

Figure 2.11 summarizes all the elements, architectures and issues of all the fields explained in the previous section.

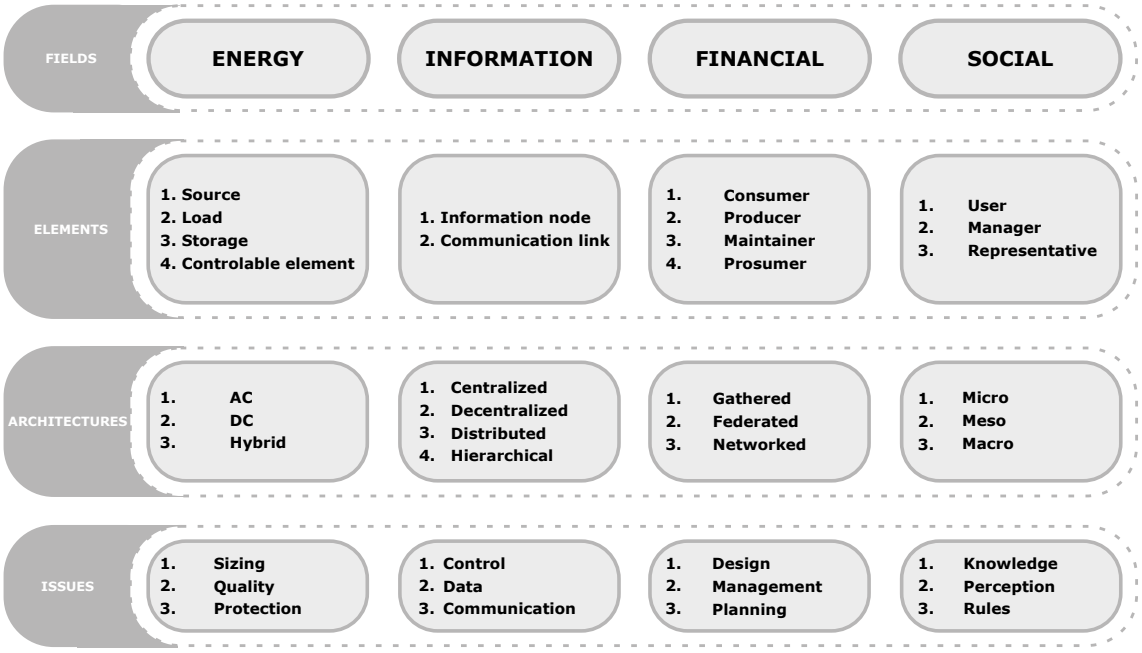


Figure 2.11: Structure of the review

This work proposes the definition of *sustainable microgrid* as one that is capable of addressing all the issues of its fields in a *satisfactory manner* at the time of analysis. The definition of what is satisfactory for each field depends on each case, leaving the possibility of qualitative analysis. However, satisfaction must be translated into thresholds compatible with analytic tools. Table 2.12 provides an overview of each field's issues.

Table 2.12: List of the issues of microgrid sustainability

Component	Issues	Description	Reference
Energy	Sizing	Provide a compromise between power production capacity and foreseeable load	[70, 71, 72]
	Quality	Cost-benefit analysis of the equipment used to build the microgrid	[80, 81, 82]
	Protection	Number of protection elements for different kinds of faults and the overall cost linked to their presence or absence from the microgrid	[87, 65, 64, 88]
Information	Control	Taking into account all 6 control issues listed in table 2.7	[6, 99, 100]
	Data	Finding a balance between data collection and data processing listed in table 2.8	[119, 120, 121, 122, 123]
	Communication	Finding a balance between data rate, geographic coverage and latency as shown in figure 2.8	[97, 92]
Finance	Design	Address the most relevant aspects of ownership and governance in order to minimize the LCOE of the microgrid	[151, 152, 153, 154, 155, 156]
	Management	Strike a balance between minimizing maintenance costs, provide good service and maximize payment collection	[163, 164, 165, 166, 167, 168]
	Planning	Anticipate investments in order to replace equipment and expand services while keeping the evolution of the LCOE as low as possible	[191, 192, 193, 194, 18, 195]
Social	Knowledge	A communication shaped around the comprehension of the technology	[226, 158, 227]
	Rules	Communicating about the vision of the project, its ownership structure, its pricing mechanism and its relation with the local culture	[39, 232, 233]
	Perception	A communication about the benefits of the microgrid, transparency on its nuisances taking into account the vision of the community	[228, 229, 224]

The bibliographical study of the various reviews on the subject of microgrids is presented in Table 2.13. This work underpins all the descriptions of this chapter. It is clear that no article has managed to address and assemble all the elements, architectures, and issues related to microgrids together. This constitutes a core contribution to this thesis.

2.2.8 Conclusions from the proposed experimental framework

The *experimental framework* proposed in this chapter has built the representation of the energy and community microgrid aspects through the same formalism. This top-down approach based on the literature review allows to create homomorphic representations of a microgrid, which are the basis of building models that are inter-operable. The next chapter will use this experimental frame to build three different models: a structural, a generative and a replicative model. The objective is to explore how the M&S theory can be leveraged to explore this unified modeling approach and better represent the complexity of the multi-disciplinary nature of microgrids in the study of their sustainability.

Table 2.13: Recent literature review on microgrid relating to the topics addressed in this review.

	Energy						**Financial**						**Information**							**Social**						**Type**	
References	Quality	Sizing	Protection	AC	DC	Hybrid	Sizing	Planing	Management	Gathered	Federated	Networked	Control	Data	Communication	Centralized	Decentralized	Distributed	Hierarchical	Knowledge	Perception	Rules	Micro	Meso	Macro	Analytic	Systemic
[22]	x			x	x	x		x		x	x	x	x	x	x	x	x	x	x						x		x
[51]	x		x	x	x	x							x													x	
[39]	x	x					x	x	x											x	x		x	x	x		x
[120]														x												x	
[146]							x	x	x	x	x										x	x			x		x
[33]	x	x					x	x	x				x	x						x	x	x					x
[245]				x	x	x							x		x	x	x	x	x								
[96]													x	x	x	x	x	x	x								
[246]	x			x	x	x			x				x		x	x	x	x	x							x	
[16]	x								x	x	x										x	x	x		x	x	
[206]	x	x	x						x		x		x	x	x					x	x	x	x	x	x		x
[247]	x	x	x				x	x	x				x	x						x	x	x					x
[89]	x		x					x	x				x	x	x										x	x	
[29]	x	x	x	x	x	x	x	x	x				x								x				x	x	x
[35]	x	x					x	x	x				x	x	x					x	x	x	x	x	x		x
[6]	x			x	x	x							x		x	x	x	x	x							x	
[248]	x	x					x	x	x				x								x						x
[249]	x								x	x	x	x		x	x										x	x	
[3]													x	x	x	x	x		x							x	
[24]	x	x	x				x	x	x				x	x						x	x	x			x		x
[34]	x			x	x	x	x	x	x		x		x		x	x	x	x	x			x				x	x
[40]	x		x	x	x	x		x	x				x			x	x		x							x	
[250]	x		x	x	x	x							x		x	x	x									x	
[232]							x	x	x													x			x	x	
[84]	x	x	x	x	x	x		x					x						x							x	
[227]	x	x					x	x	x					x						x	x	x	x				x
[99]	x			x	x	x							x			x	x	x	x							x	

Chapter 3

Model: Formalisms and their microgrid applications

The previous chapter has proposed a novel experimental framework to describe *community* and *energy* microgrids. Based on this framework a single formalism was applied to describe two distinct microgrid perspectives. The objective was to lay the foundations to explore the creation of inter-operable microgrid models. The joint experimental frame has described the microgrid into four fields: energy, information, financial and social. Each field was described by elements, architectures and issues. Elements are associated via architecture to address the issues of a given field. Sustainability was defined as addressing all the issues of all fields in order to keep their indicators below an arbitrary thresholds set by those deploying the microgrid.

This chapter will explore the joint modelling of a microgrid through three different models: a structural, a generative and a replicative model. The structural model will explore how to replicate the generic structure provided by the experimental framework from the previous chapter. The generative model will explore how to create an inter-operable model whose behaviour is built from I/O functions that can exchange data among themselves despite belonging to different fields of expertise. The replicative model will explore the coupling of different simulators whose I/O are build from real-world data to explore its capacity to replicate the model behaviour of different fields of expertise which were not originally meant to operate together.

This chapter focuses on formally describing the models. **The reader is advised to remember that the results from the simulations of the models presented in this chapter are available in chapter 4.** This choice was made to keep the thesis aligned with the way the M&S theory is presented and to formally separate the process of creating a model and simulating it. Figure 3.1 illustrates this theoretical progression.

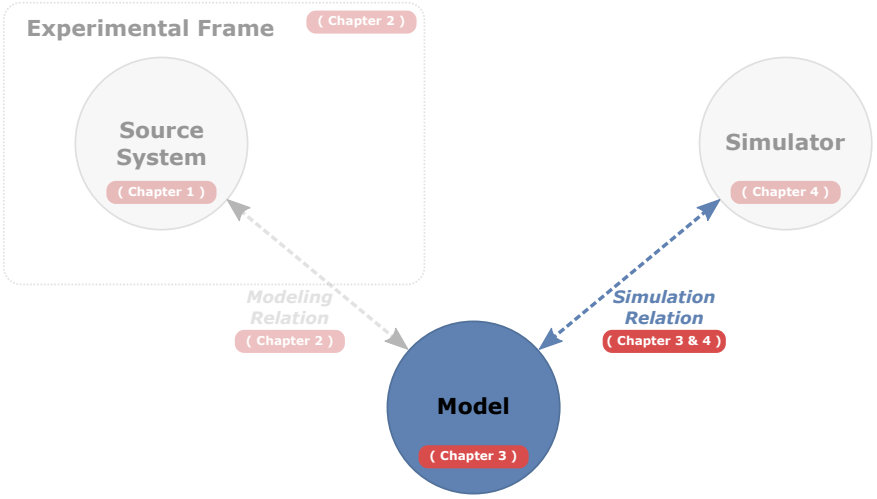


Figure 3.1: Scope of the Chapter 3

3.1 M&S Theory: Modelling formalism

3.1.1 What are models?

Previous chapters have shown that a *source system* is scrutinized via an *experimental frame* yielding *input/output data*. Coupled with the notion of components and their initial state, a system can then be described by functions that provide a unique output trajectory for a given input trajectory. These states can then be leveraged to create output events, which can then drive the process of generating data, effectively creating a model.

This chapter will dive deeper into the available modeling formalisms and link them back to the data generated by the experimental frame of chapter 2. First, the levels of system specification will be revisited and expanded, allowing for expressing all the elements that compose a given model. Then, the component coupling model will be detailed, showing two possible techniques. Finally, a unifying formalism is presented.

These modeling concepts and techniques will be applied to the experimental framework to create a first lumped model. From these initial models, a system analysis process will be taken in two subsequent model iterations, effectively going from the upper to the lower levels of system knowledge.

Figure 3.2 we can identify the transformation of our systemic microgridstructure into a PDEVS model.

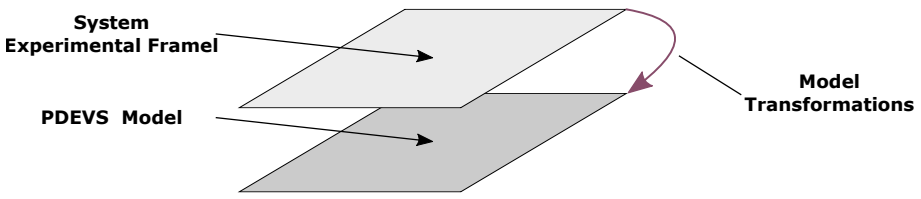


Figure 3.2: Model transformation layers within this part

3.1.2 More on time and trajectories

As stated previously, the fundamental notion of a dynamic system is the passage of time [5]. The choice of the type of representation of the time base is directly influenced by the type of source system to be modeled. The essential formal notions related to time are its *bounds* and *time base*. The boundaries of time define its beginning and end but also if there are gaps in the time representation. The time base defines if the time is continuous or discrete. Accurately defining time has a direct impact on state transition, events and overall system behavior.

Trajectories are the means of time tagging input/output pairs. There are four types of segments: continuous, constant, event, and sequences. Continuous segments are made of real numbers and can be split into piecewise sections. Constant segments are made of integers whose values remain unchanged for a certain length of time. Event segments map the occurrence of an event at given points in time, with all the rest being mapped to non-events. The sequence is the union of all segments over a discrete time base.

Different types of source systems will require different time bases with different bounds, leading to different sequences built from various segments.

3.1.3 More on System specifications - from data to system

A model contains different elements depending on which level of system specification the modeler is working in. Table 3.1 summarizes which elements belong to the different levels.

Table 3.1: Detailed system specification hierarchy [5]

Level	**Name**	Time	Input	Output	In. Seg.	I/O R.	I/O F.	State	St. Tr. F.	Output F.
0	I/O Frame	X	X	X	-	-	-	-	-	-
1	I/O Behavior	X	X	X	X	X	-	-	-	-
2	I/O Function	X	X	X	X	-	X	-	-	-
3	I/O System	X	X	X	X	-	-	X	X	X

At lower levels, the modeler must focus on the inputs and outputs of the system, together with its associated time points. Once relations are built (I/O R.), input segments can be used to experiment with the level 1 model. With enough relations, functions can be mapped leading to a more robust representation. Finally, states can be defined based on the system function knowledge. These are then specialized between functions related to state transitions and functions related to the output.

The contribution of each level of specification is given in figure 3.3.

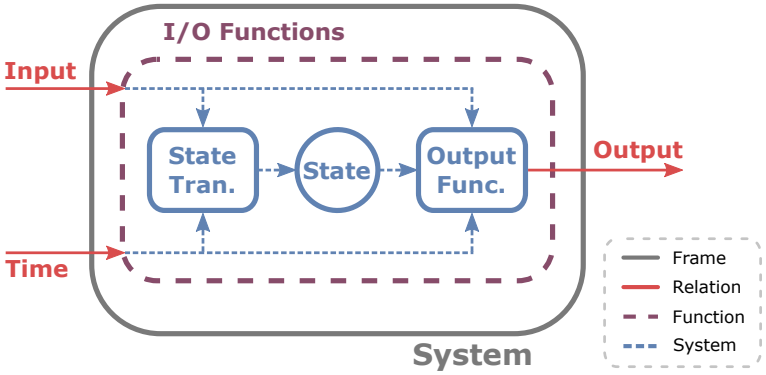


Figure 3.3: Detailed representation of a system with the contribution from each level of specification

When dealing with a system, a modeler will be confronted with a series of issues from data to function definition. Handling these issues mean using one of the possible modeling formalisms described below.

3.1.4 More on System specifications – component coupling

A source system can be represented by multiple interconnected components, each contributing to its overall behavior. These smaller components, known as basic components, can be used to create more complex models of the system. In the M&S theory of Ziegler, there are two types of models that can be created using this approach: *multi-component systems* and *network of systems*. The former refers to a system made up of multiple components that work together to achieve a specific goal, while the latter refers to a network of interconnected systems that interact with each other to produce a desired outcome.

Multi-component systems is based on the idea of *influence*. Components can influence (*influencers*) or be influenced (*influencees*) by others. Each local state transition function takes the state set of the influencers and maps them into new states of the influencees. Each local output

function also takes into account the state set of the influencer when computing its contribution to the overall system. This coupling is illustrated in figure 3.4.

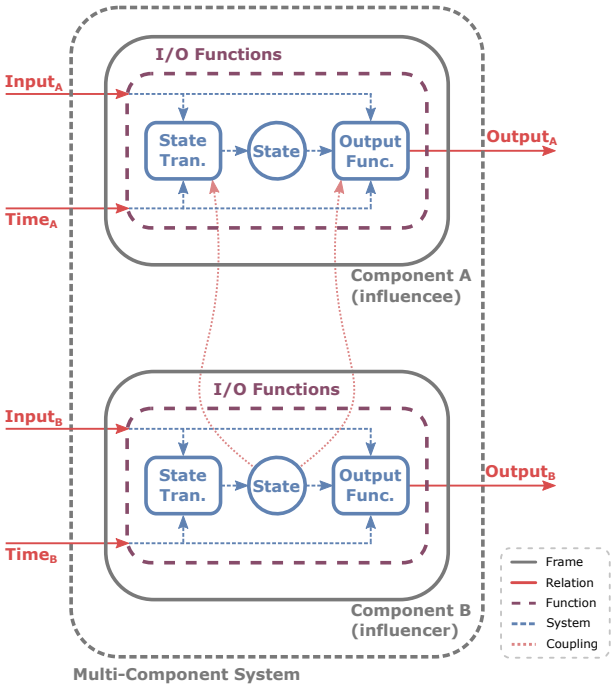


Figure 3.4: A multi-component system

Network of systems is based on the notion of *ports*. Components are coupled together via their I/O ports. This method requires a *coupling map* composed of external input, external output and internal couplings. They connect all the component ports between each other and with the general I/O ports of the network. This coupling is illustrated in figure 3.5.

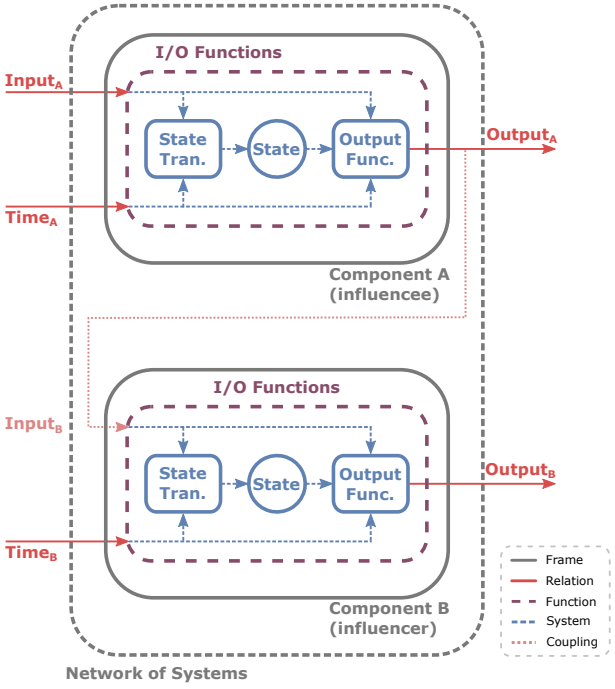


Figure 3.5: A network of systems

The obvious difference between the two approaches is the strong *modularity* offered by network of systems. Multi-component systems might be a stepping stone in the modeling process of a

complex problem. There are methods for converting multi-component systems into a network of systems, making both techniques equally interesting for modelers.

Notice the fact that the time input for components was left deliberately separated. Some source systems might be composed of multiple types of sub systems, which might be best represented in different *time bases*. To handle such complex systems, different modeling formalisms can be used and coupled together.

3.1.5 Modeling formalisms

There are three basic formalisms used to describe any system, namely, differential equation systems, discrete time systems and discrete event systems. They are summarized in table 3.2.

Table 3.2: System specification formalisms

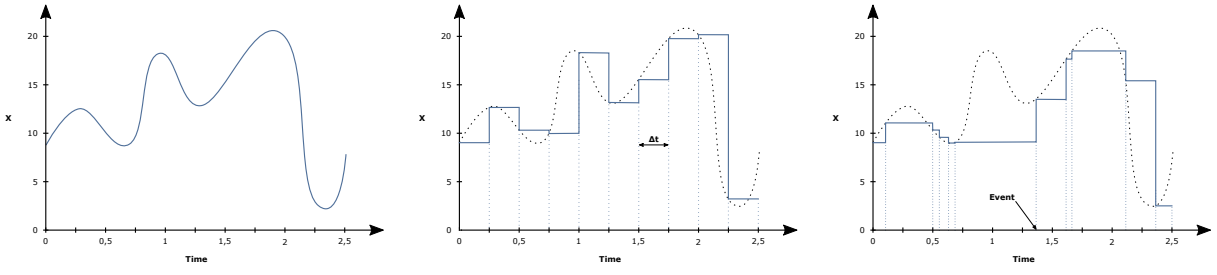
System Specification	Time	States	Implementation
Differential Equation	Continuous	Continuous	Differential Equations solved by numerical integration
Discrete time	Discrete	Discrete	Fixed-time recurrent calculations using previous states and inputs
Discrete event	Continuous	Discrete	Event scheduler

Differential equations systems have continuous states and time as represented in figure 3.6(a). Widely used in natural sciences, these systems are heavily based on ordinary differential equations. Their simulators are typically built on numerical integration methods, whose main issues are accuracy and stability.

Discrete time systems have well-defined time steps lengths and its states remains constant between two steps as represented in figure 3.6(b). States are updated based on the states of the previous time step and the system inputs. Computations are done at every step event when no input or state changes took place. Its main issues are calculation costs and optimization.

Discrete event systems have continuous time but its states are only updated when events occur as represented in figure 3.6(c). This method focuses on the *interesting* points in time. Its main issue is how to *find and schedule events* in time.

In M&S theory, discrete event system specification (DEVS) is used as a mean to create models that can interface all types of systems.



(a) Evolution of a state variable of a system in a continuous time formalism

(b) Evolution of a state variable of a system in a discrete time formalism

(c) Evolution of a state variable of a system in a discrete event formalism

Figure 3.6: The different time formalism

3.1.6 The DEVS formalism

The DEVS formalism is a component-based modeling formalism, especially for complex systems, which allows to describe a system capable of wrapping hybrid models. It proposes the semantic elements necessary to make a viable and functional model. The basis is generic enough to allow the integration of many different models in terms of their temporal representation or their raw form.

The formalisms of the DEVS family define atomic component models as the basis for expressing behavior. Coupled components are also defined as aggregating other components, themselves atomic or coupled.

This work will then use the Parallel DEVS (PDEVS) extension which brings advances on the problems of conflicts during the execution of several models simultaneously. This formalism can be used to interface different types of models, creating a flexible method for modeling multi-disciplinary systems which is the case for the study of sustainability in microgrids.

3.1.7 The PDEVS atomic model

The basic structure of an atomic model is summarized in figure 3.7, with its associated equation system shown in equation 3.1. In mathematical terms, this system is an ordered list of elements or a tuple.

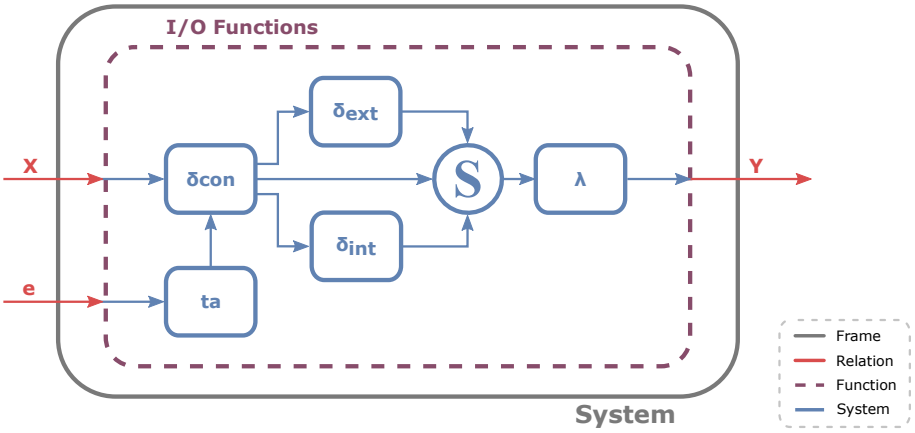


Figure 3.7: A DEVS-based model

$$M =< X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta > \tag{3.1}$$

Where :

$X = \{(p, v) \mid p \in IPorts, v \in X_p\}$ is the set of input ports and values;
$Y = \{(p, v) \mid p \in OPorts, v \in Y_p\}$ is the set of output ports and values;
S is the set of sequential states;
$\delta_{ext} : Q \times X \to S$ is the *external state transition function;*
$\delta_{int} : S \to S$ is the *internal state transition function;*
$\delta_{con} : Q \times X \to S$ is the *confluent transition function;*
$\lambda : S \to Y$ is the output function;
$ta : S \to R_0^+ \cup \infty$ is the *time advance function;*

The DEVS formalism main contribution is to break down the state transition function into three sub-functions, namely *External State Transition* (δ_{ext}), *Internal State Transition* (δ_{int}) and *Time Function* (ta).

In PDEVS, X and Y represent the inputs and outputs of the component, respectively. The notion of ports is generally integrated within the formalism to be able to dissociate each of the inputs and outputs of a component, rater than dealing with complex sets gathering all inputs or

outputs. With ports, each input and output can be associated with a name. This allows, for example, to interface a component with other components or to specify by name the couplings within a coupled component. Here, **IPorts** and **OPorts** represent respectively the set of input and output ports. X_p (resp. Y_p) is the set of admissible values for the input (resp. output) port p. Thus, X (or Y) represents a set of couples (port, value), which represent all possible inputs (or outputs) for the component.

S allows defining the set of sequential states of the component, i.e. the values that can be reached by the component's state. S can be translated into a list of variables, each associated with a set of definitions, i.e. the possible values for the variable. The Cartesian product of these definition sets represents the state space that the component can evolve within.

The δ functions govern the state transitions of the component according to an event triggering its evolution. These triggers can be internal (triggered by time) or external (triggered by an event) coming from one of the component inputs. The conflict between two simultaneous events is managed by the confluence function δ_{con}, which does the arbitration in this specific case. According to the transition functions, the set of inputs of a considered δ function will be either S or Q, with $Q = (S, e)$, e representing the time elapsed in the state since the last state change. Thus, δ_{ext} and δ_{con} take this time into account, whereas δ_{int}, which is already triggered by the time flow, does not need to consider e as it is known that $e = ta(s)$ when δ_{int} is triggered. δ_{ext} and δ_{con} also have X as an input, in order to be able to take the input into account in the state change.

The λ function defines how the component will produce events on its output ports in reaction to an internal event. At each state change by internal transition, the λ function is called and determines if an event should be emitted on an output port.

The ta function defines the time associated with a state of the component to trigger an internal event. For a state s after $ta(s)$ units of time the δ_{int} function will be called to change the state of the component. There are two particular values for ta : 0 and ∞. When $ta(s) = 0$, the time will not elapse before the change of state (instantaneous change from the point of view of the model) because the function ta is imminent. If $ta(s) = \infty$, then the δ_{int} function will never be called in this state, the only possible change coming from an external transition.

This system begs the question: what happens when an input arrives at the same time as the time function finishes counting? In other words, what has access to the states first, the internal or the external transition function? These questions are addressed by the simulator in the next chapter.

3.1.8 The PDEVS coupled model

A PDEVS coupled model is also a finite ordered list of elements or a tuple. Figure 3.8 illustrates a coupled model.

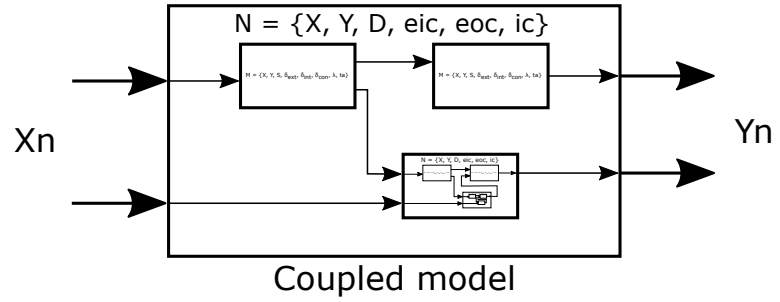


Figure 3.8: Graphical representation of PDEVS

The system of equations that describes a coupled model is given in equation 3.2.

$$N =< X, Y, D, \{M_d\}, EIC, EOC, IC > \tag{3.2}$$

Where :

$X = \{(p, v) \mid p \in IPorts, v \in X_p\}$ is the set of input ports and values;

$Y = \{(p, v) \mid p \in OPorts, v \in Y_p\}$ is the set of output ports and values;
D is the set of components;
$\{M_d\}$ is the set containing models for components in D;
$EIC : \{(N, X_N), (M_d, X_{M_d})\}$ is the set of coupling with external input;
$EOC : \{(M_d, Y_{M_d}), (N, Y_N)\}$ is the set of coupling with external output;
$IC : \{(M_d, Y_{M_d}), (M_f, X_{M_f})\, d \neq f\}$ is the set of internal coupling;

The notion of input and output with X and Y is the same as for the atomic components, the coupled components has the same possibilities of interfacing.

D defines the set of components that are contained within the coupled component, it gives a name or a reference to each model. Associated with these references, $\{M_d\}$ contains their respective model. Since the input and output of atomic and coupled models are defined in the same way, it is quite possible to have a coupled model contained within a coupled model. Indeed, the property of *closure under coupling*, expressing that a coupled component can be matched to an equivalent atomic component, has been proven on PDEVS, guarantee that such a hierarchic description will still result in a valid PDEVS component.

Finally EIC, EOC, IC detail respectively the links with the external inputs, the links with the external outputs and the internal links of the coupled model. These three tuples define the interfacing of the coupled model with the different models that compose it.

3.1.9 Examples of DEVS-based models

The mathematical notation that underpins DEVS based documentation can be daunting at times, requiring a more user-friendly representation. This section introduces the graphic representation that will be adopted throughout this work, given by figure 3.10.

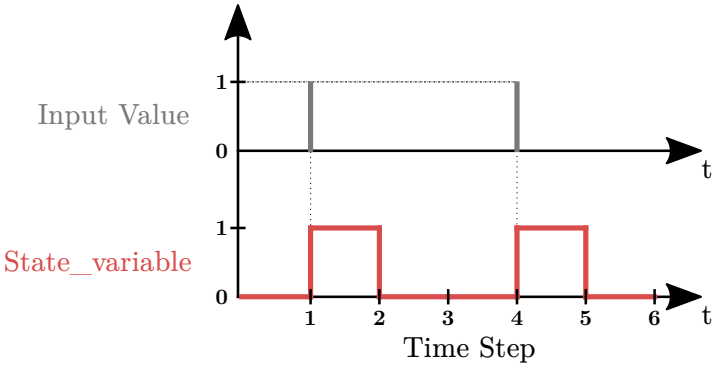


Figure 3.9: Chronogram of the behaviour of the *State_variable* value

Here is represented a simple system that adds the value of the input it receives to its internal *State_variable* value and then subtract 1 to this new *State_variable* value after a time step of 1. The behaviour of this system is described more precisely in a graph in figure 3.9, with the *State_variable* in this case being 0 and the input value being 1.

To simplify the description of this system via PDEVS, we will first adopt a state machine that describes the model, as shown in 3.10.

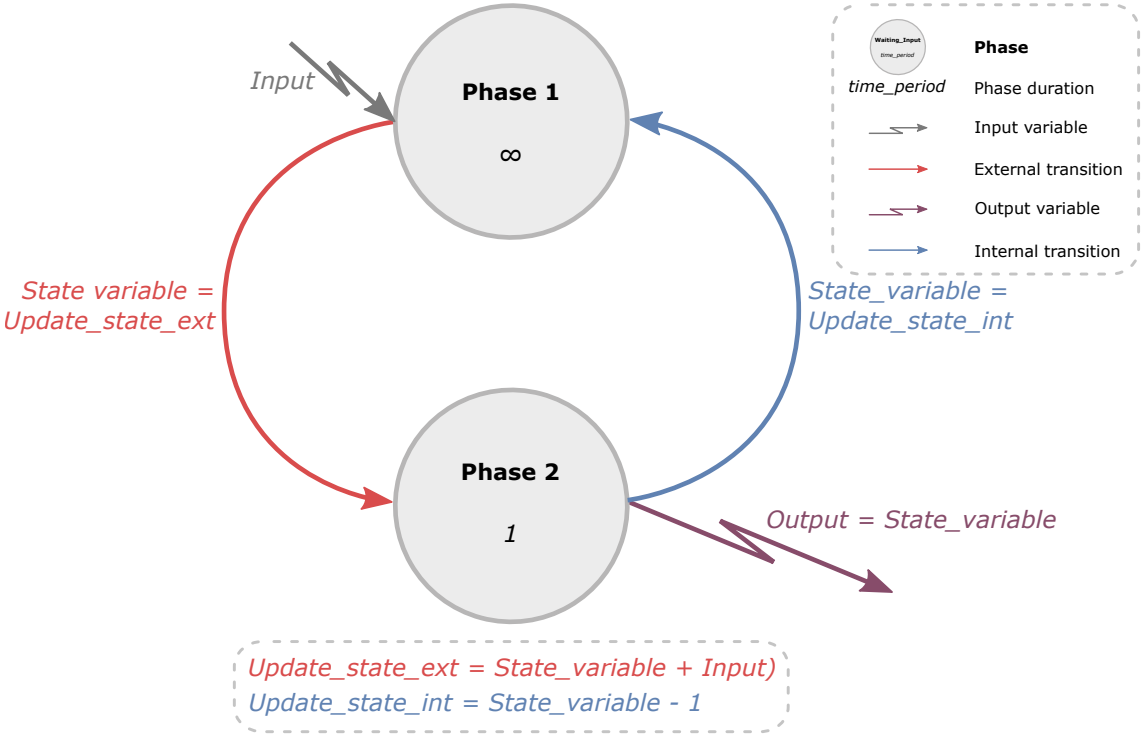


Figure 3.10: A graphical representation of DEVS model state dynamic

The proposed example model state is composed of two variables that represent the possible values of the system's state. The first variable, called *Phase* can take on either of two values, "$Phase_1$" or "$Phase_2$" and is used to model the system as a state machine. The second variable, called *State_variable* is a numerical variable that is used for calculating state changes. To better understand the dynamics of this system represented in figure 3.10, we will describe its internal workings in more detail.

Initialised at the phase "$Phase_1$", the model waits an infinite amount of time as *ta* in this phase is ∞. This means that the model will never change its state from an internal event, but only as a result of an external event. When the model receives an input variable through its unique input port *Input*, its triggers an external transition. The reception of an external event on port *Input* is depicted by the gray arrow. This event updates the state, changing the value of *Phase* to "$Phase_2$", as depicted by the red arrow. The value of *State_variable* which is obtained using the *Update_state_ext* function, as depicted by the red text. Here, *Update_state_ext* takes as an input the value received on its input port and the current value of *State_variable* and return the addition of these two values.

The *ta* value associated with the "$Phase_2$" state is set to 1, which means that the model becomes imminent after 1 time unit has elapsed. When this time is elapsed, two things occur: the emission of an event on the *Output* port, depicted by the purple arrow, and an internal transition depicted by the blue arrow. Here, the value of the output event is set to the current value of *State_variable*. The internal transition updates both state variables again. The *Phase* variable is set to "$Phase_1$", as depicted by the blue arrow, and the *Update_state_int* function is executed to provide the new value of *State_variable*. This function only takes the current state as an argument, as there is no incoming message on an internal event. Here, *Update_state_int* return the current value of *State_variable* subtracted by 1.

After that, the model has returned to the "$Phase_1$" state with an infinite *ta* and will wait for another input variable to be received. This process repeats indefinitely, allowing the system to exhibit complex behavior.

This graphic representation is intended at being understandable by people familiar with state machines, but does not expose all information about the model. E.g. the definition sets for ports and variables does not appear on that representation. For the same model, we provide a mathematical representation using formal PDEVS in equation 3.3.

$$PDEVS_{example_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.3}$$

Where :

$X = \{(Input, v) \mid v \in \mathbb{R}\}$
$Y = \{(Output, v) \mid v \in \mathbb{R}\}$
$S = \{(\boldsymbol{Phase}, State_Variable) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Phase_1}\text{”}, \text{“}\boldsymbol{Phase_2}\text{”}\}, State_Variable \in \mathbb{R}\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Phase_1}\text{”}, State_variable), e, (Input, v)\big) = (\text{“}\boldsymbol{Phase_2}\text{”}, Update_state_ext(State_variable, v))$
$\delta_{int}(\text{“}\boldsymbol{Phase_2}\text{”}, State_variable) = (\text{“}\boldsymbol{Phase_1}\text{”}, Update_state_int(State_variable))$
$\lambda(\text{“}\boldsymbol{Phase_2}\text{”}, State_variable) = (Output, State_variable)$
$ta(\text{“}\boldsymbol{Phase_1}\text{”}, State_variable) = \infty$
$ta(\text{“}\boldsymbol{Phase_2}\text{”}, State_variable) = 1$

The input port of the example model, belonging to X, is named $Input$ and can take values from the set of real numbers, denoted as $\mathbb{R}$. The output port, belonging to Y and named $Output$, also has the ability to take values from $\mathbb{R}$. The state of the model, denoted as S, consists of two variables $\boldsymbol{Phase}$ and $State_variable$. $Phase$ correspond to the round states of the state machine, and can have two possible values: $Phase_1$ and $Phase_2$. $State_Variable$ can take values from $\mathbb{R}$. The initial state of the model, denoted as q_{init}, is set such that the phase is equal to $Phase_1$ and the state variable is initialized with an arbitrary value $Initial_state$. The external transition function, denoted as δ_{ext}, uses the $Input$ port value to calculate the new state using the function $Update_state_ext$ and changes the phase from $Phase_1$ to $Phase_2$. This transition is only called when there is an input present. The internal state function, denoted as δ_{int}, calculates the new value of $State_variable$ using function $Update_state_int$ and changes the phase from $Phase_1$ to $Phase_2$. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function is triggered. When this function is implicit, such as with this example, its adapts it default behavior which, as defined by Zeigler [5], is to obtain the new state by applying $\delta_{ext}(\delta_{int}, 0, X)$. The output function, denoted as $\lambda(Phase, State_variable)$, sends the value of $State_variable$ through the $Output$ port to any connected component. The phases are linked to specific time-advances, with $Phase_1$ linked to an infinite time-advance that represents a wait and $Phase_2$ linked to a time-advance of 1.

To go further we will create a slightly more complex model by coupling three atomic together which we will call $Example_model_1$, $Example_model_2$ and $Example_model_3$. These models will be identical, except for $Example_model_1$ which has a different initialization as described in equation 3.4 in order not to have all the models in infinite ta which would mean they all wait for each other.

$$\begin{aligned} s_0 PDEVS_{example_model}1(\boldsymbol{Phase}, State_variable) &= (\text{“}\boldsymbol{Phase_1}\text{”}, 1) \\ s_0 PDEVS_{example_model}2(\boldsymbol{Phase}, State_variable) &= (\text{“}\boldsymbol{Phase_2}\text{”}, 0) \\ s_0 PDEVS_{example_model}3(\boldsymbol{Phase}, State_variable) &= (\text{“}\boldsymbol{Phase_2}\text{”}, 0) \end{aligned} \tag{3.4}$$

We therefore connect the output of $Example_model_1$ to the input of $Example_model_2$, the output of $Example_model_2$ to the input of $Example_model_3$, and the output of $Example_model_3$ to the input of $Example_model_1$, like depicted in figure **??**.

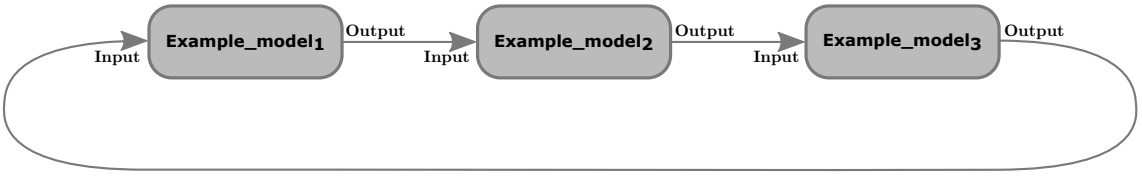


Figure 3.11: Coupling of the three example models

The equation 3.5 describes the coupled model with the connections we have mentioned.

$$PDEVS_{root_model} = (X, Y, D, \{M_d\}, EIC, EOC, IC) \tag{3.5}$$

Where :

$$\begin{aligned}
&X = \{\} \\
&Y = \{\} \\
&D = \{Example_model_1,\ Example_model_2,\ Example_model_3\} \\
&\{M_d\} = \{PDEVS_{Example_model_1}, PDEVS_{Example_model_3}, PDEVS_{Example_model_3}\} \\
&EIC = \{\} \\
&EOC = \{\} \\
&IC = \Big\{\Big((Example_model_1, "Output"), (Example_model_2, "Input")\Big), \\
&\qquad \Big((Example_model_2, "Output"), (Example_model_3, "Input")\Big), \\
&\qquad \Big((Example_model_3, "Output"), (Example_model_1, "Input")\Big)\Big\}
\end{aligned}$$

This coupling of these model, with the initialization that have been made, will see their *State_variable* change as described in the chronogram figure 3.12. The value will transferred from one model to another and the *State_variable* value will go back to 0 each time the *Update_state_int* function is called, since the variable sent is initialized to 1 in $Example_model_1$.

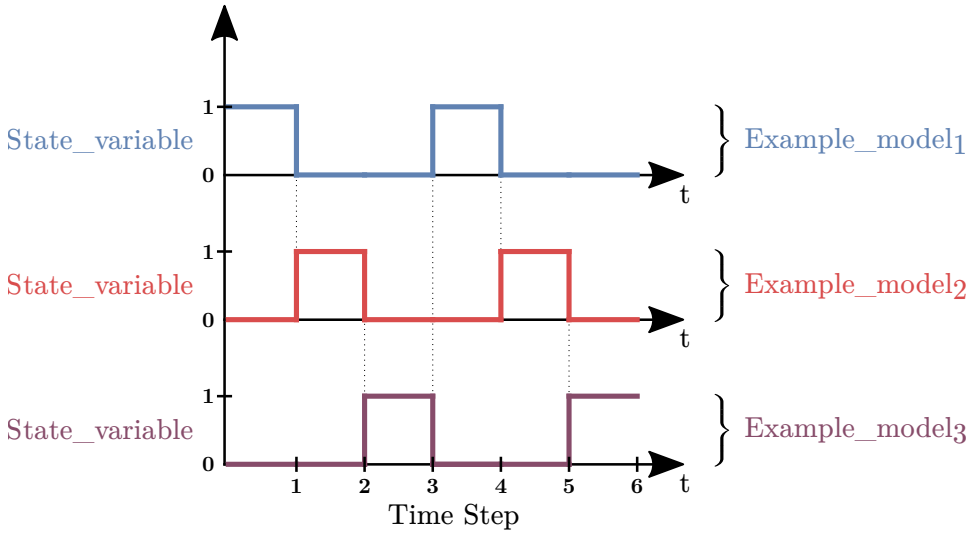


Figure 3.12: Chronogram of the behaviour of the *State_variable* of each example model

Thus, the sequence of a simulation step can be described by the sequence diagram in figure 3.13. In this sequence diagram, we have added an additional colour code with the dashes of each of the atomic models. An atomic model with its dotted line in black corresponds to a $ta = \infty$, an atomic model with its dotted line in blue corresponds to a $ta = 1$, and an atomic model with its dotted line in red corresponds to a $ta = 0$.

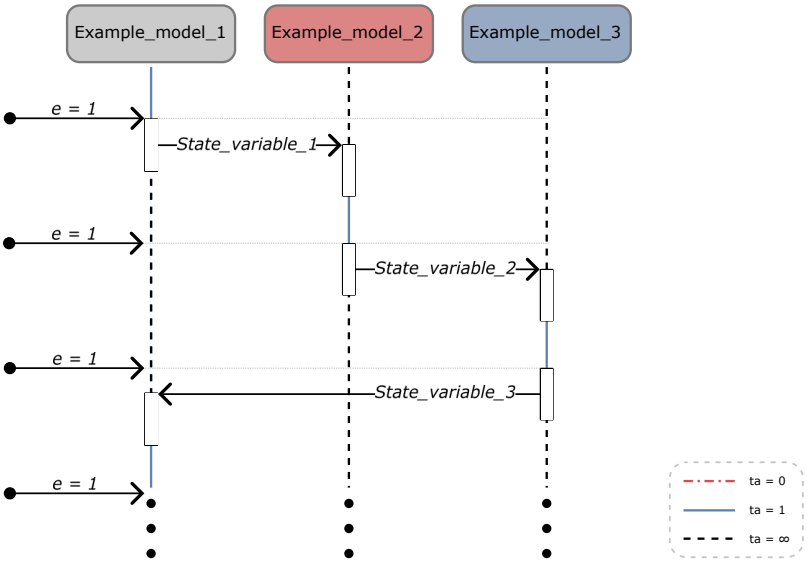


Figure 3.13: Diagram sequential

3.2 Microgrid: From the structure to the I/O functions

The modeling effort proposed in this work will take place following different steps. The objective is to consolidate the links between the four main fields of microgrids and explore the relationships of the model for different levels of specification. Three different models will be presented: a structural, a generative and a predictive model.

The structural model constitutes a first iteration aimed at testing the systemic approach. It yields a model that does not provide results tied to a physical reality of the microgrid i.e. it is not linked to real-life I/O functions and data. The generative model goes into greater detail, using low-level coupling between elements to give more detail to the I/O behavior of the components. The predictive model uses the relationships of previous models to couple different simulators, yielding a powerful co-simulation approach that is linked to I/O functions that were determined from actual data in each simulator.

3.2.1 Structural model: Assembling fields

The proposed structure model is graphically represented throughout the following figures and the association of all field together will be explained step-by-step in this section [251]. At this stage the modeling effort focuses on understanding a microgrid multi-disciplinary structure rather than recreating its data. The circles represent the four base fields of a microgrid, they represent the association of all the internal flows of a field from one internal element to another internal element. The fields are made of different elements who share the same unit in their flow: these elements produce, consume or transform the same unit. The flow is an interaction between one or several elements of the same field, i.e., the energy field has flows in Watts. Flows are not represented graphically in the model below and are considered to be internal to the fields.

The arrows between the fields represent the exchanges from one field to another. We define an exchange that goes from one field to another as being unilateral: it takes from one field to provide to another field.

The first step in the association is to create four components which encapsulate the fields themselves. Its objective is to describe a high level model capable of providing the basis for later going to lower levels of knowledge. These components are shown in figure 3.14. This disposition is used to avoid line crossings when drawing the final model, and does not indicate a more central role of the social field.

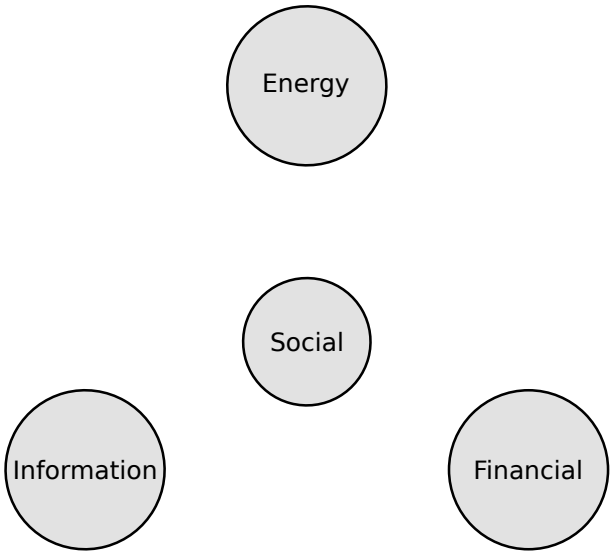


Figure 3.14: Structural model step 1 - Laying out the fields

By using the M&S theory, each component must be described by its input port(s), output port(s), state, a state transition function(s) and an output function(s). By using a systemic ex-

perimental frame, the overall model must be a closed loop, with every field interacting with all the others. The links between fields will be called *exchanges* in this work.

Components Xs and Ys: Field Exchanges

The energy and information fields share a strong link, represented by figure 3.15. Measurements are sent from the Energy to the Information field, while control signals are sent back. These exchanges represent links at all the information layer levels for any sort of energy system architecture, going from real-time control to application level monitoring.

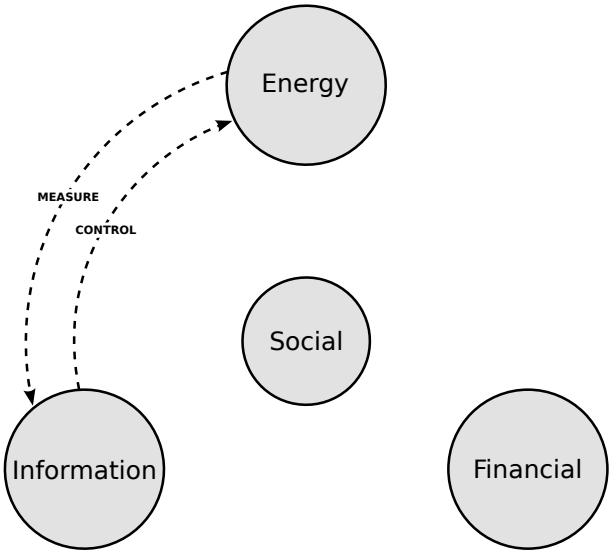


Figure 3.15: Structural model step 2 - Energy and information exchanges

The energy and financial fields exchanges are shown in figure 3.16. Productive activity converts watts into monetary currency, effectively bringing more income to those using the microgrid. Conversely, the microgrid tariffs, maintenance and expansion are examples of economic involvement, which converts currency back into energy production capacity. These exchanges apply to all layers of the energy and financial architecture.

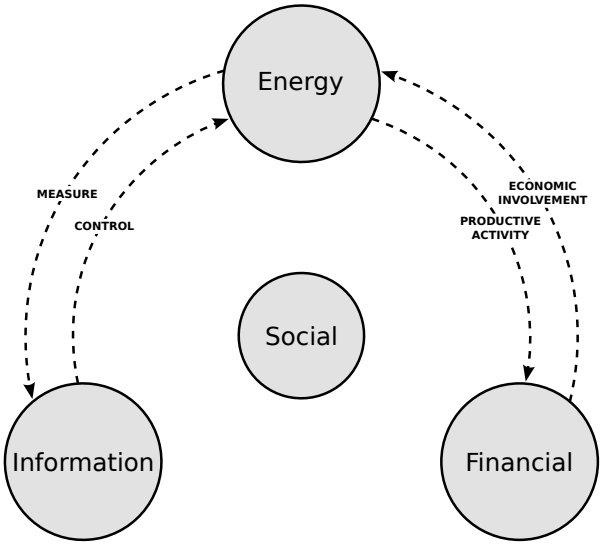


Figure 3.16: Structural model step 3 - Energy and financial exchanges

The figure 3.17 illustrates the exchange between the information and financial fields. Data related to energy consumption and system maintenance is sent from the information field to the

financial field to initiate currency transfers. Conversely, financial resources are used in the information field to address issues such as maintenance, expansion, or human-machine interfaces. These exchanges occur at all levels of the information and financial field architecture.

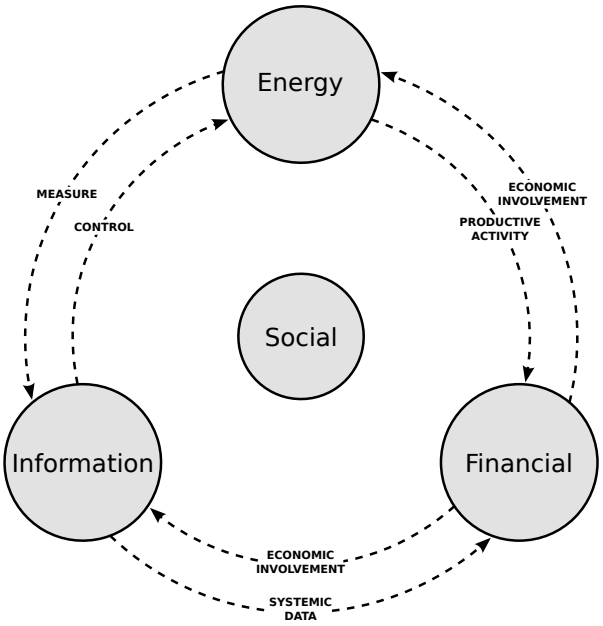


Figure 3.17: Structural model step 4 - Information and financial exchanges

Finally, figure 3.18 summarizes all the exchanges between social and other fields. The social field receives a perceived impact from the energy field, personalized data from the information field and supervision actions from the financial field. These exchanges model *knowledge, perception and rules*, all parts of the issues of community microgrids. The social field sends back a usage to the energy field which summarizes user attitude towards the microgrid, rules to the financial field which represent the type of tariff strategy chosen by the social stakeholders and rules to the information field which represent how the control, data and communication should be handled, typically on how the data is collected for billing and how privacy is preserved.

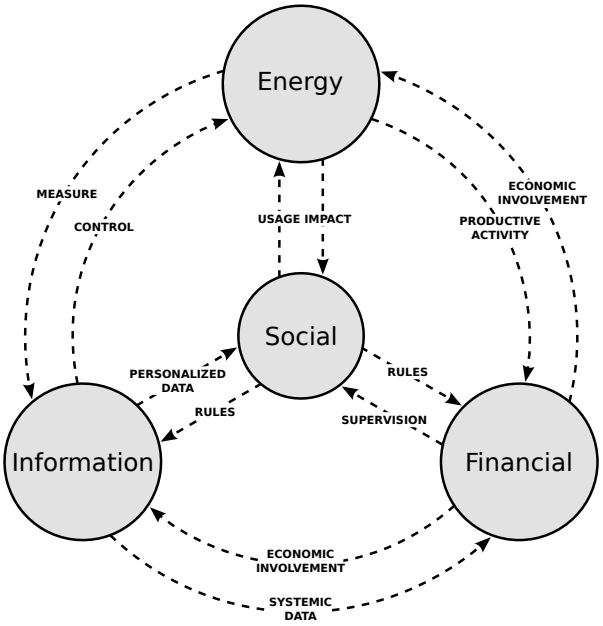


Figure 3.18: Structural model step 5 - Social-to-X exchanges

Table 3.3 summarizes the exchanges and the flows of the model.

Table 3.3: Description of the exchanges.

		Outputs			
		Energy	**Information**	**Financial**	**Social**
Inputs	**Energy**	**Energy Flow**	**Control :** Data sent to the power grid for its control	**Ec. Invol. :** Investment (capital or labor) that is brought for the good functioning of the power grid	**Usage :** Involvement of the users for the good usage of the microgrid
	Information	**Measurements :** Data retrieved from the electrical grid for control or monitoring	**Information Flow**	**Ec. Invol. :** Investment for the improvement or maintenance of the data and control equipment	**Rules :** Rules relating to the data generated within the microgrid
	Financial	**Prod. Act. :** Economic activity linked to the microgrid	**Sys. Data :** Highly aggregated data used by the microgrid managers	**Financial flow**	**Rules :** Rules relating to the ownership of power grid and its tariff system
	Social	**Impact :** Impact of the electricity usage on the population	**Perso. Data :** Usage data provided to the microgrid users	**Supervision :** Follow-up by the microgrid manager to ensure community compliance with the rules	**Social flow**

A comprehensive survey has been developed to assess the overall state of health and identify key weaknesses of a microgrid use case for future sustainability. The survey is designed to be answered by an expert with strong systemic microgrid expertise and covers all relevant fields of the microgrid model through a series of questions pertaining to exchanges and flows. The questionnaire includes a range of questions for each flow and exchange of a microgrid, with each answer being scaled on a rating scale of excellent to bad. An average score is then calculated for each flow and exchange to provide a comprehensive understanding of the microgrid's general state.

This survey aims to gather all important elements for the sustainability of a microgrid that have been identified in the state-of-the-art literature on microgrid systems. Due to the breadth of topics covered in the survey, it can be challenging to find a single individual with expertise in all relevant areas. Therefore, several experts may be consulted to collectively answer and rate the questions related to the microgrid. The survey's strategy is to gather all critical criteria for the operation of microgrids in a simple set of questions. While these questions are general in nature, they can be adapted to target specific actors within the microgrid for a more precise and applied rating. Additionally, the questionnaire seeks to delve deeper into the indicators of each exchange and utilize the results for the rating of flows. The specific questions are presented in Annex 4.2.3 and are drawn from the various issues highlighted in the literature review, including three questions pertaining to the three main issues in the field.

Components States: State of Health

In this structural model, each flow and exchange will be considered to have a single variable, namely its *State of Health*. This choice is made at this stage of modeling because this model is being built from a higher system specification level downwards. Thus, this is not based on I/O measurements whose functions are later described and built into a system. Since we integrate a state of health in each flows and exchanges, we will have to build a DEVS model for each flow but also for each exchange.

The experimental frame from chapter 2 has shown that on a long period of time the exchanges will have a great impact on the health of the microgrid. This assumption is based on the observation

that no matter how well sized and how much was invested onto the microgrid at its installation, poor management or weak community involvement have proved to undermine the whole microgrid enterprise [252].

Thus, the graphical representation of this microgrid diagnosis model in figure 3.19 suits better the reality of this model, with a black box component in charge of representing the exchanges.

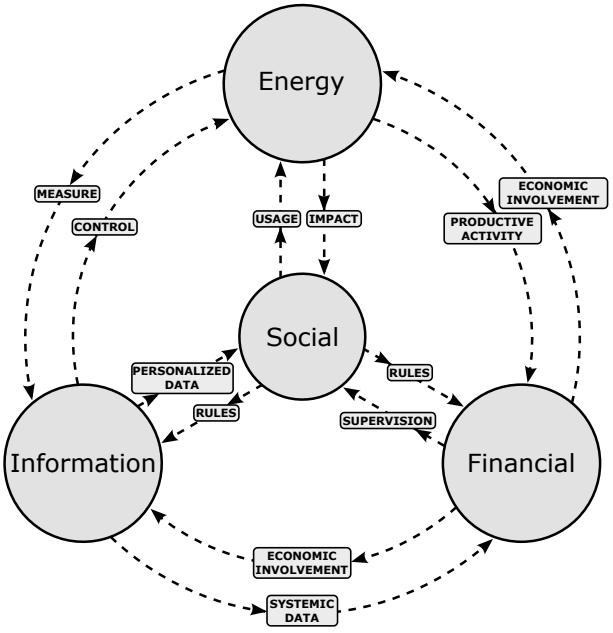


Figure 3.19: Full structural model with all the atomic components represented

The equilibrium of the entire microgrid is governed by the inner flows and the exchanges between the different fields. Dynamically speaking, these exchanges and flows can lead to the failure or success of a microgrid.

To enter into what is called a virtuous circle, all fields must give as much as they receive. We propose to assess the health of each component of the microgrid by rating it on a scale of four health states (from ++ to - -) in order to accurately evaluate the impacts of its flows and exchanges. On a short term scale, the most important characteristic to identify the health level of the fields flows are detailed in Table 3.4.

Table 3.4: Evaluation of the different component stock level

Inner flow	Excellent (++)	Good (+)	Poor (-)	Bad(- -)
Energy	The energy produced capacity is significantly superior to consumption at all times of the year	The energy produced is superior to the consumption with exceptional power cuts	The energy produced capacity is equivalent to the consumption with regular power cuts	The energy produced capacity is inferior to the consumption with repeated power cuts
Information	The microgrid data and command is processed in its entirety and can forecast the evolution of the microgrid over a certain period of time	The microgrid data and command is well processed and can forecast the evolution of the microgrid over a short period of time	The microgrid data and command is poorly processed	The microgrid data and command is slow and unreliable
Financial	The money available is considerably superior to the operating cost of the microgrid	The money available is fairly superior to the operation cost of the microgrid	The money available is equivalent to the operation cost of the microgrid	The money available is inferior to the operation cost of the microgrid
Social	The community benefits greatly from the microgrid with a significantly increased quality of life	The community benefits from the microgrid with a positive impact on quality of life	The community benefits insufficiently from the microgrid with contrasting impacts on its quality of life	The community's quality of life is being reduced by the microgrid

A microgrid with well-functioning fields will have increased resilience for a short period of time, regardless of the quality of its exchanges. However, over the long term, the efficiency of the microgrid's exchanges will be much more impactful, as poor efficiency will gradually deplete the stocks of its fields. In other words, a microgrid with high-quality fields can only withstand contingencies for so long if its exchanges are not efficient. It is therefore important for a microgrid to have both well-functioning fields and efficient exchanges in order to maintain its resilience over the long term. To summarize this, it can be said that the flows of the fields represent the static sustainability of the microgrid and the exchanges represent the dynamic sustainability of the microgrid.

Each exchange therefore has associated indicators in order to be able to judge its good or bad functioning in Table 3.5.

Table 3.5: Evaluation of the different flows and exchanges.

		Outputs			
		Energy	**Information**	**Financial**	**Social**
Inputs	**Energy**	From energy sustainability with all objectives fulfilled (++) to no sustainability objectives validated (- -)	microgrid control is very accurate and stable (++) to microgrid control is poor and unstable (- -)	Funds are re-invested into power hardware with high maintenance (++) to no funds are re-invested into the power hardware with poor maintenance (- -)	Community uses the power grid perfectly well (++) to community uses the power grid poorly (- -)
	Information	Many precise measurements of the energy production are available (++) to no automated and imprecise measurement is available (- -)	From high data value and consummate control (++) to no objective validated for the data and control (- -)	Funds are re-invested into information hardware and software (++) to no funds are re-invested (- -)	The community fully respects the rules established for data collection (++) to community does not respect the rules at all (- -)
	Financial	Energy enables productive uses (++) to energy has no productive uses (- -)	High quality data is available for planning and operation (++) to no data is available (- -)	From financial resilience (++) to no resiliency objectives validated (- -)	Payment collection is highly efficient (++) to payment collection is not reliable (- -)
	Social	All energy needs of the community are satisfied (++) to not even the basic needs are satisfied (- -)	High quality data is available for the users to follow consumption (++) to no data is available (- -)	Funds are re-invested into the community (++) to no funds are re-invested into the community (- -)	From high social acceptance (++) complete rejection by the community (- -)

The *State of Health* variable has four degrees of representation, imaged by the colors based on the equation 3.6 where the Dark Green correspond with the (++) notation and the Red with the (- -) notation. The proposed rating scale will facilitate the integration of the questionnaire results as a score ranging from 0 to 3, with the utilization of a color-coded system serving as a visual representation of the microgrid's overall health status.

$$\begin{aligned} 2.25 < SoH \leq 3 &\rightarrow Dark\,Green \\ 1.5 \leq SoH \leq 2.25 &\rightarrow Light\,Green \\ 0.75 \leq SoH \leq 1.5 &\rightarrow Yellow \\ 0 \leq SoH < 0.75 &\rightarrow Red \end{aligned} \tag{3.6}$$

This microgrid model is composed of four flow health state models shown in figure 3.20 and twelve exchange health state models shown in figure 3.21, which are connected together to form this systemic balance model. The flow component has a singular output, with each of the exchange components receiving an identical state of health value. This singular output is then linked to the three exchange components. The state transition functions will thus act upon this health bar, making it higher or lower to model the possible evolution of the grid.

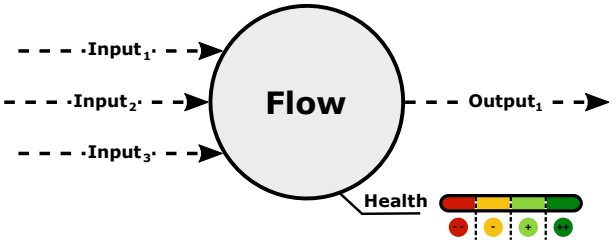


Figure 3.20: View of the fluxes with its inputs, outputs and state

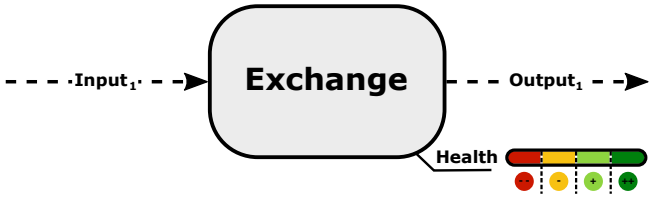


Figure 3.21: View of the exchanges with its inputs, outputs and state

PDEVS model: Setting thresholds and handling events

Flow Model

This system translates to the idea that the health of each field is a reservoir. Exchanges can either deplete or accumulate health in the reservoir. A flow can only deplete the health of its associated reservoir, representing its corrosion. The output function is related to the outgoing Exchanges. The output function, based on elapsed time and previous state, generates the output. When the field health is high, the output will itself also have a high value. Overall, this represents a virtuous microgrid cycle [24]. Conversely, if the health of a field declines, its output will also have a lower value leading to a vicious cycle [24].

A graphical representation of this flow model is represented in figure 3.22 inspired from a state machine graph.

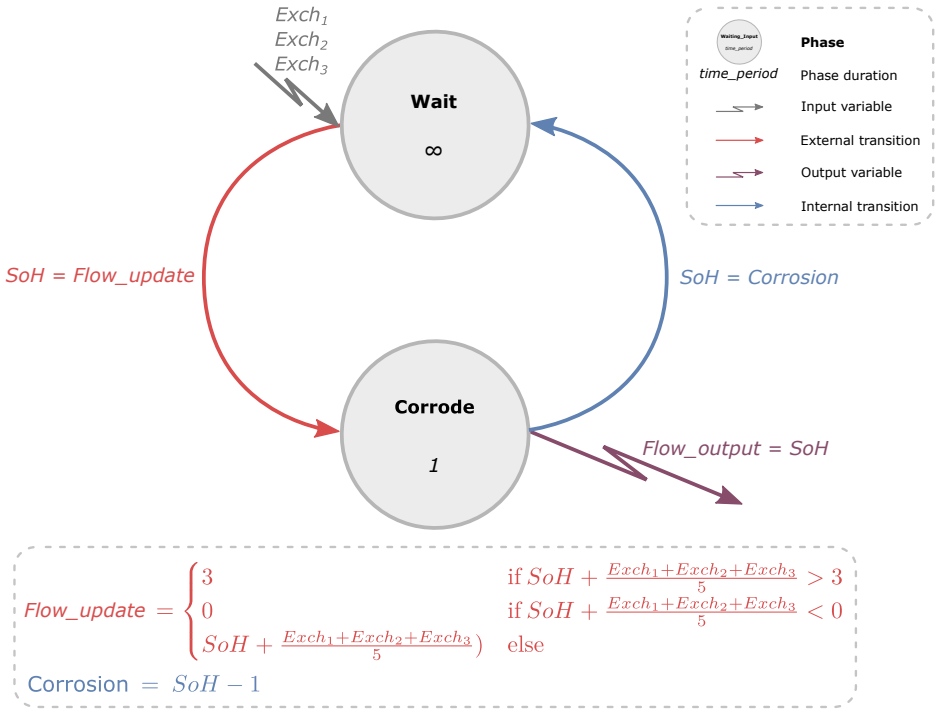


Figure 3.22: Graphical representation of the flux model states and transition functions

The proposed flow state of health model is composed of two variables: $Phase$ and SoH. The first variable, $Phase$, can take on two values: "$Wait$" or "$Corrode$", and is used to model the system as a state machine. The second variable, SoH, is a numerical variable used for calculating state of health changes. The system is initially set to the "$Corrode$" phase. When in this phase, the systems stays for 1 time unit. After this resting time, the $Output_function$ is triggered, which emits the current value of the SoH on port $Send_flow$. After the output function is complete, the $Corrosion$ function is executed, which to recalculate the SoH variable and updates the phase to "$Wait$". From the "$Wait$" state, the system waits for an input variable to be received in $Exch_1$,$Exch_2$ and $Exch_3$ (all received simultaneously), triggering an external transition. This transition is performed by the $Flux_update$ function, which recalculates the SoH variable using the updated exchanges value

from inputs and updates the phase to "*Corrode*". In this model, a division by five of the input exchanges values has been selected in order to achieve a reasonable convergence time of the flow component during the simulation process. This process repeats indefinitely.

The chronogram in figure 3.23 succinctly illustrates the expected behavior of the state of health *SoH* variable as a result of the diverse exchanges value received in input.

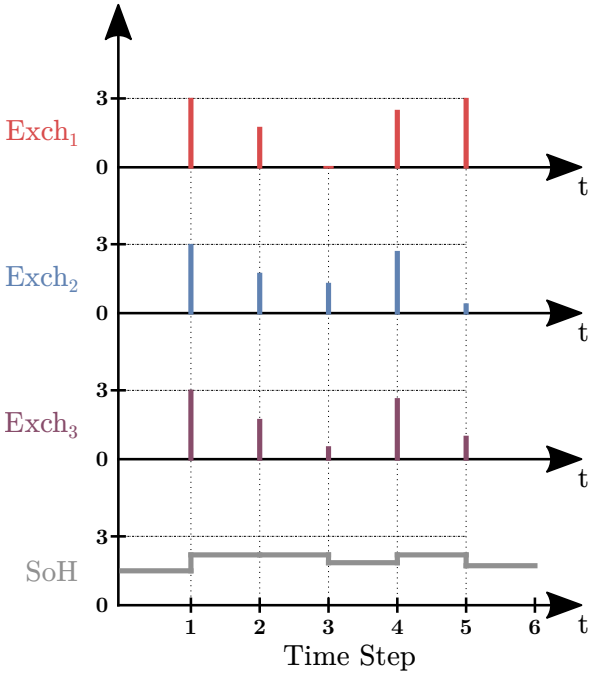


Figure 3.23: Chronogram of the behaviour of *SoH* in the flow model

The mathematical representation of this model using PDEVS formalism is given by equation 3.7.

$$PDEVS_{flux_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.7}$$

Where :

$$X = \{(p, v) \mid p \in \{Exch_1, Exch_2, Exch_3\}, v \in [0; 3]\}$$
$$Y = \{(Send_flow, v) \mid v \in [0; 3]\}$$
$$S = \{(\boldsymbol{Phase}, SoH) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait}\text{”}, \text{“}\boldsymbol{Corrode}\text{”}\}, SoH \in [-1; 3]\}$$
$$\delta_{ext}\Big(\big(\text{“}\boldsymbol{Wait}\text{”}, SoH\big), e, \big((Exch_1, v_1), (Exch_2, v_2), (Exch_3, v_3)\big)\Big) = (\text{“}\boldsymbol{Corrode}\text{”}, Flow_update)$$
$$\text{With } Flow_update = \begin{cases} 3 & \text{if } SoH + \frac{v_1+v_2+v_3}{5} > 3 \\ 0 & \text{if } SoH + \frac{v_1+v_2+v_3}{5} < 0 \\ SoH + \frac{v_1+v_2+v_3}{5} & \text{else} \end{cases}$$
$$\delta_{int}(\text{“}\boldsymbol{Corrode}\text{”}, SoH) = (\text{“}\boldsymbol{Wait}\text{”}, SoH - 1)$$
$$\lambda(\text{“}\boldsymbol{Corrode}\text{”}, SoH) = (Send_flow, SoH)$$
$$ta(\text{“}\boldsymbol{Wait}\text{”}, SoH) = 1$$
$$ta(\text{“}\boldsymbol{Corrode}\text{”}, SoH) = 0$$

The model's starting point, or initialization, is determined based on the responses given in the questionnaire. This process is outlined in equation 3.8. The initial state of an individual flux d, denoted s_{0d}, will have its own unique initial value for the variable *SoH*, referred to as $Initial_health_d$ during the initialization process. The initial phase of all flux is "***Corrode***".

$$s_{0d} = (\text{“}\boldsymbol{Corrode}\text{”}, Initial_Health_d) \tag{3.8}$$

With $Initial_Health_d$ determined by the questionnaire

The inputs ports X of the flux model are named $Exch_1$, $Exch_2$ and $Exch_3$ and can take values from 0 to 3. The output port Y, named $Flow_output$, also has the ability to take values from 0 to 3. The states of the model, denoted as S, consist of a phase and a state variable. The phase, named $Phase$, can have two possible values: “$Wait$” and “$Corrode$”. The state variable, named SoH, can take values from -1 to 3. The external transition function, denoted as δ_{ext}, uses the values available on ports $Exch_1$, $Exch_2$ and $Exch_3$ to calculate the new SoH value using $Flow_update$ and changes the phase from “$Wait$” to “$Corrode$”. This transition is only called when there is input present. The internal transition function, denoted as δ_{int}, also calculates the new SoH with the effect of corrosion and changes the phase from “$Corrode$” to “$Wait$”. The internal structure of the system is designed to prevent any conflicts between δ_{int} and δ_{ext}. However, if there is a situation where the internal transition function δ_{int} and external transition function δ_{ext} overlap, the confluence function δ_{con} will activate the internal transition function δ_{int} to resolve the conflict. The output function, denoted as $\lambda(Phase, SoH)$, sends the value of SoH through the $Flow_output$ port to any connected component. The phases are linked to specific time-advances, with “$Wait$” linked to an infinite value time-advance and “$Corrode$” linked to a time-advance of 1.

Exchange Model

This model describes the relationship between Exchanges and Flows, specifically focusing on the native flow associated with each Exchange. The native flow refers to a specific flow that is linked to a particular Exchange and its health is used as an indicator for the Exchange's State of Health (SoH). The health of the native flow can have a direct impact on the SoH of the Exchange. The equation models the idea that if the native flow is in poor health, it will lead to a decrease in the Exchange's SoH over time, even if the Exchange is well-designed and efficient to begin with. The SoH of the Exchange is continuously updated based on the health of its native flow, and the equation takes into account that a flow in poor condition cannot provide a high-quality Exchange, even if the Exchange is well-designed and efficient. This concept allows to model the fact that a poor-performing flow will eventually affect negatively the Exchange in the long term, even if it's initially well-designed and efficient.

A graphical representation of this exchange model is represented in figure 3.24. We can see here the two phases $Wait_update$ and $Exchange$ which allow to receive the last evaluation of the state of health of the flows, to recalculate the exchanges according to that and to be able to send its new state of health to its flow.

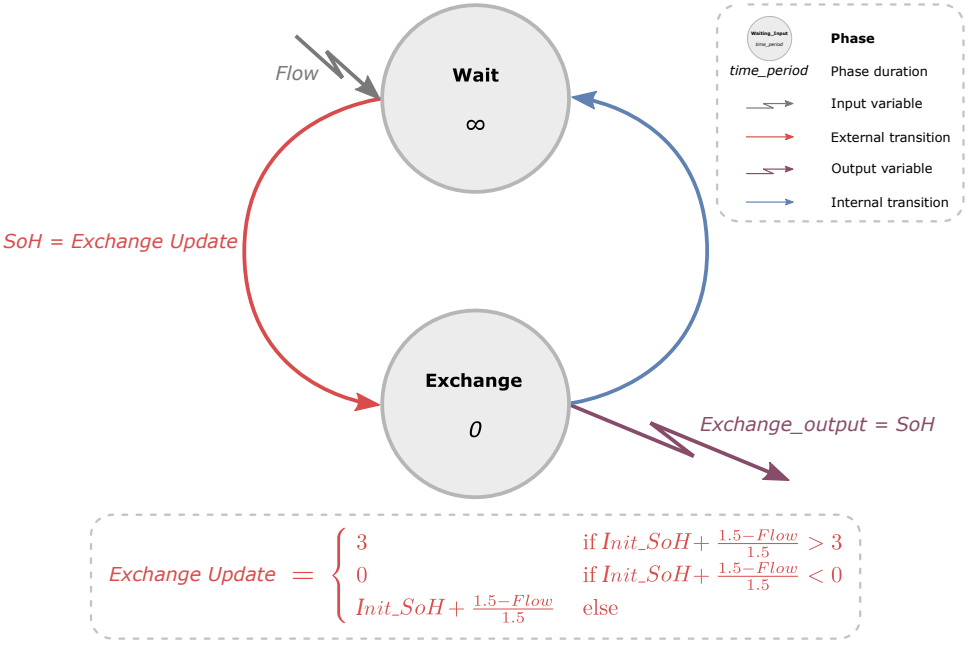


Figure 3.24: Graphical representation of the exchange model states and transition functions

The proposed exchange state of health model is composed of three variables: *Phase*, *SoH* and *Init_SoH*. The first variable, *Phase*, can take on two values: "*Wait_update*" or "*Exchange*", and is used to model the system as a state machine. The second variable, *SoH*, is a numerical variable used for calculating state of health changes. The third variable, *Init_SoH*, is a numerical variable that stores the initial value of *SoH* given by the questionnaire. The system is initially set to the "*Wait_update*" phase, which phase associated time is infinite, i.e. it can only exit this phase by receiving an external event. From the "*Wait_update*" phase, the system waits for an input variable to be received in the *Flow* input, triggering a phase transition. The flow will influence the exchange state of health with a maximum boost from a flow rated 3, a maximum decrease from a flow rated 0 and a neutral influence from a flow rated 1.5. This transition is performed by the *Exchange_update* function, which recalculates the state variable with the updated flow value in input and updates the phase to "*Exchange*". This triggers the output function, which emits the current value of the *SoH* to the *Exchange_output* output. After the output function is complete, the internal transition function is executed, which updates the phase to "*Wait*". This process repeats indefinitely.

The chronogram in figure 3.25 succinctly illustrates the expected behavior of the state of health *SoH* variable as a result of the diverse exchanges value received in input.

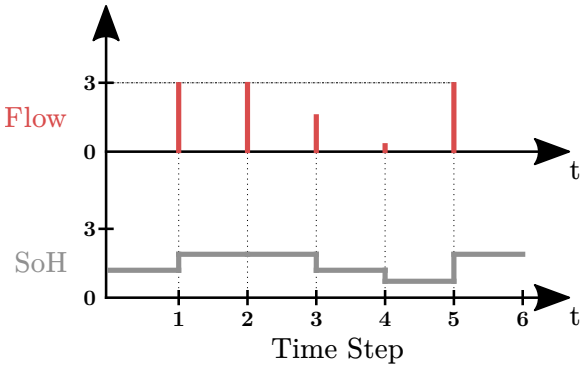


Figure 3.25: Chronogram of the behaviour of *SoH* in the exchange model

The PDEVS mathematical representation of this model is given by equation 3.9.

$$PDEVS_{exchange_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.9}$$

Where :

$X = \{(Flow, v) \mid v \in [0;3]\}$
$Y = \{(Exchange_output, v) \mid v \in [0;3]\}$
$S = \{(\boldsymbol{Phase}, SoH, Init_SoH) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait}\text{”}, \text{“}\boldsymbol{Update}\text{”}\}, SoH \in [0;3], Init_SoH \in [0;3]\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait}\text{”}, SoH, Init_SoH), e, (Flow, v)\big) = (\text{“}\boldsymbol{Update}\text{”}, Exchange_Update, Init_SoH)$

$$\text{With } Exchange_Update = \begin{cases} 3 & \text{if } Init_SoH + \frac{1.5-v}{1.5} > 3 \\ 0 & \text{if } Init_SoH + \frac{1.5-v}{1.5} < 0 \\ Init_SoH + \frac{1.5-v}{1.5} & \text{else} \end{cases}$$

$\delta_{int}(\text{“}\boldsymbol{Update}\text{”}, SoH, Init_SoH) = (\text{“}\boldsymbol{Wait}\text{”}, SoH, Init_SoH)$
$\lambda(\text{“}\boldsymbol{Update}\text{”}, SoH, Init_SoH) = (Exchange_output, SoH)$
$ta(\text{“}\boldsymbol{Wait}\text{”}, SoH, Init_SoH) = \infty$
$ta(\text{“}\boldsymbol{Update}\text{”}, SoH, Init_SoH) = 0$

The model's starting point, or initialization, is determined based on the responses given in the questionnaire. This process is outlined in equation 3.10. The initial state of an individual flux d, denoted s_{0d}, will have its own unique starting value for the variables SoH and $Init_SoH$, referred to as $Initial_health_d$ during the initialization process. The initial phase of all exchanges is “$\boldsymbol{Wait}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait}\text{”}, Initial_Health_d, Initial_Health_d) \quad (3.10)$$
With $Initial_Health_d$ determined by the questionnaire

The input port X of the exchange model is named $Flow$ and can take values from 0 to 3. The output port Y, named $Exchange_output$, also has the ability to take values from 0 to 3. The states of the model, denoted as S, consist of a phase, a state variable and the initial value of its state variable. The phase, named $Phase$, can have two possible values: “$Wait$” and “$Exchange$”. The state variable, named SoH, can take values from 0 to 3. The initialisation state variable, named $Init_SoH$, can take values from 0 to 3 is used to store the value of the initial state of health of the exchange. The external transition function, denoted as δ_{ext}, uses the values available on port $Flow$ to calculate the new SoH value using $Exchange_update$ and changes the phase from “$Wait$” to “$Corrode$”. This transition is only called when there is input present. The internal transition function, denoted as δ_{int}, changes the phase from “$Corrode$” to “$Wait$”. The internal structure of the system is designed to prevent any conflicts between δ_{int} and δ_{ext}. However, if there is a situation where the internal transition function δ_{int} and external transition function δ_{ext} overlap, the confluence function δ_{con} will activate the internal transition function δ_{int} to resolve the conflict. The output function, denoted as $\lambda(Phase, SoH, Init_SoH)$, sends the value of SoH through the $Exchange_output$ port to any connected component. The phases are linked to specific time-advances, with “$Wait$” linked to an infinite value time-advance and “$Exchange$” linked to a time-advance of 0.

Final coupled model

The root model, or final coupled model, of our systemic-based modeling of a microgrid in PDEVS allows us to connect all of the previously presented atomic models of flux and exchange. This coupled model represents the completion of our structural diagnosis modeling, which integrates the microgrid's fields of energy, information, finance, and social for a more comprehensive understanding of its operation. The root model serves as the central hub for the microgrid model, allowing for the simulation of interactions between the atomic models. This can provide insight into the microgrid's operation and the interconnections between its various fields.

PDEVS, the modeling framework used for the root model, offers powerful simulation capabilities for accurately and efficiently modeling the complex interactions within the microgrid. This allows for gaining insights into the microgrid's behavior that would not be possible with other modeling

techniques. The coupled model presented in this work completes our systemic-based modeling of a microgrid. By connecting all atomic flux and exchange models, we can comprehensively understand the microgrid's operation and the relationships between its various fields. This information can be used to inform decision-making regarding the microgrid's design and operation.

$$PDEVS_{root_model} = (X, Y, D, \{M_d\}, EIC, EOC, IC) \tag{3.11}$$

Where :

$X = \{\}$
$Y = \{\}$
$D = D_{flux} \cup D_{exchange}$
With $D_{flux} = \{Energy, Financial, Information, Social\}$
and $D_{exchange} = \{E \to F, E \to I, E \to S, F \to E, F \to I, F \to S, I \to E, I \to F, I \to S, S \to E, S \to F, S \to I\}$
$\{M_d\} = \{PDEVS_{flux_model} \mid d \in D_{flux}\} \cup \{PDEVS_{exchange_model} \mid d \in D_{exchange}\}$
$EIC = \{\}$
$EOC = \{\}$
$IC = \{ \big((Energy,"Send_flow"),(E\to F,"Flow")\big)\ ,\ \big((Energy,"Send_flow"),(E\to I,"Flow")\big),$
$\big((Energy,"Send_flow"),(E\to S,"Flow")\big)\ ,\ \big((Financial,"Send_flow"),(F\to E,"Flow")\big),$
$\big((Financial,"Send_flow"),(F\to I,"Flow")\big)\ ,\ \big((Financial,"Send_flow"),(F\to S,"Flow")\big),$
$\big((Information,"Send_flow"),(I\to E,"Flow")\big)\ ,\ \big((Information,"Send_flow"),(I\to F,"Flow")\big),$
$\big((Information,"Send_flow"),(I\to S,"Flow")\big)\ ,\ \big((Social,"Send_flow"),(S\to E,"Flow")\big),$
$\big((Social,"Send_flow"),(S\to F,"Flow")\big)\ ,\ \big((Social,"Send_flow"),(S\to I,"Flow")\big),$
$\big((E\to F,"Send_exchange"),(Financial,"Exch_1")\big)\ ,\ \big((E\to I,"Send_exchange"),(Information,"Exch_1")\big),$
$\big((E\to S,"Send_exchange"),(Social,"Exch_1")\big)\ ,\ \big((F\to E,"Send_exchange"),(Energy,"Exch_1")\big),$
$\big((F\to I,"Send_exchange"),(Information,"Exch_2")\big)\ ,\ \big((F\to S,"Send_exchange"),(Social,"Exch_2")\big),$
$\big((I\to E,"Send_exchange"),(Energy,"Exch_2")\big)\ ,\ \big((I\to F,"Send_exchange"),(Financial,"Exch_2")\big),$
$\big((I\to S,"Send_exchange"),(Social,"Exch_3")\big)\ ,\ \big((S\to E,"Send_exchange"),(Energy,"Exch_3")\big),$
$\big((S\to F,"Send_exchange"),(Finanaicial,"Exch_3")\big)\ ,\ \big((S\to I,"Send_exchange"),(Information,"Exch_3")\big)\}$

A vision of how the coupling of this proposed structural model results in messages exchanges is given in the form of a sequence diagram in figure 3.26 representing the execution sequence of all flows and exchanges during an execution step.

This sequence diagram represents the four field flows models and the twelve exchange models between them. Each block will represent the execution of an internal or external transition function that can also update the phase of its model, linked to a specific time advance function. The color of the dotted line of each model represents a specific ta, a red dotted line represents a $ta = 0$, a blue doted line represents a $ta = 1$ and a black dotted line represents a $ta = \infty$. The arrows in the sequential model represent an output function that sends a message to another model input, initiating an external function on the latter. In this coupled model, all the flows will be executed at the same time and all the exchanges will be executed at the same time because they become imminent simultaneously by the structure of this coupled model. This representation can be seen as an equivalent of the equation above, but it brings another perspective for understanding this overall model operation and behavior.

Model Summary and Overview

The proposed microgrid systemic modeling approach utilizes a multi-disciplinary approach to examine the sustainability of microgrids. By connecting the four key components of a microgrid (energy, information, financial, and social) in a homomorphic model, this approach allows for a thorough understanding of the complex interactions and dynamics between these components.

This model offers a holistic view of the microgrid, taking into consideration the nature of interconnections between its various components and their impact on one another. This can provide valuable insights into potential causes of failure or weaknesses in a microgrid, as well as the potential effects of interventions or changes to the system.

One of the major challenges facing microgrid research is the availability of reliable data sources. To overcome this challenge, the proposed model utilizes the concept of flows and exchanges to identify the key drivers and factors that influence the performance and sustainability of a microgrid. By quantifying and analyzing these flows and exchanges, researchers can gain a deeper understanding of the underlying mechanisms and dynamics of the microgrid, and can use this information to develop more effective strategies for enhancing its sustainability, even in the absence of complete data.

To validate the proposed model, simulations are conducted using the appropriate simulation framework, and the results are compared with existing literature. The simulation structure and results can be found in section section 4.2.1 which starts in page 127. Additionally, this model can also be used as a diagnostic tool to identify potential areas for improvement in a microgrid by comparing simulation outputs to real-life data and observations.

In conclusion, the proposed microgrid systemic modeling approach provides a valuable tool for research on the sustainability of microgrids. It offers a systemic and dynamic view of the microgrid, which can aid researchers in better understanding the complex interactions and dynamics that influence the performance and sustainability of microgrids. The use of flows and exchanges as a data source further enhances the applicability of the model even in the absence of complete data, making it a reliable solution for microgrid research. This approach can ultimately support the development of more effective strategies and interventions for enhancing the sustainability of microgrids in the future.

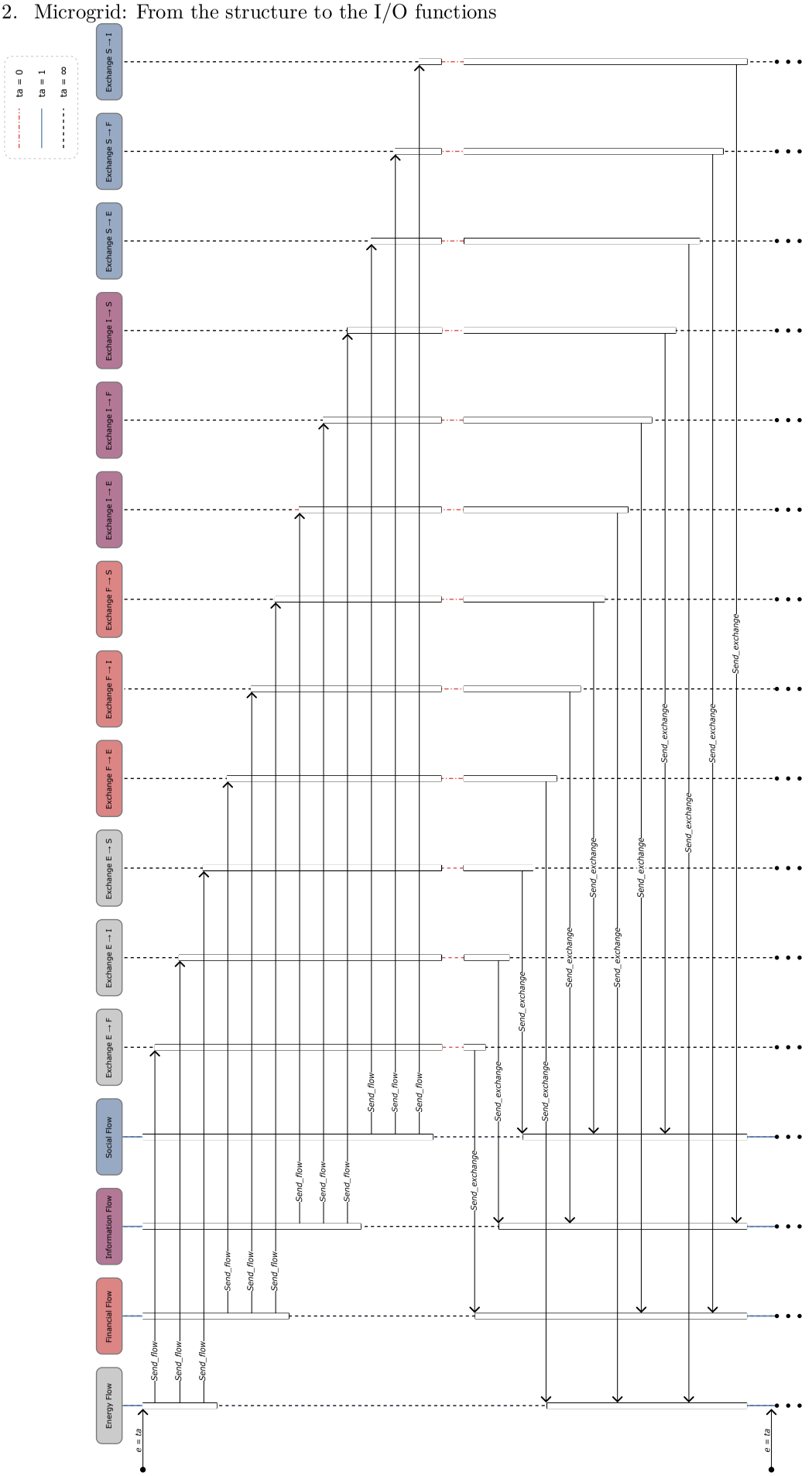


Figure 3.26: The sequential diagram of the full generative model

3.2.2 Generative Model: Assembling components

With the structural model in place, it is now possible to take a step further down the system specification ladder towards an I/O function description capable of generating data at the level of the internal components for each field. In practical terms, this means taking a further step into linking elements and architectures to the internal transition, external transition and output functions.

Inner-Field Components: Morphism at a component level

As stated in Chapter 2, the challenge of modeling sustainability in microgrids is creating a field-to-field morphism capable of allowing coupling between fields. While the structural model has shown that this coupling is possible, it provides no capability of generating useful data from the model, which is what this generative model seeks to achieve.

The morphism proposed in this model uses analogy to the electrical energy grid as its foundation for modeling all the fields. This means modeling the elements of each component as sources, loads, storage or decision entities. These components are detailed in table 3.6. This table shows that not all fields have all the four components. The information field does not have a load component as it was considered for this model that data is created, but not destroyed. The social field does not have a source, a storage or a load for acceptance, at least not for this model. Instead this field has a purely decision-based entity which is an individual. The energy and financial fields both have the four components.

Table 3.6: Details of components of the generative model

Field	Source	Load	Storage	Decision
Energy	Generators (G)	Loads (L)	Batteries (B)	Converters (C)
Information	Measure (M)	-	Data storage (S)	Electronic Device (D)
Financial	Revenue (R)	Expenditures (E)	Wallet (W)	Financial Institution (F)
Social	-	-	Individual Acceptance (A)	-

Like the electric components, a source component will be able to generate flow in its own field, a load component will be able to consume flow in its own field, a storage component will be able to store and redistribute flow in its own field and the decision component will be able to block or command the flow in its own field. The graphical representations of all these components for the component-based model are shown in figure 3.30. A color is established for each field and a form for each component type.

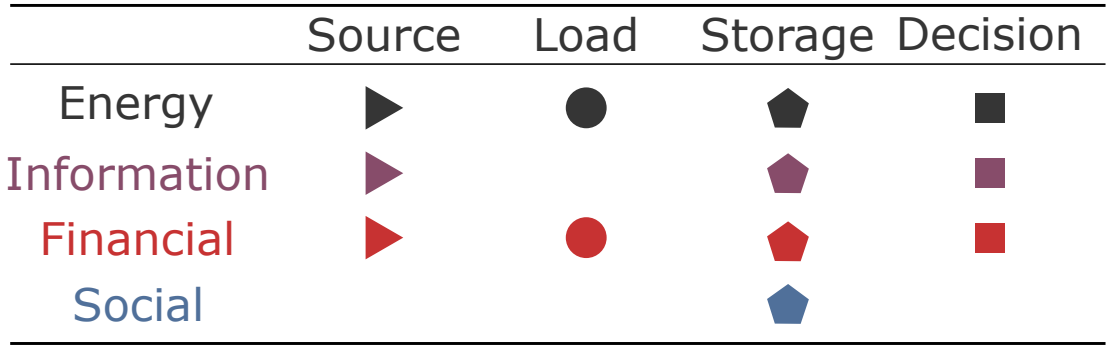


Figure 3.27: Components of the generative model

In this component model, flows represent the interaction between two components of the same field and an exchange represents an interaction of two components that are from two different fields. The figure 3.28 shows the graphical representation of the model. These interactions can be bidirectional or unidirectional. To distinguish all the elements, each element will be named with 3 characters, A letter for its type, a second letter for its field and a number to ensure name uniqueness.



(a) Flow representation (b) Exchange representation

Figure 3.28: Component model representation example

Component I/O: Flows

In the previous model, components were flows and each had one input and one output to/from the other flows by the intermediary of exchange components. In this model, the inputs and outputs of each component are linked uniquely to their internal flows. Exchanges will be represented through a coupling mechanism that will be explained in the next section.

Each component is able to communicate directly with the neighboring components through flows to form the network of the domain in question. The components of the microgrid will thus form 4 different networks as represented in the figure 3.29 with an energy grid, an information network, a financial network and a network of individuals.

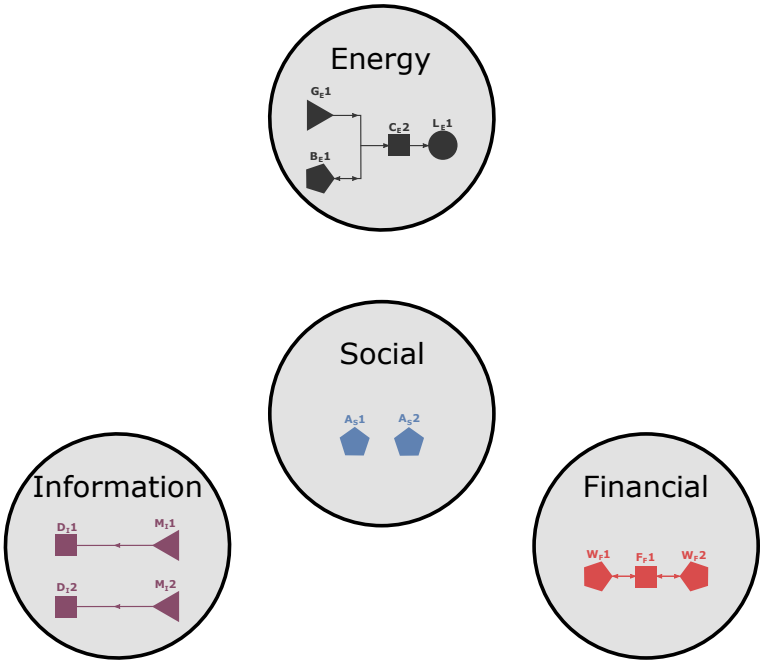


Figure 3.29: Components of the generative model with their I/O connected together with only the flows

Figure 3.29 shows all field specific components types with their inputs and outputs connected together. In this image, the energy component depicting a solar home system is represented by a source, a converter, a small battery and a load, typically charging a cell phone. The information component depicts the energy counter at the costumer end. The financial component shows two

wallets connected together by a system of payment together with the revenues and expenditures of the user and the energy provider. The social component shows two individuals whose acceptances are not directly connected together. Here the exchanges are obviously missing.

Component States: Influencers and Influencees

The external links missing from the previous models are represented here by component coupling, i.e. the influencer/influencee link. These links depict how changes in state of the influencer are mapped to the state transition functions of the influencees. The representation full link between all the components of the generative model is shown in figure 3.30.

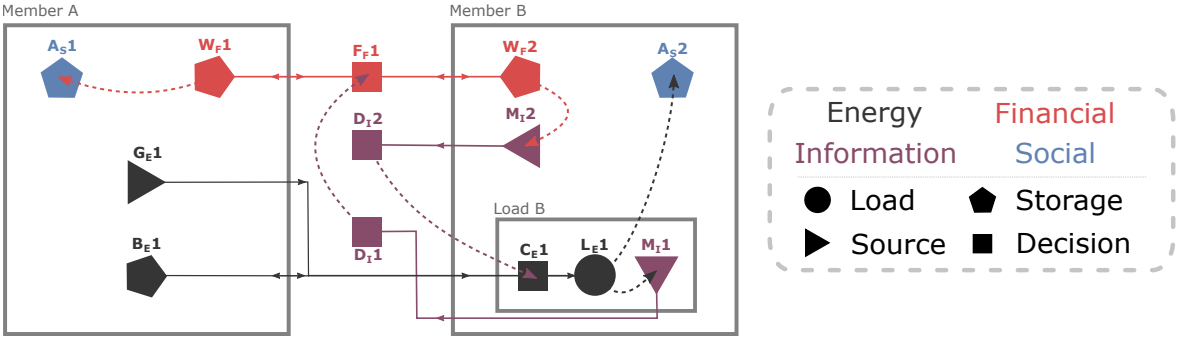


Figure 3.30: Components of the generative model with their I/O connected together adding the exchanges

In black is represented the electrical grid of this microgrid, so there is a source G_E1 and a battery B_E1 that provides energy to a load L_E1. In purple, a measurement M_I1 transmits the information to a decision entity D_I1 that triggers a payment proportional to the consumption of $Member_B$. In red are represented the financial links between these two members, where $Member_B$ consumes energy and pays $Member_A$ an amount based on the energy consumed. Each user has financial storage (a wallet W_F1 or W_F2) with a financial decision entity that can be linked to a bank F_F1. In blue is shown the acceptance of the two users with the acceptance A_S1 of $Member_A$ which will be influenced by the good payment of its customer and the acceptance A_S2 of $Member_B$ which will be influenced by the availability of energy or not.

The plain links allow us to identify the internal flows of each of the components, while the exchanges between the components are represented here by dotted arrows. To understand the full operation of the microgrid and this model, we can begin with the energy source G_E1 and the battery B_E1. These ones provide energy to the charge L_E1 of the $Member_B$ and will influence the acceptance A_S2 by a first interaction. The second interaction of L_E1 with the information layer will measure the amount of energy consumed. This measurement M_I1 of energy will then be transferred to an information decision entity D_I1, which will calculate the sum the consumer owes about what he has consumed. This information decision entity D_I1 will then interact with the financial decision entity F_F1, which will trigger a payment from wallet W_F2 to wallet W_F1. Finally, this payment will directly influence the member's acceptance A_S1 in the sense that he has received his payment. A measurement M_I2 of the wallet W_F1 is also transferred to the electronic device D_I2 that check is the $Member_B$ has enough funds to pays its future consumption. Then this device D_I2 can decides to cut the power converter C_E1 and disconnect the load L_E1 if the founds are insufficient.

Now that the couplings are established for this first model, it is possible to define its transition functions more precisely and their links with the states of influencers.

PDEVS Model: Coupling internal flows and external exchanges

This section will go deeper in the formalisation of the components states and their states transition function. This use case modeled contains 13 component that interact with each other through the different fields. Each component of the presented simple microgrid use case will be detailed and

formalized with its structural tuple based on DEVS and a graphical representation of its phases and transitions functions.

Generator Model

In this model, the generator is the component that acts as a clock of the simulation. The generator model triggers a new generation after each elapsed time of 1 following a predetermined production evolution curve. The formal representation of the generator is provided in equation 3.12 and imaged with the figure 3.31 and a chronogram of its expected state behavior in figure 3.32.

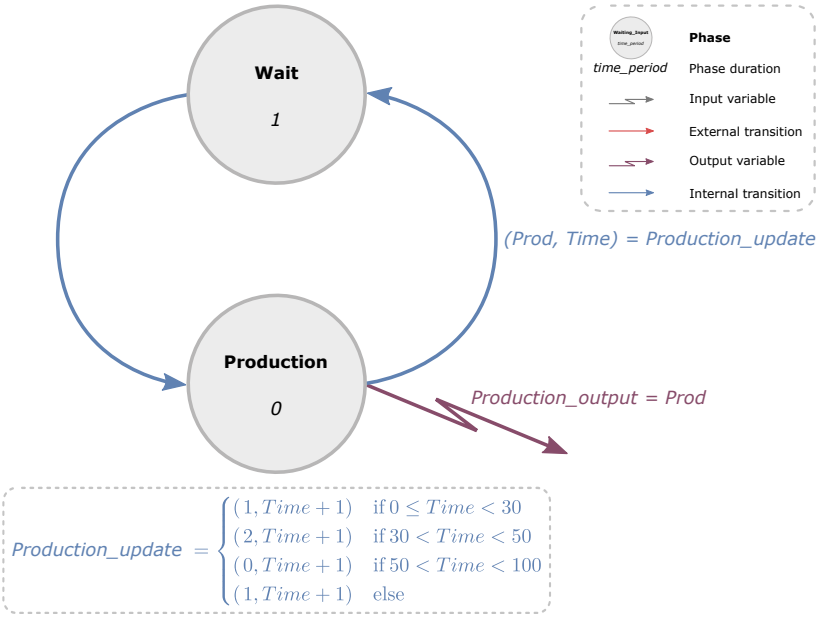


Figure 3.31: Graphical representation of the power source S1 model states and dynamics

The proposed generator model is composed of three variables: $Phase$, $Prod$ and $Time$. The first variable, $Phase$, can take on two values: "$Wait$" or "$Production$", and is used to model the system as a state machine. The second variable, $Prod$, is a numerical variable used for calculating the new energy production. The third variable, $Time$, is a numerical variable that stores the current time of the simulation in order to command the generator according to a defined production curve. The system is initially set to the "$Wait$" phase, which phase associated time is 1, i.e. it will exit this phase when the elapsed time will become 1. From the "$Wait$" state, the system wait for the elapsed time to reach 1 triggering the phase change to "$Production$", making the system imminent. This triggers the output function, which emits the current value of the $Prod$ in $Production_output$ output. After the output function is complete, the internal transition is executed using the function $Production_Update$, which updates the phase to "$Wait$" and change the value of $Prod$ in function of the current $Time$ value; it also increment the value of $Time$. This process repeats indefinitely every time step of 1.

The chronogram in figure 3.32 succinctly illustrates the expected behavior of the $Prod$ variable.

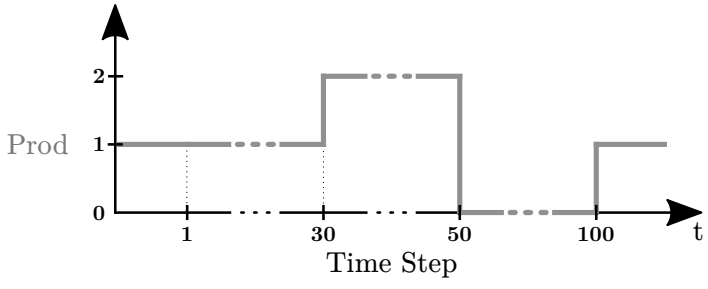


Figure 3.32: Chronogram of the behaviour of $Prod$ in the generator model

The PDEVS mathematical representation of this model is given by equation 3.12.

$$PDEVS_{generator_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.12}$$

Where:

$X = \{\}$
$Y = \{(Production_output, v) \mid v \in [0; 2]\}$
$S = \{(\boldsymbol{Phase}, Prod, Time) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait}\text{”}, \text{“}\boldsymbol{Production}\text{”}\}, Prod \in [0; 2], Time \in \mathbb{N}\}$
$\delta_{ext}\big((\boldsymbol{Phase}, Prod, Time), e, X\big) = \varnothing$
$\delta_{int}(\text{“}\boldsymbol{Wait}\text{”}, Prod, Time) = (\text{“}\boldsymbol{Production}\text{”}, Prod, Time)$
$\delta_{int}(\text{“}\boldsymbol{Production}\text{”}, Prod, Time) = (\text{“}\boldsymbol{Wait}\text{”}, Production_update)$

$$\text{With } Production_update = \begin{cases} (1, Time + 1) & \text{if } 0 \leq Time < 30 \\ (2, Time + 1) & \text{if } 30 < Time < 50 \\ (0, Time + 1) & \text{if } 50 < Time < 100 \\ (1, Time + 1) & \text{else} \end{cases}$$

$\lambda(\text{“}\boldsymbol{Production}\text{”}, Prod, Time) = (Production_output, Prod)$
$ta(\text{“}\boldsymbol{Wait}\text{”}, Prod, Time) = 1$
$ta(\text{“}\boldsymbol{Production}\text{”}, Prod, Time) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.13. The initial state of an individual generator d, denoted s_{0d}, will set the generator's production to 1 at time 0. The initial phase is set to “***Corrode***”.

$$s_{0d} = (\text{“}\boldsymbol{Wait}\text{”}, 1, 0) \tag{3.13}$$

The source has no input port since its production potential is already known in the model. The output port, named *Production_output*, has the ability to take values from 0 to 2. The state of the model consist of 3 variables for phase, production and time. The phase, named *Phase*, can have two possible values: “*Wait*” and “*Production*”. The production variable, named *Prod*, can take values from 0 to 2. The time variable, named *Time*, can take values in $\mathbb{N}$. The system has an internal transition function, denoted as δ_{int}. Applying δ_{int} when in the phase “*Wait*” changes the phase variable to “*Production*”, leaving untouched the other state variables. Applying δ_{int} when in in the phase “*Production*”, it computes the new production depending on the *Time*, increment *Time* and also changes the phase variables to “*Wait*”. The internal structure of the system is designed to prevent any case of confluence, i.e. a conflicts between δ_{int} and δ_{ext}. The output function, denoted as $\lambda(\text{“}Production\text{”}, Prod, Time)$, sends the value of *Prod* through the *Production_output* port to the connected component. The phases are linked to specific time-advances, with “*Wait*” linked to an time-advance of 1 and “*Production*” linked to a time-advance of 0.

Load Model

The Load model is activated and its power consumption is set by its associated power converter. The formal representation of the load model is provided in equation 3.14 and imaged with the figure 3.33 and a chronogram of its expected state behavior in figure 3.34.

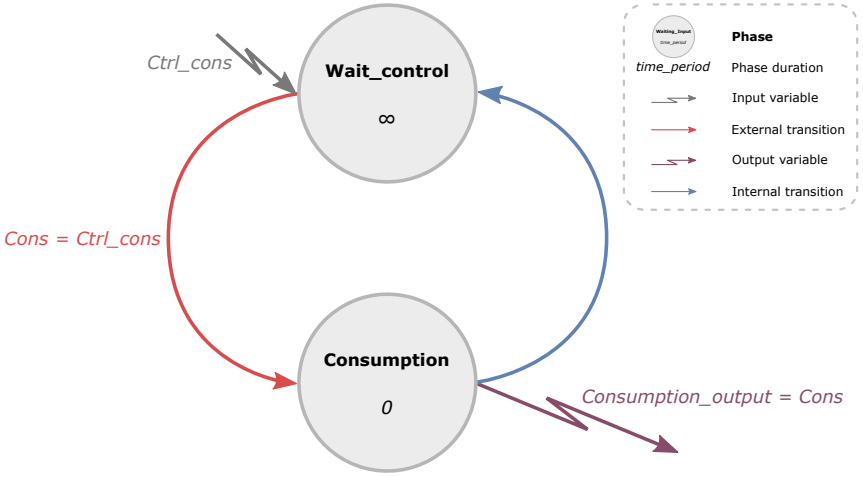


Figure 3.33: Graphical representation of the load L1 model states and dynamics

The proposed load model is composed of two variables: *Phase* and *Cons*. The first variable, *Phase*, can take on two values: "*Wait_control*" or "*Consumption*", and is used to model the system as a state machine. The second variable, *Cons*, is a numerical variable used for calculating the consumption of the load. The system is initially set to the "*Wait_control*" phase, which phase associated time is infinite and the *Cons* is set to 1. From the "*Wait_update*" state, the system waits for an input variable to be received in the input *Ctrl_cons*, triggering an external transition. The control from the power converter will activate or not the consumption of the load depending on the state of the power converter. This transition update the *Cons* value with the control value received in input and updates the phase to "*Consumption*". This triggers the output function, which outputs the value of the *Cons* in *Consumption_output* output. After the output function is complete, the internal transition is executed to update the phase to "*Wait_control*". This process repeats indefinitely.

The chronogram in figure 3.34 succinctly illustrates the expected behavior of the *Cons* variable as a result of the control received in input.

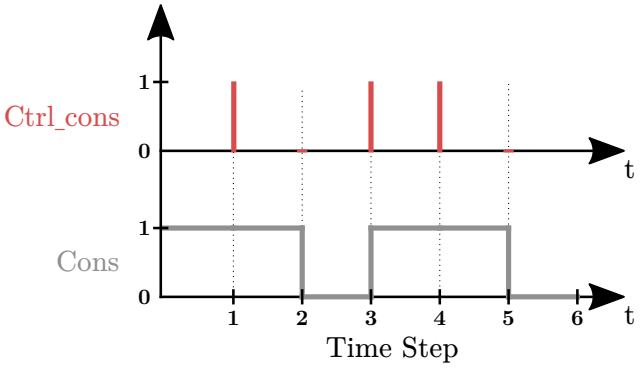


Figure 3.34: Chronogram of the behaviour of *Cons* in the load model

The PDEVS mathematical representation of this model is given by equation 3.14.

$$PDEVS_{load_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.14}$$

Where :

$$X = \{(Ctrl_cons, v) \mid v \in [0; 1]\}$$
$$Y = \{(Consumption_output, v) \mid v \in [0; 1]\}$$
$$S = \{(\boldsymbol{Phase}, Cons) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_control}\text{”}, \text{“}\boldsymbol{Consumption}\text{”}\}, Cons \in [0; 1]\}$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_control}\text{”}, Cons), e, (Ctrl_cons, v)\big) = (\text{“}\boldsymbol{Consumption}\text{”}, v)$$

$$\delta_{int}(\text{“}\boldsymbol{Consumption}\text{”}, Cons) = (\text{“}\boldsymbol{Wait_control}\text{”}, Cons)$$
$$\lambda(\boldsymbol{Phase}, Cons) = (Consumption_output, Cons)$$
$$ta(\text{“}\boldsymbol{Wait_control}\text{”}, Cons) = \infty$$
$$ta(\text{“}\boldsymbol{Consumption}\text{”}, Cons) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.15. The initial state of an individual load d, denoted s_{0d}, will set the load's consumption to 1. The initial phase is set to “$\boldsymbol{Wait_control}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait_control}\text{”}, 1) \tag{3.15}$$

The input port X of the example model is named $Ctrl_cons$ can take values in $[\![0;1]\!]$. The output port Y, named $Consumption_output$, also has the ability to take values in $[\![0;1]\!]$. The states of the model, denoted as S, consist of a phase and a consumption variable. The phase, named $Phase$, can have two possible values: “$Wait_control$” and “$Consumption$”. The consumption variable, named $Comp$, can take values in $[\![0;1]\!]$. The external transition function, denoted as δ_{ext}, uses the value available on port $Ctrl_cons$ to compute the new $Cons$ value by replacing it by the value received in input and changes the phase from “$Wait_control$” to “$Consumption$”. This transition is only called when there is input present. The internal transition function, denoted as δ_{int}, changes the phase from “$Consumption$” to “$Wait_control$”. The internal structure of the system is designed to prevent any conflicts between δ_{int} and δ_{ext}. However, if there is a situation where the internal transition function δ_{int} and external transition function δ_{ext} overlap, the confluence function δ_{con} will activate the internal transition function δ_{int} to resolve the conflict. The output function, denoted as $\lambda(Phase, Cons)$, sends the value of $Cons$ through the $Consumption_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_control$” linked to an infinite time-advance and “$Consumption$” linked to a time-advance of 0.

Battery Model

The battery model is activated by its input power and its output power set by its associated converter. The formal representation of the battery model is provided by equation 3.16 and imaged with the figure 3.35 and a chronogram of its expected state behavior in figure 3.36.

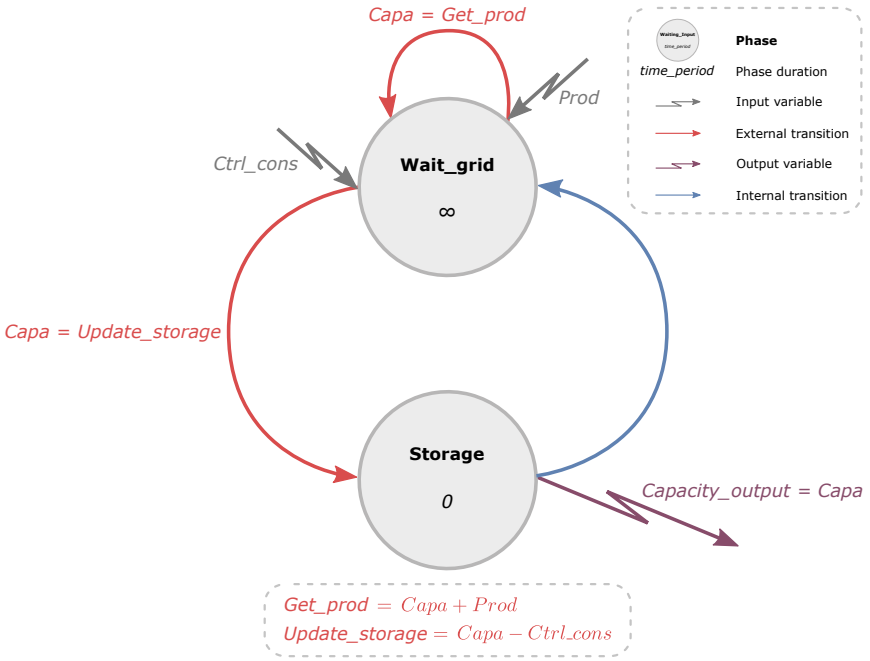


Figure 3.35: Graphical representation of the battery B1 model states and dynamics

The proposed battery model is composed of two variables: *Phase* and *Capa*. The first variable, *Phase*, can take on two values: "*Wait_grid*" or "*Storage*", and is used to model the system as a state machine. The second variable, *Capa*, is a numerical variable used for calculating the total capacity of energy stored in the battery. The system is initially set to the "*Wait_grid*" phase, which phase associated time is infinite. From the "*Wait_update*" state, the system waits for an input variable to be received in the input *Prod*, triggering an external transition. The value of produced energy will be added with the *Capa* value in order to store it. From this point, the system waits for an input variable to be received in the input *Ctrl_cons*, triggering an other external transition performed by the *Get_prod* function. The value of consumed energy will be subtracted with the *Capa* value in order to update the full grid balance and the phase will stay at "*Wait_update*". This transition is performed by the *Update_storage* function to recompute the capacity with the consumption value in input and change the phase to "*Storage*". This triggers the output function, which outputs the value of the *Capa* in *Capacity_output* output. After the output function is complete, the the internal transition is executed to update the phase to "*Wait_grid*". This process repeats indefinitely.

The chronogram in figure 3.36 succinctly illustrates the expected behavior of the *Capa* variable as a result of the production and load control received in input.

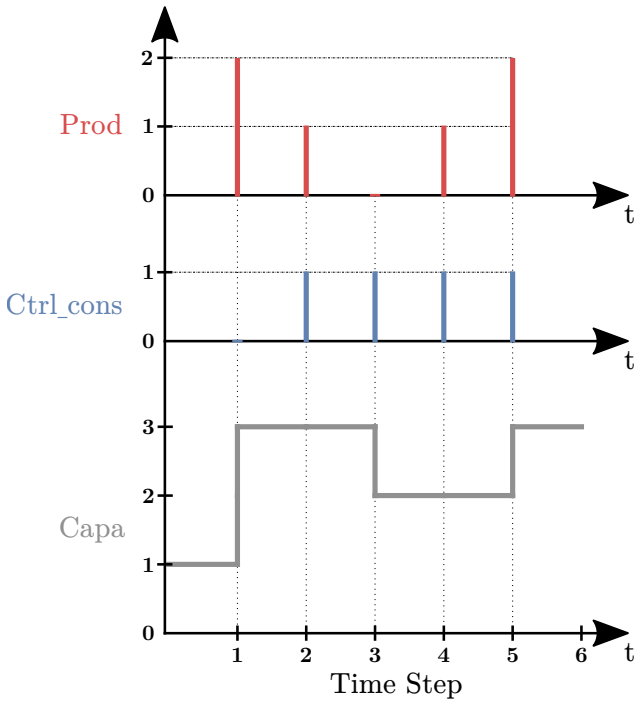


Figure 3.36: Chronogram of the behaviour of *Capa* in the battery model

The PDEVS mathematical representation of this model is given by equation 3.16.

$$PDEVS_{battery_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.16}$$

Where :

$$X = \{(Prod, v) \mid v \in [0; 2])\} \cup \{(Ctrl_cons, v) \mid v \in [0; 1]\}$$
$$Y = \{(Capacity_output, v) \mid v \in \mathbb{N}\}$$
$$S = \{(\boldsymbol{Phase}, Capa) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_grid}\text{”}, \text{“}\boldsymbol{Storage}\text{”}\}, Capa \in \mathbb{N})\}$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_grid}\text{”}, Capa), e, (Prod, v)\big) = (\text{“}\boldsymbol{Wait_grid}\text{”}, Capa + v)$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_grid}\text{”}, Capa), e, (Ctrl_cons, v)\big) = (\text{“}\boldsymbol{Storage}\text{”}, Capa - v)$$
$$\delta_{int}(\text{“}\boldsymbol{Storage}\text{”}, Capa) = (\text{“}\boldsymbol{Wait_grid}\text{”}, Capa)$$
$$\lambda(\boldsymbol{Phase}, Capa) = (Capacity_output, Capa)$$

$$ta(\text{“}\boldsymbol{Wait_grid}\text{”}, Capa) = \infty$$
$$ta(\text{“}\boldsymbol{Storage}\text{”}, Capa) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.17. The initial state of an individual battery d, denoted s_{0d}, will set the battery capacity to 0. The initial phase is set to “$\boldsymbol{Wait_grid}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait_grid}\text{”}, 0) \tag{3.17}$$

The inputs ports of the battery are named $Prod$ that can take values in $[\![0; 2]\!]$ and $Ctrl_cons$ that can take values in $[\![0; 1]\!]$. The output port named $Capacity_output$, that has the ability to take values in $\mathbb{N}$. The states of the model consist of a phase and a capacity variable. The phase, named $Phase$, can have two possible values: “$Wait_grid$” and “$Storage$”. The capacity variable, named $Comp$, can take values in $\mathbb{N}$. The system has an external transition function, denoted as δ_{ext}. Applying δ_{ext} when a value is received in the $Prod$ input computes the new $Capa$ value by adding the $Prod$ to the old $Capa$ value and keep the phase to “$Wait_grid$”. Applying δ_{ext} when a value is received in the $Ctrl_cons$ input computes the new $Capa$ value by adding the $Prod$ to the old $Capa$ value and changes the phase from “$Wait_grid$” to “$Storage$”. The internal transition function, denoted as δ_{int}, update the state and changes the phase from “$Storage$” to “$Wait_grid$”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, Capa)$, sends the value of $Capa$ through the $Capacity_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_grid$” linked to an infinite time-advance and “$Storage$” linked to a time-advance of 0.

Power converter Model

The power converter model behaves as a switch triggered by the electronic device 1. If it is ”ON”, it behave like a short circuit and compute the power flow to determine if their is enough power for the load consumption or not. If it is ”OFF”, it behaves like an open circuit. Its formal representation is provided in equation 3.18 and imaged with the figure 3.37 and a chronogram of its expected state behavior in figure 3.38.

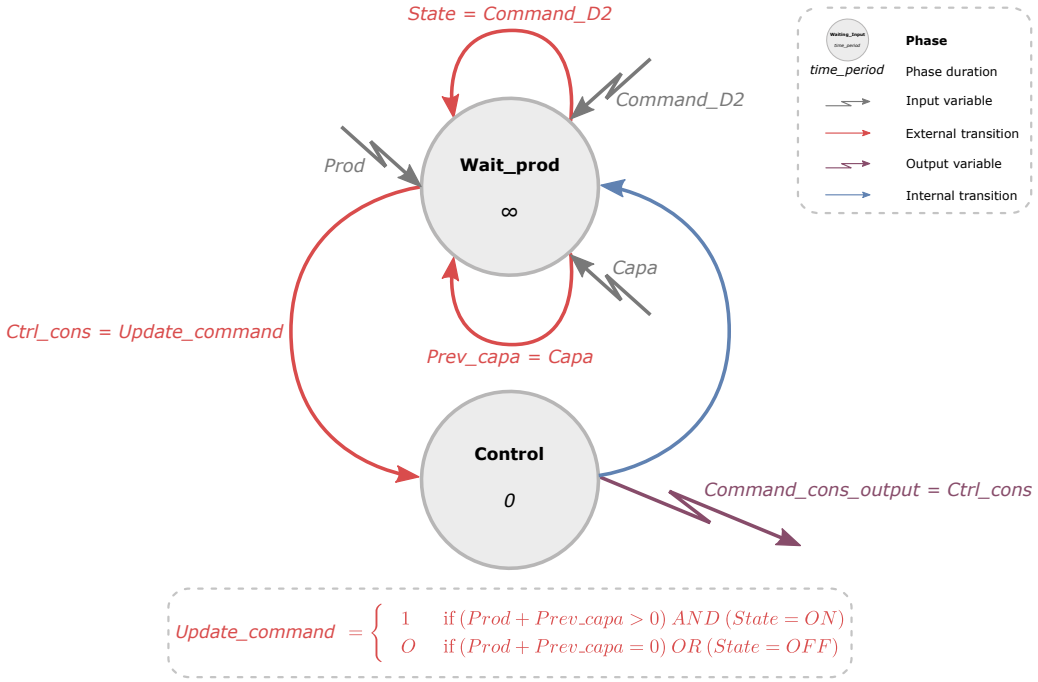


Figure 3.37: Graphical representation of the power converter C1 model states and dynamics

The proposed power converter model is composed of four variables: $Phase$, $State$, $Prev_capa$ and $Ctrl_cons$. The first variable, $Phase$, can take on two values: “$Wait_prod$” or “$Control$”,

and is used to model the system as a state machine. The second variable, *State*, is a numerical variable used to store the state of the electronic device that control the power converter. The third variable, *Prev_capa*, is a numerical variable used to store the value of the previous capacity of the battery. The fourth variable, *Ctrl_cons*, is a numerical variable used for calculating the command of the consumption of the load. The system is initially set to the "*Wait_prod*" phase, which phase associated time is infinite, the *State* is set to *ON*, the *Prev_capa* value is set to 0 and the *Ctrl_cons* is set to 1. From the "*Wait_prod*" state, the system waits for an input variable to be received in the input *Prod*, triggering an external transition. Based on the different variables of the grid, the power converter will let the load consume or not its nominal power. This transition is performed by the *Update_command* function to recompute the *Ctrl_cons* with the updated control value in input and updates the phase to "*Control*". This triggers the output function, which outputs the value of the *Ctrl_cons* in *Command_cons_output* output. After the output function is complete, the internal transition is executed, which updates the phase to "*Wait_prod*". From the "*Wait_prod*" state, the system waits for an input variable to be received in the input *Capa*, triggering an external transition. The value of the old battery capacity will be stored with the *Capa* variable. This transition recomputes the *Ctrl_cons* with the updated control value in input and keep the phase to "*Wait_prod*". From the "*Wait_prod*" state, the system waits for an input variable to be received in the input *Command_D2*, triggering an external transition. The value of the command state of D2 will be stored with the *State* variable. This transition updates the *State* with the updated control value in input and keep the phase to "*Wait_prod*". This process repeats indefinitely.

The chronogram in figure 3.38 succinctly illustrates the expected behavior of the $Ctrl_cons$ variable as a result of the production, previous battery capacity and command received in input.

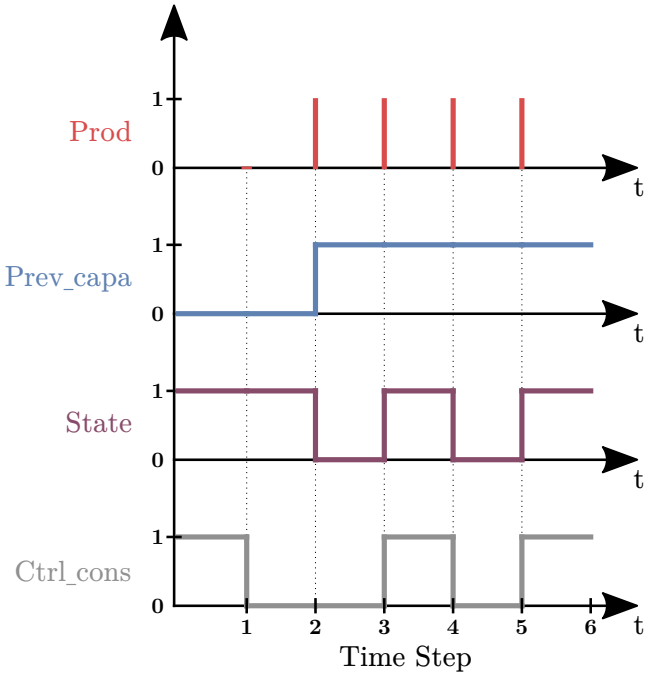


Figure 3.38: Chronogram of the behaviour of $Ctrl_cons$ in the power converter model

The PDEVS mathematical representation of this model is given by equation 3.18.

$$PDEVS_{converter_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.18}$$

Where :

$X = \{(Command_D2, v) \mid \in \{ON, OFF\}\} \cup \{(Capa, v) \mid v \in \mathbb{N}\} \cup \{(Prod, v) \mid v \in [0; 2]\}$
$Y = \{(Command_cons_output, v) \mid v \in [\![0; 1]\!]\}$
$S = \{(\boldsymbol{Phase}, State, Prev_capa, Ctrl_cons) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_prod}\text{”}, \text{“}\boldsymbol{Control}\text{”}\},$

$$State \in \{ON, OFF\}, Prev_capa \in \mathbb{N}, Ctrl_cons \in [\![0;1]\!]\}$$

$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_grid}\text{”}, State, Prev_capa, Ctrl_cons), e, (Command_D2, v)\big) = (\text{“}\boldsymbol{Wait_grid}\text{”}, v, Prev_capa, Ctrl_cons)$$

$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_grid}\text{”}, State, Prev_capa, Ctrl_cons), e, (Capa, v)\big) = (\text{“}\boldsymbol{Wait_grid}\text{”}, State, v, Ctrl_cons)$$

$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_prod}\text{”}, State, Prev_capa, Ctrl_cons), e, (Prod, v)\big) = (\text{“}\boldsymbol{Control}\text{”}, State, Prev_capa, Update_command)$$

$$\text{With } Update_command = \begin{cases} 1 & \text{if } (v + Prev_capa > 0)\ AND\ (State = ON) \\ 0 & else \end{cases}$$

$$\delta_{int}(\text{“}\boldsymbol{Control}\text{”}, State, Prev_capa, Ctrl_cons) = (\text{“}\boldsymbol{Wait_prod}\text{”}, State, Prev_capa, Ctrl_cons)$$

$$\lambda(\boldsymbol{Phase}, State, Prev_capa, Ctrl_cons) = (Command_cons_output, Ctrl_cons)$$

$$ta(\text{“}\boldsymbol{Wait_prod}\text{”}, State, Prev_capa, Ctrl_cons) = \infty$$

$$ta(\text{“}\boldsymbol{Command}\text{”}, State, Prev_capa, Ctrl_cons) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.19. The initial state of an individual power converter d, denoted s_{0d}, will set the converter's state to ON, the previous capacity of the power converter to 0 and the control of the load to 1. The initial phase is set to “$\boldsymbol{Wait_prod}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait_prod}\text{”}, ON, 0, 1) \tag{3.19}$$

The inputs ports X of the power converter model are named $Command_D2$ that can take values in $\{ON, OFF\}$, $Capa$ that can take values in $\mathbb{N}$ and $Prod$ that can take values in $[\![0;2]\!]$. The output port Y, named $Command_cons_output$, also has the ability to take values in $[\![0;1]\!]$. The states of the model, denoted as S, consist of a phase, a state variable, a battery capacity variable and a consumption control variable. The phase, named $Phase$, can have two possible values: “$Wait_prod$” and “$Control$”. The state variable, named $State$, can take values in $\{ON, OFF\}$. The battery capacity variable, named $Capa$, can take values in $\mathbb{N}$. The consumption control variable, named $Ctrl_cons$, can take values in $[\![0;1]\!]$. The system has an external transition function, denoted as δ_{ext}. Applying δ_{ext} when a value is received in the $Command_D2$ input saves the value of the new command in $State$. Applying δ_{ext} when a value is received in the $Capa$ input saves the value of the battery capacity in $Prev_capa$. Applying δ_{ext} when a value is received in the $Prod$ input determines the value of the load command in function of the grid capacity and its state, saves it in $Ctrl_cons$ and changes the phase from “$Wait_prod$” to “$Control$”. The internal transition function, denoted as δ_{int}, changes the phase from “$Consumption$” to “$Wait_control$”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, State, Prev_capa, Ctrl_cons)$, sends the value of $Ctrl_cons$ through the $Command_cons_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_prod$” linked to an infinite time-advance and “$Control$” linked to a time-advance of 0.

Wallet 1 Model

The wallet 1 model is triggered by the financial institution that sends it a payment. It computes its new account statement and sends the payment to the financial institution. Its formal representation is provided in equation 3.20 and imaged with the figure 3.39 and a chronogram of its expected state behavior in figure 3.40.

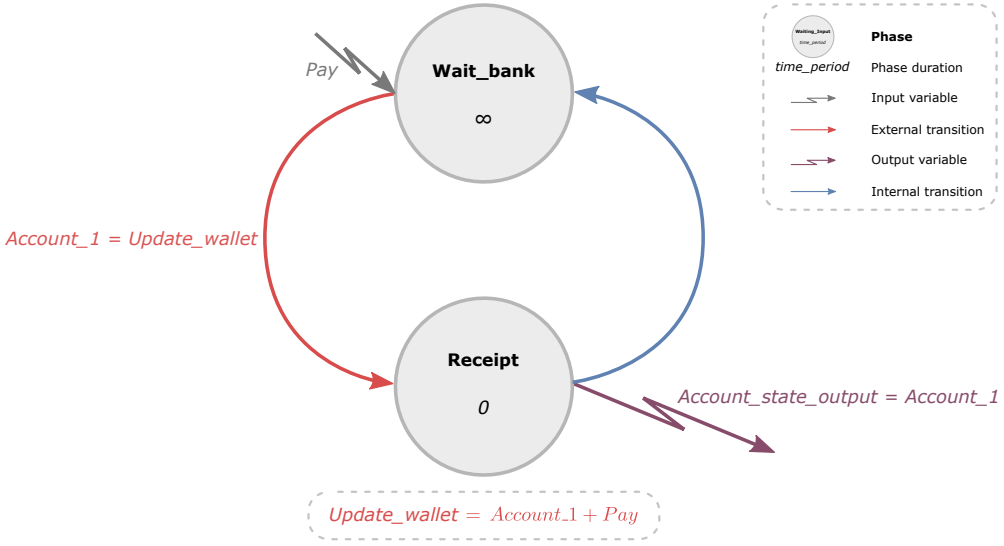


Figure 3.39: Graphical representation of the wallet W1 model states and dynamics

The proposed producer wallet model is composed of two variables: *Phase* and *Account_1*. The first variable, *Phase*, can take on two values: "*Wait_bank*" or "*Receipt*", and is used to model the system as a state machine. The second variable, *Account_1*, is a numerical variable used for calculating the amount of money in the producer wallet. The system is initially set to the "*Wait_bank*" phase, which phase associated time is infinite and the *Account_1* is set to 0. From the "*Wait_bank*" state, the system waits for an input variable to be received in the input *Pay*, triggering an external transition. Based on the amount received from the bank, a new account value will be computed. This transition is performed by the *Update_wallet* function to recompute the *Account_1* value with the value of the payment from the bank in input and updates the phase to "*Receipt*". This triggers the output function, which outputs the value of the *Account_1* in *Account_state_output* output. After the output function is complete, the internal transition is executed to update the phase to "*Wait_bank*". This process repeats indefinitely.

The chronogram in figure 3.40 succinctly illustrates the expected behavior of the $Account_1$ variable as a result of the payment received in input.

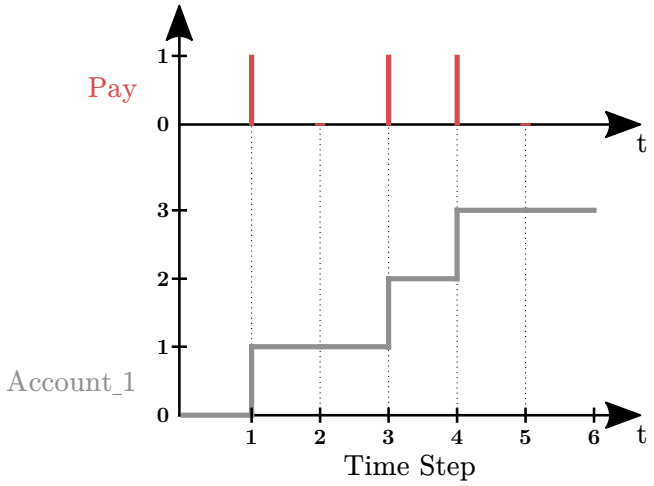


Figure 3.40: Chronogram of the behaviour of $Account_1$ in the wallet 1 model

The PDEVS mathematical representation of this model is given by equation 3.20.

$$PDEVS_{wallet_1_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.20}$$

Where :

$$X = \{(Pay, v) \mid v \in \mathbb{N}\}$$
$$Y = \{(Account_state_output, v) \mid v \in \mathbb{N}\}$$
$$S = \{(\boldsymbol{Phase}, Account_1) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_bank}\text{”}, \text{“}\boldsymbol{Receipt}\text{”}\}, Account_1 \in \mathbb{N}\}$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_bank}\text{”}, Account_1), e, (Pay, v)\big) = (\text{“}\boldsymbol{Receipt}\text{”}, Account_1 + Pay)$$
$$\delta_{int}(\text{“}\boldsymbol{Receipt}\text{”}, Account_1) = (\text{“}\boldsymbol{Wait_bank}\text{”}, Account_1)$$
$$\lambda(\boldsymbol{Phase}, Account_1) = (Account_state_output, Account_1)$$
$$ta(\text{“}\boldsymbol{Wait_bank}\text{”}, Account_1) = \infty$$
$$ta(\text{“}\boldsymbol{Receipt}\text{”}, Account_1) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.21. The initial state of an individual wallet d, denoted s_{0d}, will set the wallet's value to 0. The initial phase is set to “$\boldsymbol{Wait_bank}$”.

$$s_{0d}(\text{“}\boldsymbol{Wait_bank}\text{”}, 0) \tag{3.21}$$

The input port X of the example model are named Pay can take values in $\mathbb{N}$. The output port Y, named $Account_state_output$, also has the ability to take values in $\mathbb{N}$. The states of the model, denoted as S, consist of a phase and an account variable. The phase, named $Phase$, can have two possible values: “$Wait_bank$” and “$Receipt$”. The account variable, named $Account_1$, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the Pay variable to compute the new $Account_1$ value by adding the payment to the current $Account_1$ value and changes the phase from “$Wait_bank$” to “$Receipt$”. The internal transition function, denoted as δ_{int}, computes the new phase from “$Receipt$” to “$Wait_bank$”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(\text{“}Receipt\text{”}, Acount_1)$, sends the value of $Account_1$ through the $Account_state_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_bank$” linked to an infinite time-advance and “$Receipt$” linked to a time-advance of 0.

Wallet 2 Model

The wallet 2 model is triggered by the financial institution that send a bill. It computes its new account statement and sends the receipt to its owner acceptance. Its formal representation is provided in equation 3.22 and imaged with the figure 3.41 and a chronogram of its expected state behavior in figure 3.42.

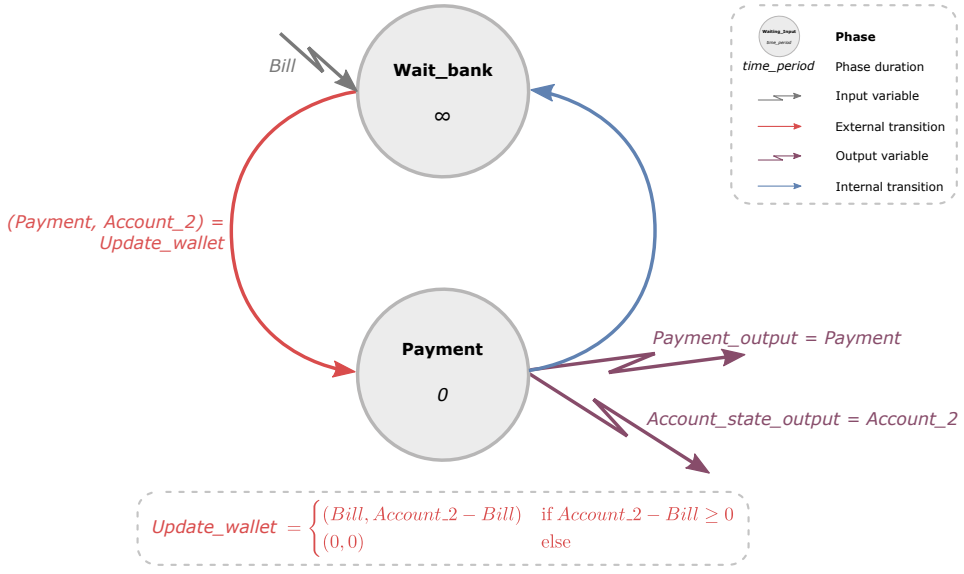


Figure 3.41: Graphical representation of the wallet W2 model states and dynamics

The proposed consumer wallet model is composed of three variables: *Phase*, *Payment* and *Account_2*. The first variable, *Phase*, can take on two values: "*Wait_bank*" or "*Paying*", and is used to model the system as a state machine. The second variable, *Payment*, is a numerical variable used for storing the payment value depending on the consumption. The second variable, *Account_2*, is a numerical variable used for calculating the amount of money in the producer wallet. The system is initially set to the "*Wait_bank*" phase, which phase associated time is infinite, the *Payment* is set to 0 and the *Account_2* is set to 100. From the "*Wait_bank*" state, the system waits for an input variable to be received in the input *Bill*, triggering an external transition. Based on the value of the bill received from the bank, a new payment and account value will be calculated. This transition is performed by the *Update_wallet* function to recompute the *Payment* and *Account_2* values with the value of the bill from the bank in input and updates the phase to "*Payment*". This triggers two output functions. One sends the value of the *Account_2* in *Account_state_output* output. the other sends the value of *Payment* in *Payment_output* output. After the output function is complete, the internal transition is executed to update the phase to "*Wait_bank*". This process repeats indefinitely.

The chronogram in figure 3.42 succinctly illustrates the expected behavior of the $Account_1$ and *Payment* variables as a result of the bill received in input.

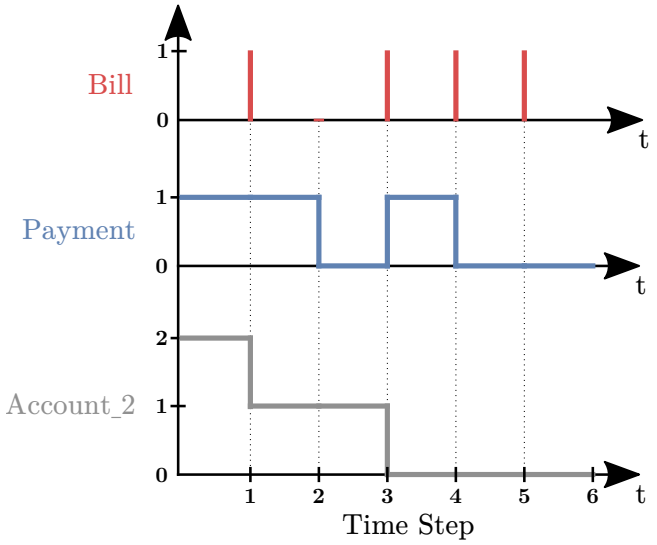


Figure 3.42: Chronogram of the behaviour of $Account_2$ and *Payment* in the wallet 2 model

The PDEVS mathematical representation of this model is given by equation 3.22.

$$PDEVS_{wallet_2_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.22}$$

Where :

$X = \{(Bill, v) \mid v \in \mathbb{N}\}$

$Y = \{(Payment_output, v) \mid v \in \mathbb{N}\} \times \{(Account_state_output, v) \mid v \in \mathbb{N}\}$

$S = \{(\boldsymbol{Phase}, Payment, Account_2) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_bank}\text{”}, \text{“}\boldsymbol{Paying}\text{”}\}), Payment \in \mathbb{N}, Account_2 \in \mathbb{N}\}$

$\delta_{ext}\big((\text{“}\boldsymbol{Wait_bank}\text{”}, Payment, Account_2), e, (Bill, v)\big) = (\text{“}\boldsymbol{Paying}\text{”}, Update_wallet)$

$$\text{With } Update_wallet = \begin{cases} (v, Account_2 - v) & \text{if } Account_2 - v \geq 0 \\ (0, 0) & \text{else} \end{cases}$$

$\delta_{int}(\text{“}\boldsymbol{Paying}\text{”}, Payment, Account_2) = (\text{“}\boldsymbol{Wait_bank}\text{”}, Payment, Account_2)$

$\lambda(\boldsymbol{Phase}, Payment, Account_2) = \{(Payment_output, Payment), (Account_state_output, Account_2)\}$

$ta(\text{“}\boldsymbol{Wait_bank}\text{”}, Payment) = \infty$

$ta(\text{“}\boldsymbol{Paying}\text{”}, Payment) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.23. The initial state of an individual wallet d, denoted s_{0d}, will set the wallet's value to 100. The initial phase is set to "***Wait_bank***".

$$s_{0d} = (\text{“}\boldsymbol{Wait_bank}\text{”}, 0, 100) \tag{3.23}$$

The input port X of the example model are named $Bill$ can take values in $\mathbb{N}$. The output port Y, named $Payment_output$, also has the ability to take values in $\mathbb{N}$. The states of the model, denoted as S, consist of a phase a payment variable and an account variable. The phase, named $Phase$, can have two possible values: "$Wait_bank$" and "$Paying$". The payment variable, named $Payment$, can take values in $\mathbb{N}$. The account variable, named $Account_1$, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the $Bill$ variable to compute the new $Account_1$ value and the $Payment$ value using the abstract function $Update_wallet()$ by subtracting the bill to the current $Account_2$ value, replacing the $Payment$ value by the $Bill$ value and changes the phase from "$Wait_bank$" to "$Receipt$". The internal transition function, denoted as δ_{int}, computes the new phase from "$Paying$" to "$Wait_bank$". In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(\text{“}Paying\text{”}, Payment, Acount_2)$, sends the value of $Payment$ through the $Payment_output$ port and the value of $Account_2$ through the $Account_state_output$ to the connected component. The phases are linked to specific time-advances, with "$Wait_bank$" linked to an infinite time-advance and "$Paying$" linked to a time-advance of 0.

Financial institution Model

The financial institution model is triggered by the electronic device 2 that send the consumption of the load. It computes the new bill and sends it to the wallet 1. Once the payment is received it sends it to the wallet 2. Its formal representation is provided in equation 3.24 and imaged with the figure 3.43 and a chronogram of its expected state behavior in figure 3.44.

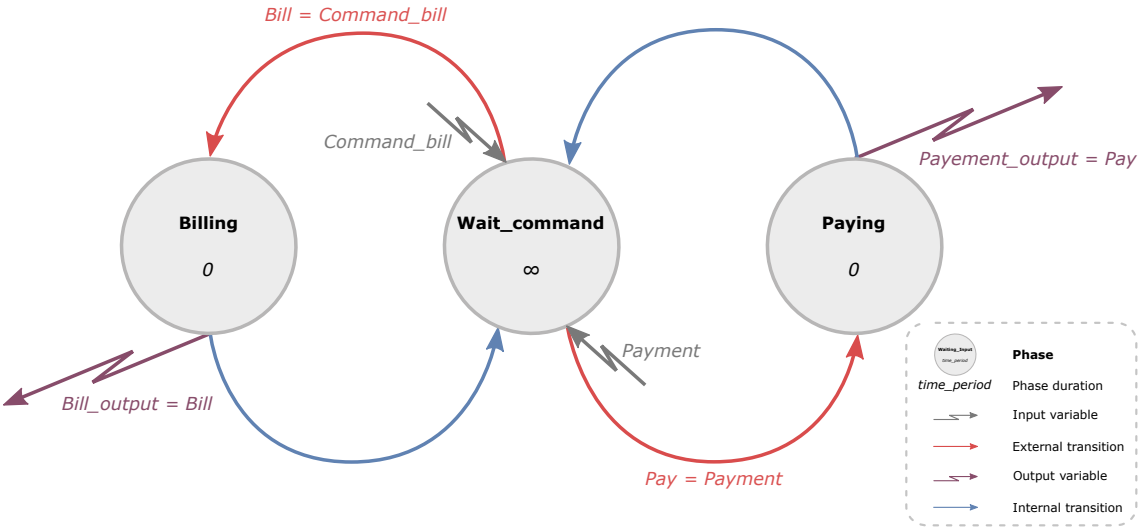


Figure 3.43: Graphical representation of the bank B1 model states and dynamics

The proposed financial institution model is composed of three variables: $Phase$, Pay and $Bill$. The first variable, $Phase$, can take on three values: "$Wait_command$", "$Paying$" or "$Billing$", and is used to model the system as a state machine. The second variable, Pay, is a numerical variable used for storing the payment coming from the consumer. The third variable, $Bill$, is a numerical variable used for storing the billing coming from the consumption of energy. The system is initially set to the "$Wait_command$" phase, which phase associated time is infinite, Pay is set to 0 and $Bill$ is set to 1. From the "$Wait_command$" state, the system waits for an input variable to be received in the input $Command_bill$, triggering an external transition. Based on the value of the bill received from the electronic device, a new bill will be computed. This transition

recomputes the *bill* with the command bill value in input and updates the phase to "*Billing*". This triggers the output function, which outputs the value of the *Bill* in *Bill_output* output. After the output function is complete, the internal transtion is executed, which updates the phase to "*Wait_command*". From their, the system waits for an input variable to be received in the input *Payment*, triggering an external transition. Based on the value of the payment received from the consumer account, a new pay value will be computed. This transition recomputes the *pay* value with the payment value received from the consumer in input and updates the phase to "*Paying*". This triggers the output function, which outputs the current value of the *Pay* in *Payment_output*. After the output function is complete, the internal transition is executed, which updates the phase to "*Wait_command*". This process repeats indefinitely.

The chronogram in figure 3.44 succinctly illustrates the expected behavior of the *Bill* and *Pay* variables as a result of the bill and payment received in input.

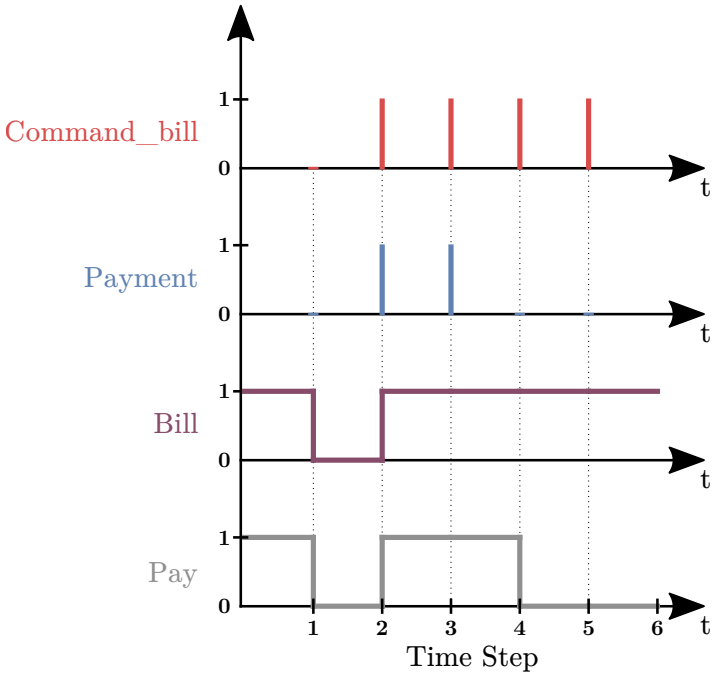


Figure 3.44: Chronogram of the behaviour of *Bill* and *Pay* in the financial institution model

The PDEVS mathematical representation of this model is given by equation 3.24.

$$PDEVS_{bank_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.24}$$

Where :

$X = \{(p, v) \mid p \in \{Command_bill, Payment\}, v \in \mathbb{N}\}$
$Y = \{(p, v) \mid p \in \{Payment_output, Bill_output\}, v \in \mathbb{N}\}$
$S = \{(\boldsymbol{Phase}, Pay, Bill) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_command}\text{”}, \text{“}\boldsymbol{Paying}\text{”}, \text{“}\boldsymbol{Billing}\text{”}\}, Pay \in \mathbb{N}, Bill \in \mathbb{N}\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait_command}\text{”}, Pay, Bill), e, (Command_bill, v)\big) = (\text{“}\boldsymbol{Billing}\text{”}, Pay, v)$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait_command}\text{”}, Pay, Bill), e, (Payment, v)\big) = (\text{“}\boldsymbol{Paying}\text{”}, v, Bill)$
$\delta_{int}(\text{“}\boldsymbol{Billing}\text{”}, Pay, Bill) = (\text{“}\boldsymbol{Wait_command}\text{”}, Bill, Bill)$
$\delta_{int}(\text{“}\boldsymbol{Paying}\text{”}, Pay, Bill) = (\text{“}\boldsymbol{Wait_command}\text{”}, Pay, Bill)$
$\lambda(\text{“}\boldsymbol{Billing}\text{”}, Pay, Bill) = (Bill_output, Bill)$
$\lambda(\text{“}\boldsymbol{Paying}\text{”}, Pay, Bill) = (Payment_output, Pay)$
$ta(\text{“}\boldsymbol{Wait_command}\text{”}, Pay, Bill) = \infty$
$ta(\text{“}\boldsymbol{Paying}\text{”}, Pay, Bill) = 0$
$ta(\text{“}\boldsymbol{Billing}\text{”}, Pay, Bill) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.25. The initial state of an individual bank d, denoted s_{0d}, will set the bank's pay to 0 and bill to 0. The initial phase is set to "***Wait_command***".

$$s_{0d} = (\text{“}\boldsymbol{Wait_command}\text{”}, 0, 0) \tag{3.25}$$

The inputs ports X of the bank model are named $Command_bill$ which can take values in $\mathbb{N}$ and $Payment$ which can take values in $\mathbb{N}$. The outputs ports Y are named $Payment_output$ which can take values in $\mathbb{N}$ and $Bill_output$ which also takes values in $\mathbb{N}$. The states of the model, denoted as S, consist of a phase, a payment variable and a billing variable. The phase, named $Phase$, can have two possible values: "$Wait_command$", "$Paying$" and "$Billing$". The payment variable, named Pay, can take values in $\mathbb{N}$. The billing variable, named $Bill$, can take values in $\mathbb{N}$. The system has an external transition function, denoted as δ_{ext}. Applying δ_{ext} when a value is received in the $Command_bill$ input saves the value of the new bill based on the consumption in $Bill$. Applying δ_{ext} when a value is received in the $Payment$ input saves the value of the payment made by the consumer in Pay. The system has an internal transition function, denoted as δ_{int}. Applying δ_{int} when the phase is "$Billing$", changes the phase from "$Billing$" to "$Wait_command$". Applying δ_{int} when the phase is "$Paying$", changes the phase from "$Paying$" to "$Wait_command$". In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(\text{“}Billing\text{”}, Pay, Bill)$, sends the value of $Bill$ through the $Bill_output$ port to the connected component. The output function, denoted as $\lambda(\text{“}Paying\text{”}, Pay, Bill)$, sends the value of Pay through the $Payment_output$ port to the connected component. The phases are linked to specific time-advances, with "$Wait_command$" linked to an infinite time-advance, "$Paying$" linked to a time-advance of 0 and "$Billing$" linked to a time-advance of 0.

Measurement M1 Model

The measurement M1 model is triggered by the consumption ($Cons$) of an associated load, which is then sent out. Its formal representation is provided in equation 3.26 and imaged with the figure 3.45 and a chronogram of its expected state behavior in figure 3.46.

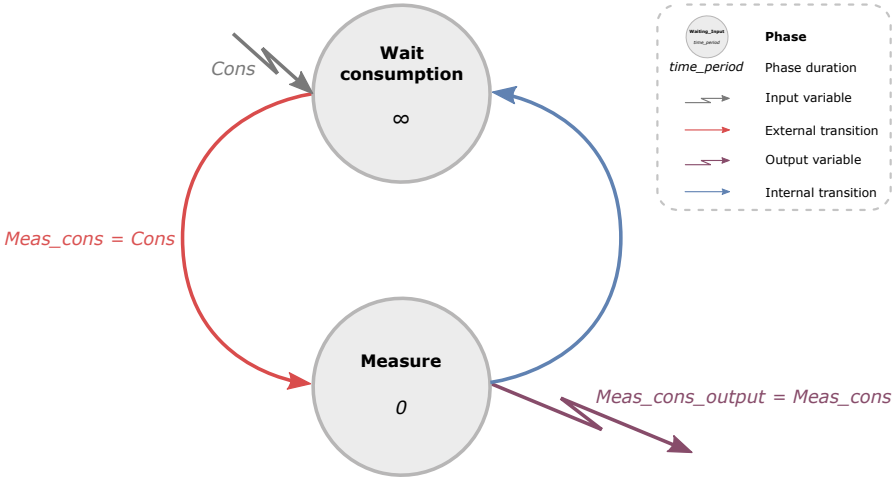


Figure 3.45: Graphical representation of the measurement M1 model states and dynamics

The proposed measurement model is composed of two variables: $Phase$ and $Meas_cons$. The first variable, $Phase$, can take on two values: "$Wait_consumption$" or "$Measure$", and is used to model the system as a state machine. The second variable, $Meas_cons$, is a numerical variable used store the measured consumption of the load. The system is initially set to the "$Wait_consumption$" phase, which phase associated time is infinite. and the $Meas_cons$ is set to 0. From the "$Wait_consumption$" state, the system waits for an input variable to be received in the input $Cons$, triggering an external transition. Based on the value of the consumption received from the load, a new measure will be computed. This transition updates the $Meas_cons$ with the

current value of the consumption in input and updates the phase to "$Measure$". This triggers the output function, which outputs the current value of the $Meas_cons$ in $Meas_cons_output$. After the output function is complete, the internal transition is executed, which updates the phase to "$Wait_cons$". This process repeats indefinitely.

The chronogram in figure 3.46 succinctly illustrates the expected behavior of the $Meas_cons$ variables as a result of the consumption received in input.

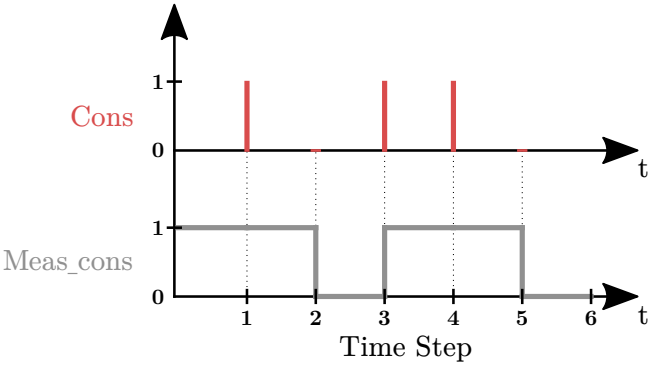


Figure 3.46: Chronogram of the behaviour of $Meas_cons$ in the measure 1 model

The PDEVS mathematical representation of this model is given by equation 3.26.

$$PDEVS_{measure_1_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.26}$$

Where :

$X = \{(Cons, v) \mid v \in [\![0; 1]\!]\}$
$Y = \{(Meas_cons_output, v) \mid v \in [\![0; 1]\!]\}$
$S = \{(\boldsymbol{Phase}, Meas_cons) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_consumption}\text{”}, \text{“}\boldsymbol{Measure}\text{”}\}, Meas_cons \in [\![0; 1]\!]\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait_consumption}\text{”}, Meas_cons), e, (Cons, v)\big) = (\text{“}\boldsymbol{Measure}\text{”}, v)$
$\delta_{int}(\text{“}\boldsymbol{Measure}\text{”}, Meas_cons) = (\text{“}\boldsymbol{Wait_consumption}\text{”}, Meas_cons)$
$\lambda(\boldsymbol{Phase}, Meas_cons) = (Meas_cons_output, Meas_cons)$
$ta(\text{“}\boldsymbol{Wait_consumption}\text{”}, Meas_cons) = \infty$
$ta(\text{“}\boldsymbol{Measure}\text{”}, Meas_cons) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.27. The initial state of an individual measure d, denoted s_{0d}, will set the measure's value to 0. The initial phase is set to "$\boldsymbol{Wait_consumption}$".

$$s_{0d} = (\text{“}\boldsymbol{Wait_consumption}\text{”}, 0) \tag{3.27}$$

The input port X of the example model are named $Cons$ can take values in $[\![0; 1]\!]$. The output port Y, named $Meas_cons_output$, also has the ability to take values in $[\![0; 1]\!]$. The states of the model, denoted as S, consist of a phase and a measure variable. The phase, named $Phase$, can have two possible values: "$Wait_consumption$" and "$Measure$". The measure variable, named $Meas_cons$, can take values in $[\![0; 1]\!]$. The external transition function, denoted as δ_{ext}, uses the $Cons$ variable to compute the new $Meas_cons$ Value by replacing it by the $Cons$ value and changes the phase from "$Wait_consumption$" to "$Measure$". The internal transition function, denoted as δ_{int}, changes the phase from "$Measure$" to "$Wait_consumption$". In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, Meas_cons)$, sends the value of $Meas_cons$ through the $Meas_cons_output$ port to the connected component. The phases are linked to specific time-advances, with "$Wait_consumption$" linked to an infinite time-advance and "$Measure$" linked to a time-advance of 0.

Measurement M2 Model

The measurement M2 model is triggered by the wallet state (*Account*) of an associated load, which is then sent out. Its formal representation is provided in equation 3.28 and imaged with the figure 3.47 and a chronogram of its expected state behavior in figure 3.48.

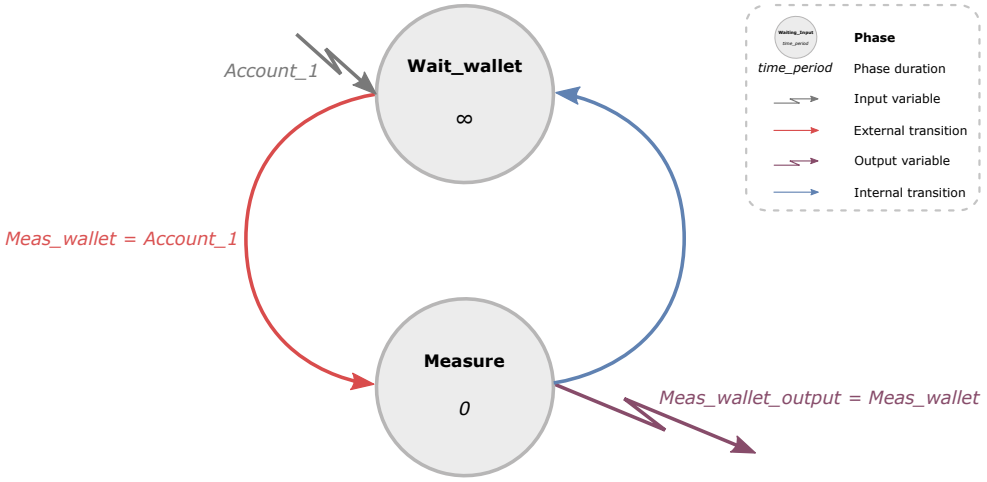


Figure 3.47: Graphical representation of the measure M2 model states and dynamics

The proposed measurement model is composed of two variables: *Phase* and *Meas_wallet*. The first variable, *Phase*, can take on two values: "*Wait_wallet*" or "*Measure*", and is used to model the system as a state machine. The second variable, *Meas_wallet*, is a numerical variable used store the measured wallet amount value of the consumer. The system is initially set to the "*Wait_wallet*" phase, which phase associated time is infinite and the *Meas_wallet* is set to 0. From the "*Wait_wallet*" state, the system waits for an input variable to be received in the input *Account_2*, triggering an external transition. Based on the value of the account amount received from the consumer wallet, a new measure will be computed. This transition updates the measure of the *Meas_wallet* with the current value of the account in input and updates the phase to "*Measure*". This triggers the output function, which outputs the value of the *Meas_wallet* in *Meas_wallet_output* output. After the output function is complete, the internal transition is executed, which updates the phase to "*Wait_wallet*". This process repeats indefinitely.

The chronogram in figure 3.48 succinctly illustrates the expected behavior of the *Meas_wallet* variables as a result of the account 2 status received in input.

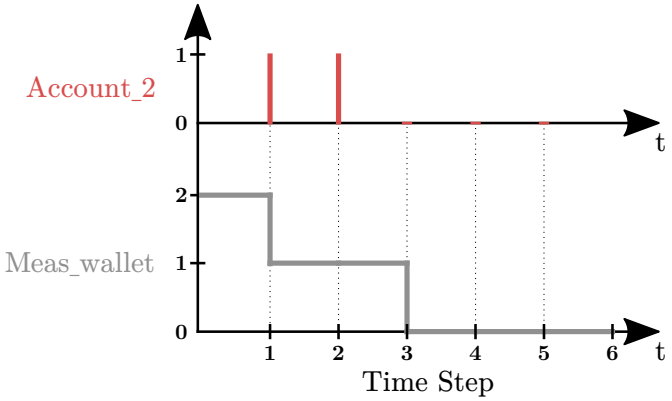


Figure 3.48: Chronogram of the behaviour of *Meas_wallet* in the measure 2 model

The PDEVS mathematical representation of this model is given by equation 3.28.

$$PDEVS_{measure_2_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.28}$$

Where :

$$X = \{(Account_2, v) \mid v \in \mathbb{N}\}$$
$$Y = \{(Meas_wallet_output, v) \mid v \in \mathbb{N}\}$$
$$S = \{(\boldsymbol{Phase}, Meas_wallet) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_wallet}\text{”}, \text{“}\boldsymbol{Measure}\text{”}\}, Meas_wallet \in \mathbb{N}\}$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait_wallet}\text{”}, Meas_wallet), e, (Account_2, v)\big) = (\text{“}\boldsymbol{Measure}\text{”}, v)$$
$$\delta_{int}(\text{“}\boldsymbol{Payment}\text{”}, Meas_wallet) = (\text{“}\boldsymbol{Wait_wallet}\text{”}, Meas_wallet)$$
$$\lambda(\boldsymbol{Phase}, Meas_wallet) = (Meas_wallet_output, Meas_wallet)$$
$$ta(\text{“}\boldsymbol{Wait_wallet}\text{”}, Meas_wallet) = \infty$$
$$ta(\text{“}\boldsymbol{Measure}\text{”}, Meas_wallet) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.29. The initial state of an individual measure d, denoted s_{0d}, will set the measure's value to 0. The initial phase is set to “$\boldsymbol{Wait_wallet}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait_wallet}\text{”}, 0) \tag{3.29}$$

The input port X of the example model are named $Account_2$ can take values in $\mathbb{N}$. The output port Y, named $Meas_wallet_output$, also has the ability to take values in $\mathbb{N}$. The states of the model, denoted as S, consist of a phase and a measure variable. The phase, named $Phase$, can have two possible values: “$Wait_wallet$” and “$Measure$”. The measure variable, named $Meas_wallet$, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the $Account_2$ variable to compute the new $Meas_wallet$ value by replacing it by the $Account_2$ value and changes the phase from “$Wait_wallet$” to “$Measure$”. The internal transition function, denoted as δ_{int}, changes the phase from “$Measure$” to “$Wait_wallet$”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, Meas_wallet)$, sends the value of $Meas_wallet$ through the $Meas_wallet_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_wallet$” linked to an infinite time-advance and “$Measure$” linked to a time-advance of 0.

Electronic device D1 model The Electronic device D1 model is triggered by the measured consumption ($Meas_Cons$) of an associated load, and then outputs the $Command_bill$ of its associated $Wallet$. Its formal representation is provided in equation 3.30 and imaged with the figure 3.49 and a chronogram of its expected state behavior in figure 3.50.

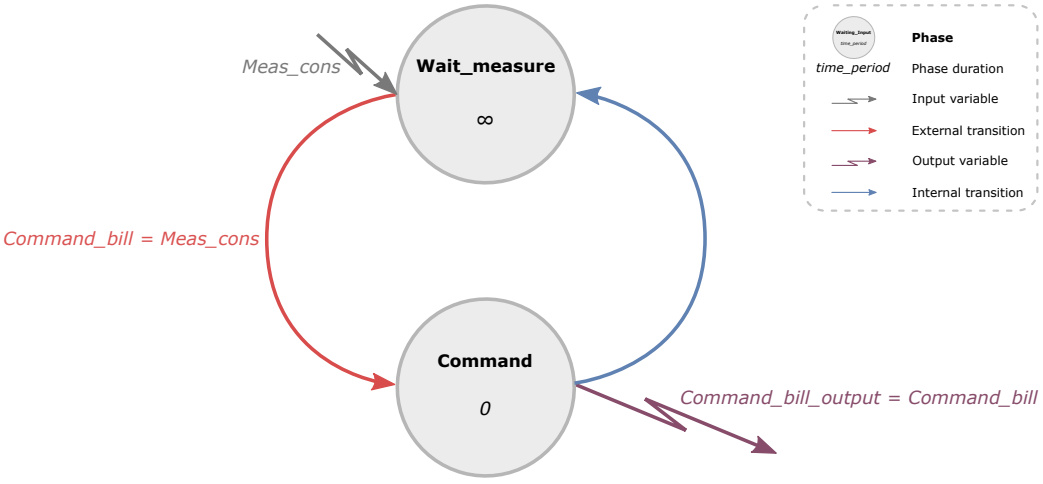


Figure 3.49: Graphical representation of the electronic device D1 model states and dynamics

The proposed electronic device model is composed of two variables: $Phase$ and $Command_bill$. The first variable, $Phase$, can take on two values: “$Wait_measure$” or “$Command$”, and is used

to model the system as a state machine. The second variable, *Command_bill*, is a numerical variable used store the value of the next consumption bill. The system is initially set to the "*Wait_consumption*" phase,which phase associated time is infinite and the *Meas_cons* is set to 0. From the "*Wait_measure*" state, the system waits for an input variable to be received in the input *Meas_cons*, triggering an external transition. Based on the value of the measured consumption received from the measure, a new command will be computed. This transition updates the command of the energy bill based on the current value of the measured consumption in input and updates the phase to "*Command*". This triggers the output function, which outputs the value of the *Command_bill* in *Command_bill_output* output. After the output function is complete, the internal transition is executed, which updates the phase to "*Wait_measure*". This process repeats indefinitely.

The chronogram in figure 3.50 succinctly illustrates the expected behavior of the *Command_bill* variables as a result of the consumption measure received in input.

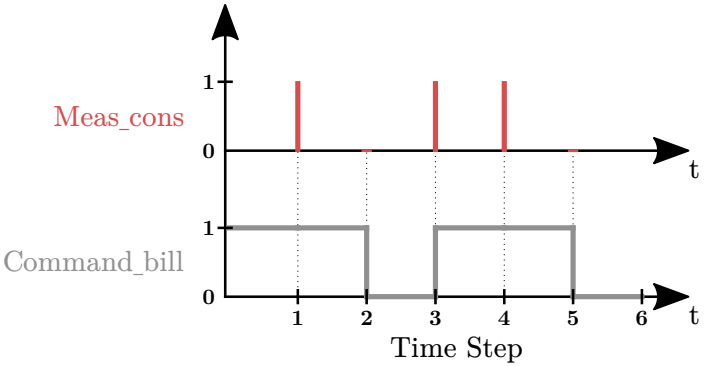


Figure 3.50: Chronogram of the behaviour of *Command_bill* in the electronic device 1 model

The PDEVS mathematical representation of this model is given by equation 3.30.

$$PDEVS_{electronic_device_1_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.30}$$

Where :

$X = \{(Meas_cons, v) \mid v \in \mathbb{N}\}$
$Y = \{(Command_bill_output, v) \mid v \in \mathbb{N}\}$
$S = \{(\boldsymbol{Phase}, Command_bill \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_measure}\text{”}, \text{“}\boldsymbol{Command}\text{”}\}, Command_bill \in \mathbb{N}\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait_measure}\text{”}, Command_bill), e, (Meas_cons, v)\big) = (\text{“}\boldsymbol{Command}\text{”}, Meas_cons)$
$\delta_{int}(\text{“}\boldsymbol{Command}\text{”}, Command_bill) = (\text{“}\boldsymbol{Wait_measure}\text{”}, Command_bill)$
$\lambda(\boldsymbol{Phase}, Command_bill) = (Command_bill_output, Command_bill)$
$ta(\text{“}\boldsymbol{Wait_measure}\text{”}, Command_bill) = \infty$
$ta(\text{“}\boldsymbol{Command}\text{”}, Command_bill) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.31. The initial state of an individual electronic device d, denoted s_{0d}, will set electronic device's command to 0. The initial phase is set to "$\boldsymbol{Wait_measure}$".

$$s_{0d} = (\text{“}\boldsymbol{Wait_measure}\text{”}, 0) \tag{3.31}$$

The input port X of the example model are named *Meas_cons* can take values in $\mathbb{N}$. The output port Y, named *Command_bill_output*, also has the ability to take values in $\mathbb{N}$. The states of the model, denoted as S, consist of a phase and a command variable. The phase, named *Phase*, can have two possible values: "*Wait_measure*" and "*Command*". The command variable, named *Command*, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the *Meas_cons* variable to compute the new *Command_bill* Value using the

abstract function $Update_measure()$ by replacing it by the $Meas_cons$ value and changes the phase from "$Wait_measure$" to "$Command$". The internal transition function, denoted as δ_{int}, changes the phase from "$Command$" to "$Wait_measure$". In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, Command_bill)$, sends the value of $Command_bill$ through the $Command_bill_output$ port to the connected component. The phases are linked to specific time-advances, with "$Wait_measure$" linked to an infinite time-advance and "$Command$" linked to a time-advance of 0.

Electronic device D2 model

The power converter D2 model is triggered by the wallet state ($Wallet_{state}$) of an associated load, and then output the state "ON" or "OFF" of its associated power converter. Its formal representation is provided in equation 3.32 and imaged with the figure 3.51 and a chronogram of its expected state behavior in figure 3.52.

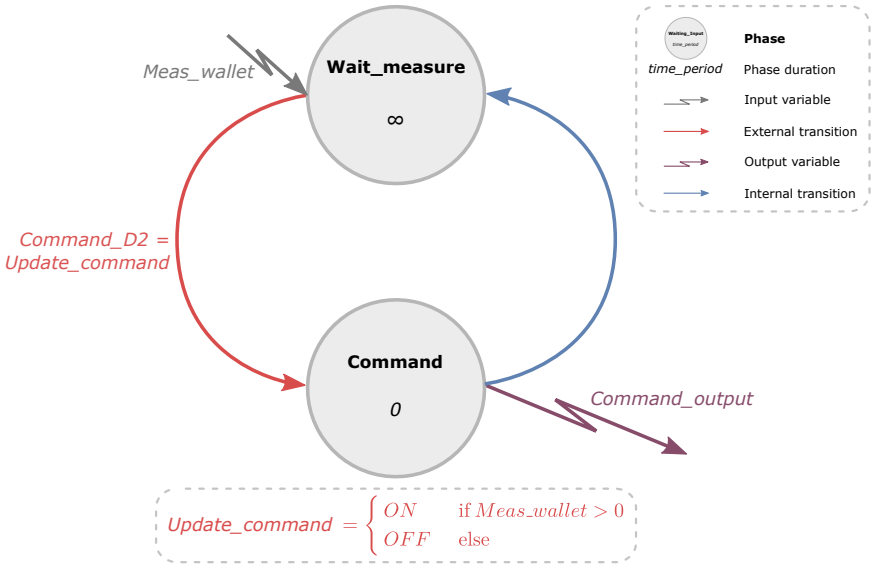


Figure 3.51: Graphical representation of the electronic device D2 model states and dynamics

The proposed electronic device model is composed of two variables: $Phase$ and $Command_D2$. The first variable, $Phase$, can take on two values: "$Wait_measure$" or "$Command$", and is used to model the system as a state machine. The second variable, $Command_D2$, is a variable used to store the state (ON or OFF) for the command of the load. The system is initially set to the "$Wait_measure$" phase, which phase associated time is infinite and the $Command_D2$ is set to ON. From the "$Wait_measure$" state, the system waits for an input variable to be received in the input $Meas_wallet$, triggering an external transition. Based on the value of the measured wallet amount received from the measure, the command will be set to OFF if the wallet is empty or ON if their are founds remaining. This transition is performed by the $Update_command$ function to update the command of the power converter based on the current value of the measured wallet amount in input and updates the phase to "$Command$". This triggers the output function, which outputs the value of the $Command_D2$ in $Command_output$ output. After the output function is complete, the internal transition is executed, which updates the phase to "$Wait_measure$". This process repeats indefinitely.

The chronogram in figure 3.52 succinctly illustrates the expected behavior of the $Command_D2$ variables as a result of the account measure received in input.

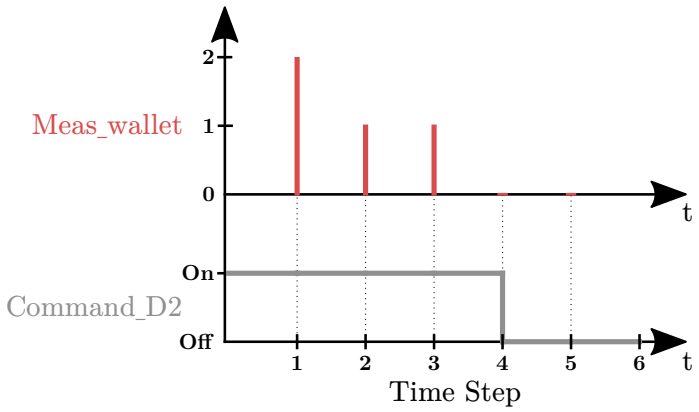


Figure 3.52: Chronogram of the behaviour of *Command_D2* in the electronic device 2 model

The PDEVS mathematical representation of this model is given by equation 3.32.

$$PDEVS_{electronic_device_2_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.32}$$

Where :

$X = \{(Meas_wallet, v) \mid v \in \mathbb{N}\}$
$Y = \{(Command_output, v) \mid v \in \{ON, OFF\}\}$
$S = \{(\boldsymbol{Phase}, Command_D2) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait_measure}\text{”}, \text{“}\boldsymbol{Command}\text{”}\}, Command_D2 \in \{ON, OFF\}\}$
$\delta_{ext}\big((\text{“}\boldsymbol{Wait_measure}\text{”}, Command_D2), e, (Meas_wallet, v)\big) = \begin{cases} (\text{“}\boldsymbol{Command}\text{”}, ON) & \text{if } v > 0 \\ (\text{“}\boldsymbol{Command}\text{”}, OFF) & \text{else} \end{cases}$
$\delta_{int}(\text{“}\boldsymbol{Command}\text{”}, Command_D2) = (\text{“}\boldsymbol{Wait_measure}\text{”}, Command_D2)$
$\lambda(\boldsymbol{Phase}, Command_D2) = (Command_output, Command_D2)$
$ta(\text{“}\boldsymbol{Wait_measure}\text{”}, Command_D2) = \infty$
$ta(\text{“}\boldsymbol{Command}\text{”}, Command_D2) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.33. The initial state of an individual electronic device d, denoted s_{0d}, will set the electronic device's command to ON. The initial phase is set to “$\boldsymbol{Wait_measure}$”.

$$s_{0d}(\text{“}\boldsymbol{Wait_measure}\text{”}, ON) \tag{3.33}$$

The input port X of the example model are named $Meas_wallet$ can take values in $\mathbb{N}$. The output port Y, named $Command_output$, can take values in $\{ON, OFF\}$. The states of the model, denoted as S, consist of a phase and a command variable. The phase, named $Phase$, can have two possible values: “$Wait_measure$” and “$Command$”. The command variable, named $Command_D2$, can take values in $\{ON, OFF\}$. The external transition function, denoted as δ_{ext}, uses the $Meas_wallet$ variable to compute the new $Command_D2$ value and changes the phase from “$Wait_measure$” to “$Command$”. The internal transition function, denoted as δ_{int}, changes the phase from “$Command$” to “$Wait_measure$”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(Phase, Command_D2)$, sends the value of $Command_D2$ through the $Command_output$ port to the connected component. The phases are linked to specific time-advances, with “$Wait_measure$” linked to an infinite time-advance and “$Command$” linked to a time-advance of 0.

Acceptance A1 model The acceptance A1 model is triggered by the transfer of money following the consumption of energy (*Receipt*) of an associated load. It then updates the acceptance value. Its formal representation is provided in equation 3.34 and imaged with the figure 3.53 and a chronogram of its expected state behavior in figure 3.54.

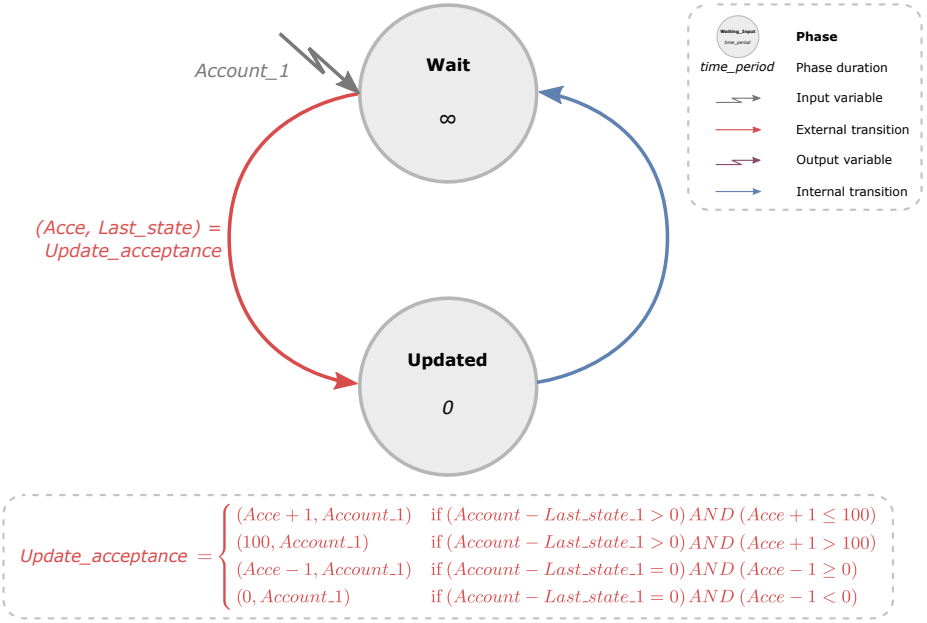


Figure 3.53: Graphical representation of the Acceptance A1 model states and dynamics

The proposed Acceptance model of the producer is composed of three variables: *Phase*, *Acce* and *Last_state*. The first variable, *Phase*, can take on two values: "*Wait*" or "*Updated*", and is used to model the system as a state machine. The second variable, *Acce*, is a numerical variable used for calculating the acceptance level of the producer. The third variable, *Last_state*, is a numerical variable used to store the last value of the producer wallet. The system is initially set to the "*Wait*" phase, which phase associated time is infinite, the *Acce* is set to 100 and the *Last_state* is set to 0. From the "*Wait*" state, the system waits for an input variable to be received in the input *Account_1*, triggering an external transition. Based on the difference between the value of the measured wallet amount received from the measure and its last amount, the acceptance will increase or decrease. This transition is performed by the *Update_acceptance* function to recompute the acceptance based on the measured value in input and updates the phase to "*Updated*". This triggers the internal transition which updates the phase to "*Wait*". This process repeats indefinitely.

The chronogram in figure 3.54 succinctly illustrates the expected behavior of the *Acce* and *Last_state* variables as a result of the account 1 value received in input.

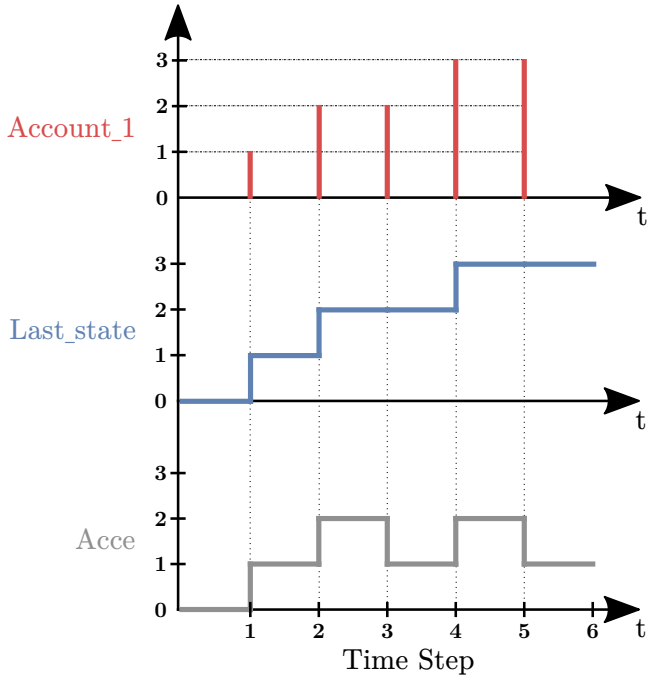


Figure 3.54: Chronogram of the behaviour of *Acce* and *Last_state* in the acceptance 1 model

The PDEVS mathematical representation of this model is given by equation 3.34.

$$PDEVS_{acceptance_1_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.34}$$

Where :

$X = \{(Account_1, v) \mid v \in \mathbb{N}\}$

$Y = \varnothing$

$S = \{(\boldsymbol{Phase}, Acce, Last_state) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait}\text{”}, \text{“}\boldsymbol{Updated}\text{”}\}, Acce \in [\![0; 100]\!], Last_state \in \mathbb{N}\}$

$\delta_{ext}\big((\text{“}\boldsymbol{Wait}\text{”}, Acce, Last_state), e, (Account_1, v)\big) = (\text{“}\boldsymbol{Updated}\text{”}, Update_acceptance)$

$$\text{With } Update_acceptance = \begin{cases} (Acce + 1, v) & \text{if } (v - Last_state_1 > 0)\ AND\ (Acce + 1 \leq 100) \\ (100, v) & \text{if } (v - Last_state_1 > 0)\ AND\ (Acce + 1 > 100) \\ (Acce - 1, v) & \text{if } (v - Last_state_1 = 0)\ AND\ (Acce - 1 \geq 0) \\ (0, v) & \text{if } (v - Last_state_1 = 0)\ AND\ (Acce - 1 < 0) \end{cases}$$

$\delta_{int}(\text{“}\boldsymbol{Updated}\text{”}, Acce) = (\text{“}\boldsymbol{Wait}\text{”}, Acce)$

$\lambda(\boldsymbol{Phase}, Acce) = \varnothing$

$ta(\text{“}\boldsymbol{Wait}\text{”}, Acce) = \infty$

$ta(\text{“}\boldsymbol{Updated}\text{”}, Acce) = 0$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.35. The initial state of an individual acceptance d, denoted s_{0d}, will set the acceptance's value to 100 and its input last state to 0. The initial phase is set to “$\boldsymbol{Wait}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait}\text{”}, 100, 0) \tag{3.35}$$

The input port X of the example model named *Account_1* can take values in $\mathbb{N}$. The model has no output port Y. The states of the model, denoted as S, consist of a phase, an acceptance variable and a wallet variable. The phase, named *Phase*, can have two possible values: “*Wait*” and “*Updated*”. The acceptance variable, named *Acce*, can take values in $\mathbb{N}$. The wallet variable, named *Last_state*, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the

$Account_1$ variable to compute the new $Acce$ value using the function $Update_acceptance$, it also replaces the value of $Last_state$ by $Account_1$ and changes the phase from "$Wait$" to "$Updated$". The internal transition function, denoted as δ_{int}, changes the phase from "$Updated$" to "$Wait$". In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. Since the system has no output port, it also does not have output functions. The phases are linked to specific time-advances, with "$Wait$" linked to an infinite time-advance and "$Updated$" linked to a time-advance of 0.

Acceptance A2 model The acceptance A2 model is triggered by the consumption of energy ($Receipt$) of an associated load. It then updates the acceptance value. Its formal representation is provided in equation 3.36 and imaged with the figure 3.55 and a chronogram of its expected state behavior in figure 3.56.

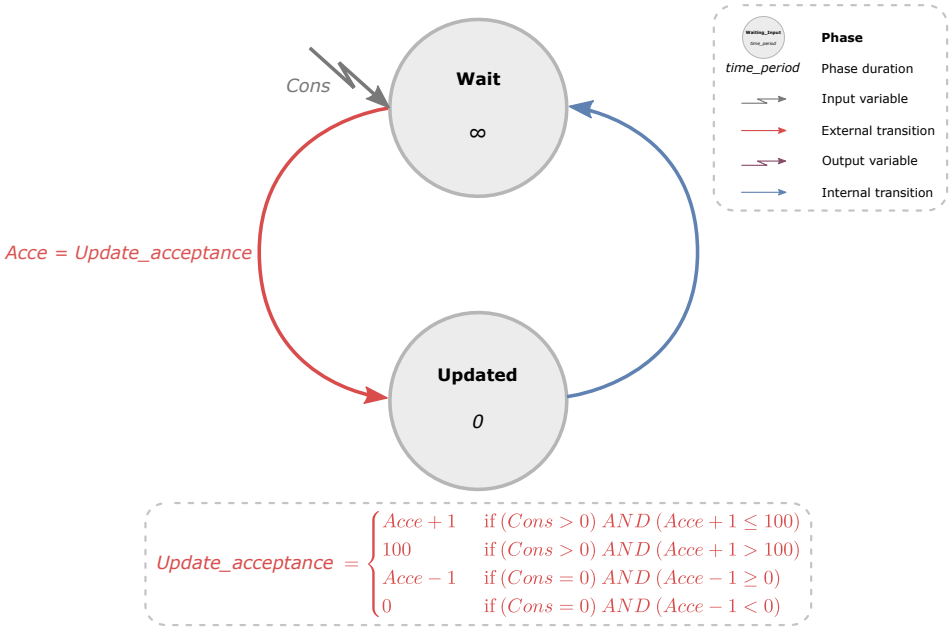


Figure 3.55: Graphical representation of the Acceptance A2 model states and dynamics

The proposed Acceptance model of the producer is composed of two variables: $Phase$ and $Acce$. The first variable, $Phase$, can take on two values: "$Wait$" or "$Updated$", and is used to model the system as a state machine. The second variable, $Acce$, is a numerical variable used for calculating the acceptance level of the producer. The system is initially set to the "$Wait$" phase, which phase associated time is infinite and the $Acce$ is set to 100. From the "$Wait$" state, the system waits for an input variable to be received in the input $Cons$, triggering an external transition. Based on the consumption value received from the load, the acceptance will increase or decrease. This transition is performed by the $Update_acceptance$ function to recompute the acceptance based on the consumption value in input and updates the phase to "$Updated$". This triggers the internal transition which updates the phase to "$Wait$". This process repeats indefinitely.

The chronogram in figure 3.56 succinctly illustrates the expected behavior of the $Acce$ variables as a result of the consumption received in input.

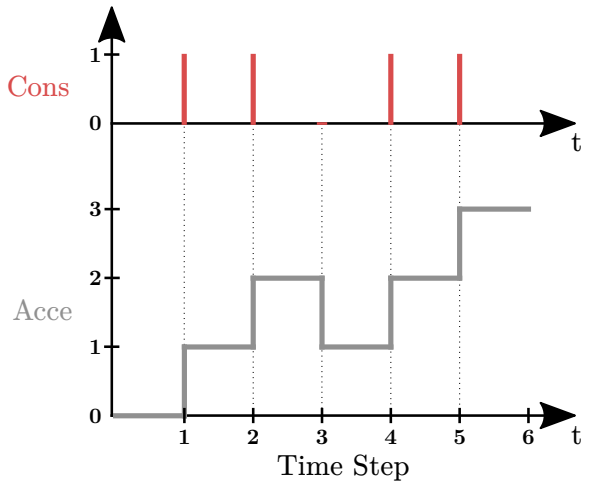


Figure 3.56: Chronogram of the behaviour of *Acce* in the acceptance 2 model

The PDEVS mathematical representation of this model is given by equation 3.34.

$$PDEVS_{acceptance_2_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.36}$$

Where :

$$X = \{(Cons, v) \mid v \in \mathbb{N}\}$$
$$Y = \varnothing$$
$$S = \{(\boldsymbol{Phase}, Acce) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Wait}\text{”}, \text{“}\boldsymbol{Updated}\text{”}\}, Acce \in [\![0; 100]\!]\}$$
$$\delta_{ext}\big((\text{“}\boldsymbol{Wait}\text{”}, Acce), e, (Cons, v)\big) = (\text{“}\boldsymbol{Updated}\text{”}, Update_acceptance)$$
$$\text{With } Update_acceptance = \begin{cases} Acce + 1 & \text{if } (v > 0)\ AND\ (Acce + 1 \leq 100) \\ 100 & \text{if } (v > 0)\ AND\ (Acce + 1 > 100) \\ Acce - 1 & \text{if } (v = 0)\ AND\ (Acce - 1 \geq 0) \\ 0 & \text{if } (v = 0)\ AND\ (Acce - 1 < 0) \end{cases}$$
$$\delta_{int}(\text{“}\boldsymbol{Updated}\text{”}, Acce) = (\text{“}\boldsymbol{Wait}\text{”}, Acce)$$
$$\lambda(\boldsymbol{Phase}, Acce) = \varnothing$$
$$ta(\text{“}\boldsymbol{Wait}\text{”}, Acce) = \infty$$
$$ta(\text{“}\boldsymbol{Updated}\text{”}, Acce) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 3.37. The initial state of an individual acceptance d, denoted s_{0d}, will set the acceptance's value to 100. The initial phase is set to “$\boldsymbol{Wait}$”.

$$s_{0d} = (\text{“}\boldsymbol{Wait}\text{”}, 100) \tag{3.37}$$

The input port X of the example model named *Cons* can take values in $\mathbb{N}$. The model has no output port Y. The states of the model, denoted as S, consist of a phase and an acceptance variable. The phase, named *Phase*, can have two possible values: “*Wait*” and “*Updated*”. The acceptance variable, named *Acce*, can take values in $\mathbb{N}$. The external transition function, denoted as δ_{ext}, uses the *Cons* variable to compute the new *Acce* value using the function $Update_acceptance$ and changes the phase from “*Wait*” to “*Updated*”. The internal transition function, denoted as δ_{int}, changes the phase from “*Updated*” to “*Wait*”. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. Since the system has no output port, it also does not have output functions. The phases are linked to specific time-advances, with “*Wait*” linked to an infinite time-advance and “*Updated*” linked to a time-advance of 0.

Final coupled model

The root model, or final coupled model, of our component-based modeling of a microgrid in PDEVS allows us to connect all of the previously presented atomic models of the different components of a microgrid. This coupled model represents the completion of our structural generative modeling, which integrates the microgrid's fields of energy, information, finance, and social With a deeper level of detail for a more comprehensive understanding of its operation. The root model serves as the central hub for the microgrid model, allowing for the simulation of interactions between the atomic models. This can provide insight into the microgrid's operation and the interconnections between its various fields. The coupled model presented here completes our component-based modeling of a microgrid. By connecting all atomic fields component models, we can comprehensively understand and model the microgrid's operation and the relationships between its various fields.

$$PDEVS_{root_model} = (X, Y, D, \{M_d\}, EIC, EOC, IC) \tag{3.38}$$

Where :

$X = \{\}$

$Y = \{\}$

$D = \{Source_1,\ Load_1,\ Battery_1,\ Converter_1,\ Wallet_1,\ Wallet_2,\ Bank_1,\ Measure_1,\ Measure_2,\ Device_1,$
$Device_2,\ Acceptance_1,\ Acceptance_2\}$

$\{M_d\} = \{PDEVS_{source_1_model}, PDEVS_{load_1_model}, PDEVS_{battery_1_model}, PDEVS_{converter_1_model},$
$PDEVS_{wallet_1_model}, PDEVS_{wallet_2_model}, PDEVS_{bank_1_model}, PDEVS_{measure_1_model},$
$PDEVS_{measure_2_model}, PDEVS_{device_1_model}, PDEVS_{device_2_model}, PDEVS_{acceptance_1_model},$
$PDEVS_{acceptance_1_model}\}$

$EIC = \{\}$

$EOC = \{\}$

$IC = \{\big((Source_1,"Production_output"),(Battery_1,"prod")\big)\ ,\ \big((Source_1,"Production_output"),(Converter_1,"prod")\big),$
$\big((Source_1,"Production_output"),(Battery_1,"prod")\big)\ ,\ \big((Converter_1,"Command_cons_output"),(Load_1,"ctrl_cons")\big),$
$\big((Converter_1,"Command_cons_output"),(Battery_1,"ctrl_cons")\big)\ ,\ \big((Battery_1,"Capacity_output"),(Converter_1,"capa")\big),$
$\big((Load_1,"Consumption_output"),(Measure_1,"cons")\big)\ ,\ \big((Load_1,"Consumption_output"),(Acceptance_1,"cons")\big),$
$\big((Measure_1,"Meas_cons_output"),(Device_1,"Meas_cons")\big)\ ,\ \big((Device_1,"Command_bill_output"),(Bank_1,"Command_bill")\big),$
$\big((Bank_1,"Bill_output"),(Wallet_2,"Bill")\big)\ ,\ \big((Wallet_2,"Payment_output"),(Bank_1,"Payment")\big),$
$\big((Wallet_2,"Account_state_output"),(Measure_2,"Account_state")\big),$
$\big((Measure_2,"Meas_wallet_output"),(Device_2,"Meas_wallet")\big),$
$\big((Bank_2,"Payment_output"),(Wallet_1,"Pay")\big)\ ,\ \big((Wallet_1,"Account_state_output"),(Acceptance_1,"Account_1")\big),$
$\big((Device_2,"Command_output"),(Converter_1,"Command_D2")\big)\}$

The coupling of this proposed structural model is given in the form of a sequence diagram in figure 3.57 representing the execution sequence of all atomic components during an execution step. This sequential diagram works as follows; are represented the four field flows models and the twelve exchange models between them. Each block will represent the execution of an internal or external transition function that can also update the phase of its model, linked to a specific time advance function. The color of the dotted line of each model represents a specific ta, a red dotted line represents a $ta = 0$, a blue doted line represents a $ta = 1$ and a black dotted line represents a $ta = \infty$. The arrows in the sequential model represent an output function that sends a message to another model input, initiating an external function on the latter. In this coupled model, each atomic component will be triggered consecutively depending on the inputs they recieve and the ta changes that it initiate. This representation can be seen as an equivalent of the equation above, but it brings another perspective for understanding this overall model operation and behavior.

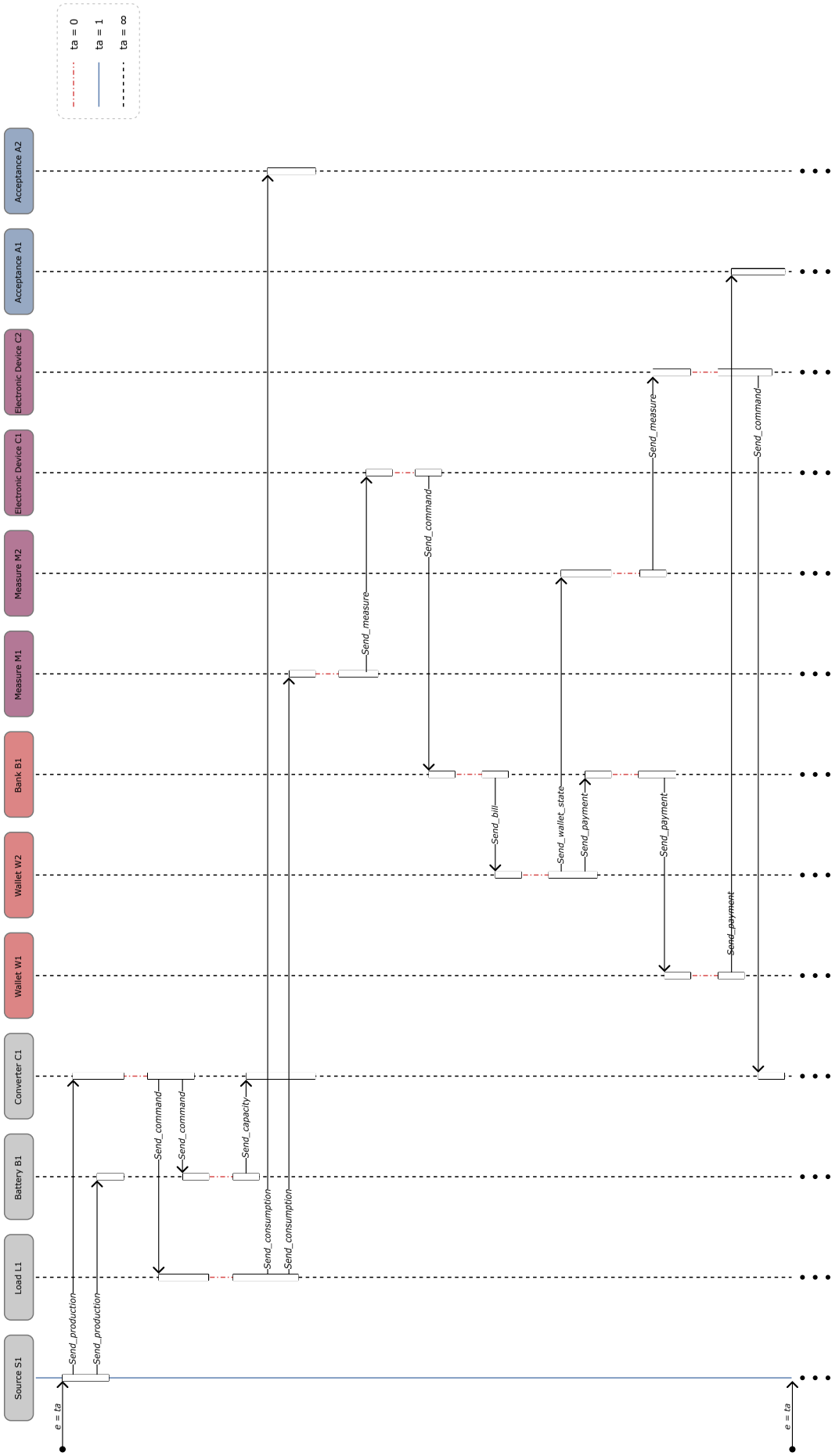


Figure 3.57: The sequential diagram of the full generative model

Model Summary and Overview

The proposed generative model is a more detailed and explicit approach to understanding and evaluating microgrids as compared to the structural model. The structural model provides a systemic view of the microgrid and considers the interconnectedness of the different components, while the generative model takes this a step further by simulating the individual components of each field in more detail. This allows for a deeper understanding of the underlying mechanisms and dynamics of the microgrid, and can aid researchers in identifying potential areas for improvement or optimization.

The generative model has a number of advantages, including its ability to simulate a microgrid with both energy and community elements. By using analogies from electric components such as load, source, battery, and converter, the generative model can represent the flows and exchanges between the different components of a microgrid. Furthermore, the model considers the social field to be composed of storage and decision, allowing for the incorporation of community involvement and its impact on the sustainability of the microgrid. This is a key advantage over the structural model, which may not provide as much detail on the social aspect of a microgrid.

Despite its advantages, the generative model also has some limitations that need to be considered. One such limitation is its difficult expansion due to its coupling nature, which requires a time-consuming operation of defining the coupling between elements on an element-by-element basis. This lack of architectural flexibility makes it challenging to model all fields of the microgrid, especially when dealing with a large number of elements. Additionally, the model may be more difficult to interpret and understand for those unfamiliar with electric components and their analogies, which can be a challenge for non-expert users. Another limitation of the generative model is that it requires a significant amount of data to perform the simulation, which can be a challenge when the data is not available or is incomplete. Furthermore, the generative model requires a high level of computational resources and time to perform the simulations, which can be a limitation when working with large-scale microgrids or when performing multiple simulations.

In conclusion, the proposed generative model offers a valuable tool for research on the sustainability of microgrids, providing a deeper understanding of the inner workings and dynamics of a microgrid. Its ability to simulate the components of each field in more detail can support the development of more effective strategies and interventions for enhancing the sustainability of microgrids. However, it's important to be aware of its limitations such as its difficult expansion, lack of interpretability for non-expert users, high data requirements, computational cost and lack of flexibility. These limitations should be considered when selecting a model for microgrid research, and the unique characteristics and requirements of each microgrid should be taken into account.

3.2.3 Replicative model: Assembling I/O functions and states into a co-simulation

The replicative model has the objective of generating low-level data from I/O functions deduced from the results of experimental frames. The different fields of the systemic microgrid model proposed in the previous chapter require different types of I/O functions to be properly simulated. The energy field is mostly based on differential equations, the information and financial fields use discrete time approaches while the social field is best suited for discrete event models.

From a Modeling and Simulation theory perspective, this can be implemented by using DEVS as a wrapper for enabling co-simulation between existing solutions. This is the approach of the proposed replicative model.

PDEVS Co-simulation: Bridging the Gap Between Fields

A co-simulation gathers models with internal components which are simulated with their own simulators and whose results are coupled through a series of provided interactions schemes. Thus, each field will have its own simulator which abstracts away internal, external, time and output functions providing the output which is then sent from one simulator to the other.

The major advantage of the co-simulation is the re-use of other models, ideally who are based on real I/O data. The major issue of the co-simulation is to ensure the good interconnection and synchronization of the different simulators involved. This replicative model proposes a DEVS-based wrapper to solve this issue and go one step down in the system specification latter.

By applying the co-simulation idea to the microgrid model developed in this work gives figure 3.58.

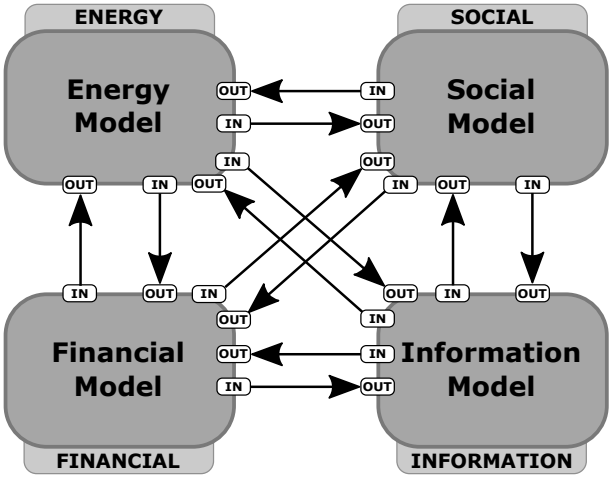


Figure 3.58: Desired co-simulation systemic microgrid model using the four main fields

The co-simulaiton model in 3.58 has the same four fields of previous models: Energy, Financial, Information, and Social. Each field has its own distinct model, and all of the models are connected together through inputs and outputs. The input/output ports and the arrows represent the DEVS wrapper that synchronizes all the different models together.

Proposed replicative model: Validating model interconnection

This section will give an overview of the propose replicative model shown in figure 4.20. The proposed replicative model includes three fields: Energy, Financial and Social. Each is represented by its respective model and the variables that interconnect each model.

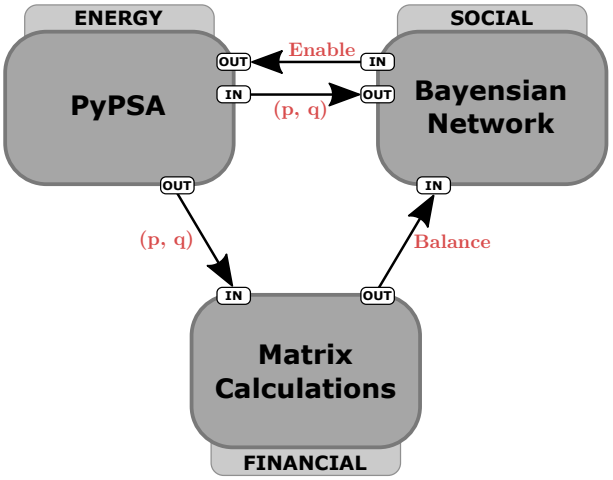


Figure 3.59: Modeled co-simulation systemic microgrid model during this work

The information field was suppressed in this model. As defined in chapter 2, communication is composed of data and control. Their issues were described as control issues linked to the operation of the energy field and data issues linked with circulating information among different fields. In this model, control is considered to be perfect, as the energy field simulation will alwasy converge to an answer. Data is represented by the links between the different fields and is considered flawless. These justify the not explicitly drawing the information system in this model.

The *Energy* field is modelled by PyPSA, a python based power flow simulator which will be explained in detail below. This simulator takes as inputs usage from the social field. This means that an user can decide not to activate a load, and this is taken into account into the power flow calculations. Its outputs are the power flow results, which are used by the financial and the social fields.

The *Financial* field is modeled by a cashflow system that represents the flows of money between different elements of the system. Its input is the power flow result, which will be used as a basis to calculate what each member needs to pay or receive. Its output is the account balance of the members, which will be used by the social field.

The *Social* field modeled by a Technology Acceptance Model that uses bayesian inference as a means to translates the user perception and experience into acceptance. It receives the power flow results and the account balance as inputs and uses them to integrate the availability of energy and the payment experience to update the user's acceptance of the technology. Its output is the enable variable, that translates if the user uses or not the technology, which means activating his associated load.

The objective of this model is to explore the interconnection between these very different models and validate that this can be a basis for studies of the microgrid sustainability.

The next sections will give more detail of each component and their respective wrapper.

Energy Model: The PyPSA package

There are many energy flow simulators, each with its advantages and disadvantages for modeling microgrids. To justify our PyPSA choice, this section will detail the benchmark and how it took into account the constraints of our systemic modeling approach.

The first constraint of this work was the need to use Python. As will be explained in section 4.1.4 page 122, PyPDEVS was chosen to write the wrapper, as it is the only available PDEVS coding framework with appropriate documentation. Thus, the first criteria for shortlisting power flow simulators was Python compatibility.

Our second constraint was that this simulator had to integrate a certain number of functionalities to model all the diversity of the energy domain, which was identified during the literature review in chapter 2. Five criteria were established in order to validate this constraint:

- The simulator must be able to integrate a large number of different electrical components allowing it to represent the diversity of the electrical elements

- It must also be able to integrate all the forms of electrical architecture described in microgrid literature
- The modeling of the quality or the aging of the microgrid must also be able to be modeled
- The simulator must allow bringing a certain number of metrics related to the dimensioning of the grid
- It must allow the possibility of modeling the elements of protection

Our third constraint is that the power flow simulator must allow a fluid and comfortable integration within the co-simulation model. In practice, this means that the class structure of the code must be easily adapted to the PyPDEVS class structure, with a minimum of code rework.

Four python solutions were identified for modeling and simulation of power flow within an electrical grid, whose different functionalities are represented in table 3.7.

Table 3.7: Open source python based power flow simulator review adapted from [8]

Comparison Criteria	**PyPSA**	**oemof**	**GridCal**	**pandapower**
multi-objective optimization	yes and manual	N/A	no	no
system sizing	yes	yes	no	manual
operational strategies/ UC	yes	yes	no	no
Sector coupling	yes	yes	no	yes
Time resolution	flexible	flexible	flexible	flexible
Optimization Constraints:				
Power Balance	yes	yes	no	yes
Linear Optimal Power Flow	yes	no	no	yes
Nonlinear Optimal Power Flow	No (planned)	no	no	yes
Multi-period Optimization	yes	yes	no	yes
Bus components:				
Variable Renewable Energy Sources	yes	yes	yes	yes
Inverters	yes	yes	yes	yes
Battery Storage	yes	yes	yes	yes
Diesel generators	yes	yes	yes	yes
Biomass	yes	yes	no	no
Hydro turbines	yes	yes	no	no
Thermal storage	yes	yes	no	no
Fuel cell	yes	-	no	no
Renewable feed-in data	partially	yes	no	no
Grid Analysis:				
DC power flow	yes	yes	yes	yes
AC power flow	yes	no	yes	yes
Short-circuit analysis	no (planned)	no	yes	yes
Systemic:				
Element diversity	Good	Good	Limited	Limited
Architecture diversity	Good	Limited	Good	Good
Quality modeling	Possible	Possible	Limited	Possible
Sizing metrics	Possible	Possible	No	Difficult
Protection modeling	Possible	Possible	Possible	Possible

This benchmark shows PyPSA as the most appropriate tool for our study object. Indeed, it easily allows the representation of a large number of electrical elements and architecture by bringing many details and characteristics to represent and model the various problems of an electrical grid while proposing an object-oriented implementation that is extremely easy to implement and use.

Now that the choice is done, some details of PyPSA are given below to explain its usage in the proposed replicative model.

PyPSA (Python for Power System Analysis) is an open-source library that offers a modular and object-oriented approach to modeling power systems, making it easy to represent and simulate interactions between microgrid components [253]. PyPSA's graph-theoretic approach and energy balance equations enable comprehensive and accurate power flow modeling. Additionally, its time series feature allows for the simulation of power flow over time, making it ideal for integrating with the other fields of the a co-simulation model.

The main components of the PyPSA power flow model include buses, generators, loads, and lines as shown in figure 3.60.

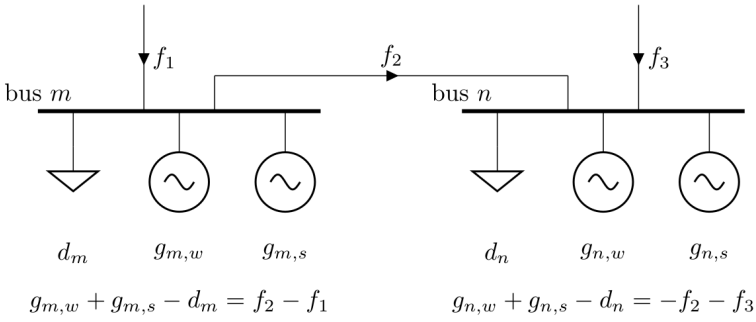


Figure 3.60: PyPSA bus model

The buses represent nodes in the power system, and the generators, loads, and lines represent the connections between the buses. The bus is the fundamental node to which all loads, generators, storage units, lines, transformers and links attach. Each bus enforces energy conservation for all elements feeding in and out of it. It is possible to attach as many components to a bus as needed. The active and reactive power flows on the lines are expressed as linear equations in terms of the bus voltage and angles.

In our study, we modeled the loads as energy consumption, sources as energy generation, batteries as energy storage, and buses as nodes in the power system, using the corresponding flow equations provided by PyPSA. Each bus node is linked to a unique individual, making its power flow calculations compatible with the cashflow and social acceptance models.

Financial Model: Creating a cashflow system

A cash flow is the net amount of cash and cash equivalents being transferred in and out of money reserve. This money reserve can represent the pocket of a person or the bank account of a business. In the proposed replicative model, cash flow modeling is the process of simulating the financial behavior of the bank accounts of the individuals that are related to the microgrid over time. This cashflow takes into account the factors such as energy consumption and production, costs, and revenue of the participants.

As we noted in the literature review in Chapter 2, the elements of the financial domain can be separated into four groups: consumers, producers, maintainers, and prosumers. As described in figure 3.61, each of these groups will have different financial behaviors, with different categories and types of income and different categories and types of expenditure. Furthermore, the financial flows between these groups may vary depending on the type of energy and financial architectures of the microgrid. Our cashflow model must then be capable of accommodating all of these different connections between individuals belonging to differnet groups and relate the inputs/output of the cashflow with the other fields of the co-simulation model.

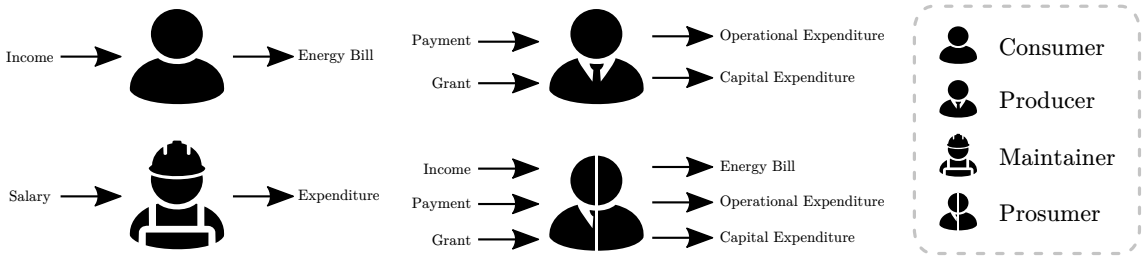


Figure 3.61: Representation of the diversity of agent within the microgrid and their modeling carateristics

In order to address this constraints, we built an object-oriented cash flow model for energy microgrids.

Our object oriented model is based on a single class of economic agent. Each agent of the system is declared with an initial balance, an average income, an average external cost and an energy price as depicted in figure 3.62 Our object oriented model for energy microgrids is designed to take into account the consumption and production of each user of the microgrid, and to calculate how much each user owes and to whom. The model takes into account various inputs, including the active and reactive power prices, which are used to calculate the cost of energy for each user. It also considers external costs and income for each user, as well as a variation function for these costs and income.

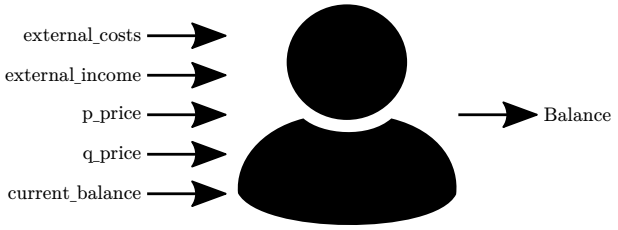


Figure 3.62: Representation of the generic agent in the financial model

These inputs are used to calculate each user's balance and to distribute payments among the microgrid community. The model also uses a random generator to simulate variations in external costs and income for each user, which makes the model more realistic. The equation used to recalculate the balance of each user is formalized in equation 3.39.

$$Balance = current_balance + external_income - external_costs + (p_price \cdot p) + (q_price \cdot q) \quad (3.39)$$

The balance of each individual is sent to the social model. This object-oriented approach gives the model the necessary flexibility to create members of the micro-grid who will differ according to their initialized economic variables. This single class expresses all the types of agents of the microgrid. The architecture aspects of the financial field are not represented in this model.

Social Model: the Technology Acceptance Model

In modeling social acceptance, there are two challenges: quantifying a user's acceptance based on different factors perceived by the user of their environment and finding a way to express the effect of time, and the user's lived experience within the microgrid. For this model, we consider that the acceptance calculation is based on the different events in the microgrid. Therefore, an agent will develop its opinion over time based on its user experience.

TAM is a widely-used model that focuses on the individual user's perception of a technology. Specifically, it posits that users' acceptance of a technology is determined by their perceptions of the technology's usefulness (PU) and ease of use (PEOU). These perceptions are influenced by external factors, such as the availability and affordability of the technology, as well as the user's

prior experience with similar technologies. TAM has been used in a variety of research contexts, including the study of community microgrids.

In our model, we have chosen to use TAM because it is a well-established and widely-used model in the field of community microgrids, and it allows us to focus on the individual user's perception of the technology. By using TAM, we are able to take different factor inherent to the microgrid system into account and model their effect on the user's acceptance of the technology.

We chose to use Bayesian networks as a way to represent TAM. Bayesian networks are a type of probabilistic graphical model that can represent complex relationships between variables and can handle uncertainty. Using Bayesian networks, we are able to model the relationships between perceived ease of use, perceived usefulness, and other variables such as the availability and affordability of electricity, as well as the user's experience level.

Thus, thanks to this model, we can solve these two difficulties. Indeed, the Technology Acceptance and Usage Model (TAM) allows us to quantify the acceptance of an agent according to many input parameters in a relatively simple way. Furthermore, the question of experience is partially solved thanks to the very concept of Bayesian networks, which allow us to integrate the effect of time spent through probability equations.

A simple example of a Bayesian network can be a network with two variables: Rain and Umbrella. Rain is the cause and Umbrella is the effect. Rain can have two states: Rainy (R) and Not Rainy(NR), and Umbrella can have two states: Taken (T) and Not Taken (NT). The conditional probability of Umbrella given Rain can be represented by the following table:

Where $P(U = T|R = R)$ represents the probability of taking an umbrella given that it is raining, and $P(U = NT|R = NR)$ represents the probability of not taking an umbrella given that it is not raining. The corresponding Bayesian network is illustrated in figure 3.63.

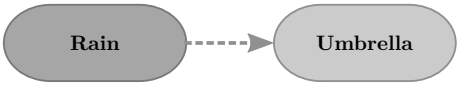


Figure 3.63: Induced Bayesian network of the example

By using the Bayesian network, we can make predictions about the Umbrella taking decision based on the weather condition. For example, if it is raining, the network can calculate the probability that an individual will take an umbrella by using the conditional probability of Umbrella given Rain $P(U = T|R = R)$ and the prior probability of Rain $P(R = R)$ which is obtained from historical data. The calculation is done using the following equation 3.40:

$$P(U = T|R = R) = \frac{P(R = R|U = T) \cdot P(U = T)}{P(R = R)} \tag{3.40}$$

This equation is called the Bayes' theorem, it is used to calculate the probability of an event A given that an event B has occurred. In this case, the event A is taking an umbrella and event B is it is raining.

In our model, we used a Bayesian network to model the relationships between the availability and affordability of energy, the experience of the user, the perceived usefulness, the perceived ease of use, the intention to use and finally the actual use depicted in figure 3.64.

The Bayesian network that has been presented is implemented for each network member. In this social model, each agent will be associated with a Boolean value representing its current use or non-use of the micro-grid, calculated using the Bayesian network-induced probability for that agent.

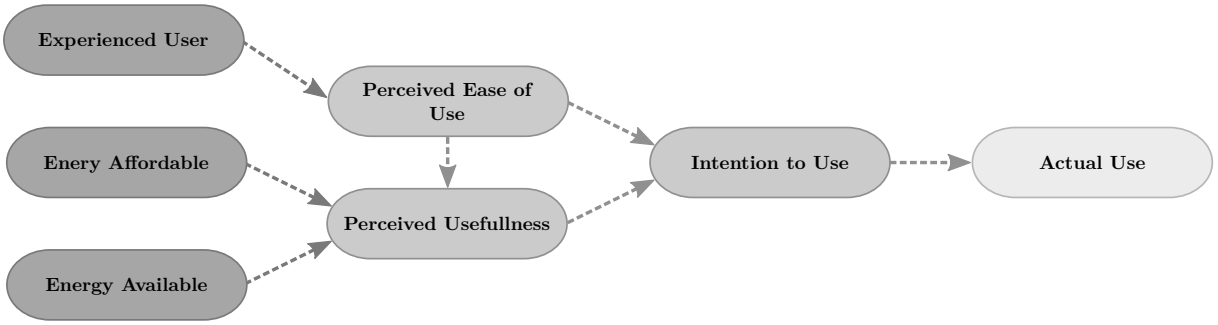


Figure 3.64: Bayesian network inspired from the TAM model

To describe a bit more the model used, the availability of electricity is connected to the perceived usefulness of the microgrid, and the affordability of electricity is connected to the perceived ease of use of the microgrid. The experience of the user variable is connected to the perceived ease of use of the microgrid. These connections are represented by conditional probability distributions associated with each edge of the Bayesian network.

The implementation of the Bayesian network have been made through Python with help of pomegranate library that allow to create dependency network between several variables. This implementation need several conditional probability tables in order to link all the dependent probabilities together and predict the actual use of the agent. One of the main challenges in developing this model was the lack of data to use to train the model.

If we look at our previous example concerning rain and umbrella, we can describe the discrete distribution of the probability of rain, as presented in Table 3.8, as well as the Bayesian network probability of the price of the umbrella being taken or not, as described in Table 3.9. In this case, the probability of rain is high, while the probability of taking an umbrella is higher when it is raining and, conversely, lower when it is not raining.

Table 3.8: Discrete distribution of the rain probability

	T	**F**
Rain	0.8	0.2

Table 3.9: Probability of the Bayesian network of the umbrella taken or not

Rain	**T**	**F**
F	0.2	0.8
T	0.8	0.2

To overcome this challenge, we used an empirical approach to determine the discrete distribution of the Bayesian network. We tried different simulations and chose the distribution that seemed to be the most balanced presented in table 3.10, table 3.11, table 3.12 and the conditional probabilities that resulted from it table 3.13, table 3.14, table 3.15 and table 3.16.

This initial estimation is designed to provide a simple but still close-to-reality model for understanding user acceptance of community microgrids and its repercussions on the system. The Bayesian network in combination with the TAM allows us to incorporate additional variables and dependencies into the model. For example, in future research, we could include additional variables such as the average social acceptance of the community of the microgrid or the level of trust in the microgrid operator to improve the accuracy of the predictions made by the model.

Table 3.10: Discrete distribution of the availability of the energy

	T	**F**
Energy Available	0.8	0.2

Table 3.11: Discrete distribution of the affordability of the energy

	T	**F**
Energy Affordable	0.8	0.2

Table 3.12: Discrete distribution of the agent's experience level

	T	**F**
Experienced User	0.2	0.8

Table 3.13: Probability of the Bayesian network of the agent's perceived ease of use

Experienced User	**T**	**F**
F	0.2	0.8
T	0.8	0.2

Table 3.14: Probability of the Bayesian network of the agent's perceived usefulness

Energy Available	**Energy Affordable**	**Perceived Ease of Use**	**T**	**F**
F	F	F	0	1
F	F	T	0.2	0.8
F	T	F	0.3	0.7
F	T	T	0.2	0.8
T	F	F	0.5	0.5
T	F	T	0.7	0.3
T	T	F	0.8	0.2
T	T	T	1	0

Table 3.15: Probability of the Bayesian network of the agent's intention to use

Perceived Usefulness	**Perceived Ease of Use**	**T**	**F**
F	F	0	1
F	T	0.7	0.3
T	F	0.6	0.4
T	T	1	0

Table 3.16: Bayesian network probability of the agent's actual use

Intention to Use	**T**	**F**
T	0,8	0.2
F	0.2	0.8

Information Model: Coupling information flows between models

To address the complexity of modeling the information flow in our co-simulation model, we have made the decision to simplify the information modeling by making it perfect. This means

that the control part of the electrical grid is directly integrated into the electrical model, and the information flows between the components are instantaneous and transparent. All components have the information they need on the whole grid to make their calculations. This approach allows for a more straightforward and efficient co-simulation process, but it does have some limitations. For example, it does not account for the specific characteristics of information flows, such as delays or communication errors. However, this simplified approach is a practical solution for the first iteration of our co-simulation model and provides a strong foundation for future research and development in this area. The goal is to eventually properly integrate an information model within the co-simulation of energy microgrids, but that requires a deeper understanding of the information flows and their specific characteristics.

One of the key challenges in microgrid modeling is the integration of information flows between different components. In our model, the control strategy of the electrical grid is primarily based on the connection and disconnection of one bus due to users' levels of acceptance of the technology. Specifically, if a user's "Actual Use" in their technology acceptance model drops to zero, the user will disconnect from the microgrid and stop using it. The rest of the control is primarily focused on managing this change in the number of electrical nodes and redirecting electrical flows.

PDEVS Model: Coupling the Different Fields on One Unified Formalism

Energy Model

The energy model allows the PyPSA library to be encapsulated so that the results of the power flow calculation can be retrieved and passed on to other models during the co-simulation. Its formal representation is provided in equation 3.41 and imaged with the figure 3.65

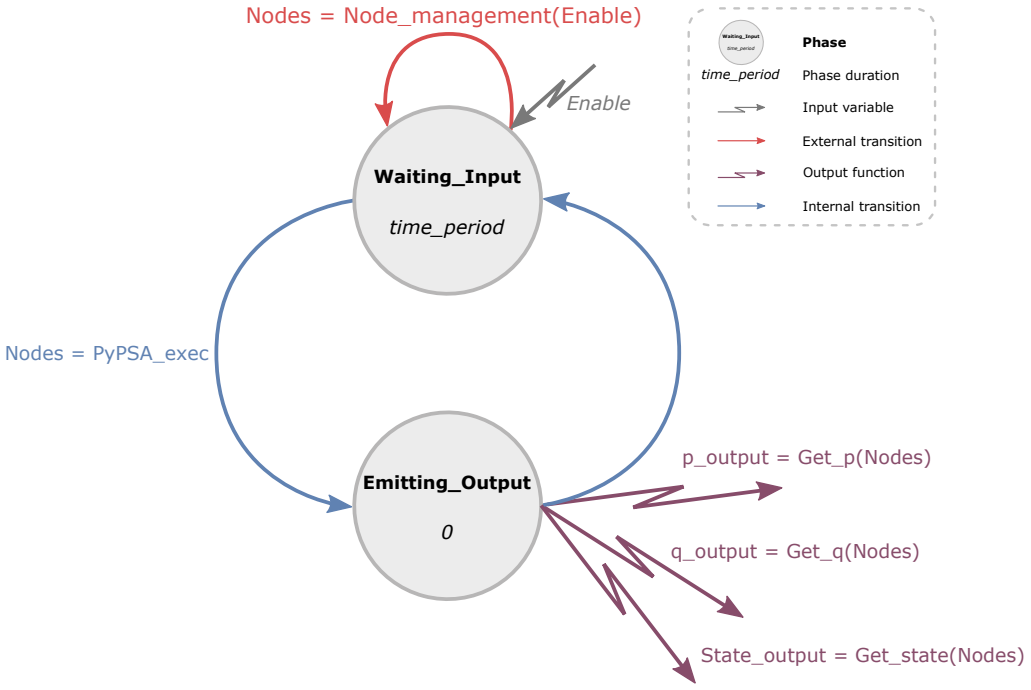


Figure 3.65: Graphical representation of the energy model states.

The proposed Energy model is composed of two main variables: *Phase* and *Nodes*. The first variable, *Phase*, can take on two values: "*Waiting_input*" or "*Emitting_output*", and is used to model the system as a state machine. The second variable, *Nodes*, is a numerical variable used to store all the data related to the energy grid. The system is initially set to the "*Waiting_input*" phase, which phase associated time is *time_period* which will be set to 1 in our case and the *Node* is initialised to a determined value by the user. From the "*Waiting_input*" state, the system waits for an input variable to be received in the input *enable*, triggering an external transition. Based on the acceptance state of individuals, the node state will will be activated or disabled.

After this transition, the elapsed time will eventually trigger an internal transition which uses the $PyPSA_exec$ to update the state of each electrical nodes with the help of PyPSA and updates the phase to "$Emitting_output$". This triggers the output function, which outputs the value of p in p_output output, q in q_output output. After the output function is complete, the internal transition is executed, which updates the phase to "$Wait_measure$" and $State$ in $State_output$. This process repeats indefinitely. As this system is much more complex in its internal states, we will not present a state chronogram to image its behaviour. The mathematical representation of this model composed of a number N_{nodes} electrical node in the microgrid is given by the equation 3.41.

$$PDEVS_{Energy_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.41}$$

Where :

$$X = \{(enable, v) \cup v \in \{Connected, Disconnected\}^{N_{nodes}}\}$$

$$Y = \{(p_output, v) \cup v \in \mathbb{R}^{N_{nodes}}\} \mid \{(q_output, v) \cup v \in \mathbb{R}^{N_{nodes}}\} \cup \{(State_output, v) \mid v \in \{Connected, Disconnected\}^{N_{nodes}}\}$$

$$S = \{(\boldsymbol{Phase}, Nodes) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Waiting_input}\text{”}, \text{“}\boldsymbol{Emitting_output}\text{”}\}, Nodes \in \mathcal{N}\}$$

With $\mathcal{N}$ the set of possible states for a node.

$$\delta_{ext}\big((\text{“}\boldsymbol{Waiting_input}\text{”}, Nodes), e, (enable, v)\big) = (\text{“}\boldsymbol{Emitting_output}\text{”}, Node_management(v))$$

$$\delta_{int}(\text{“}\boldsymbol{Waiting_input}\text{”}, Nodes) = (\text{“}\boldsymbol{Emitting_output}\text{”}, PyPSA_exec(Nodes))$$

$$\delta_{int}(\text{“}\boldsymbol{Emitting_output}\text{”}, Nodes) = (\text{“}\boldsymbol{Waiting_input}\text{”}, Nodes)$$

$$\lambda(\text{“}\boldsymbol{Emitting_output}\text{”}, Nodes) = \big(p_output, Get_p(Nodes)\big), \big(q_output, Get_q(Nodes)\big), \big(State_output, Get_state(Nodes)\big) \quad brown2017$$

$$ta(\text{“}\boldsymbol{Waiting_input}\text{”}, Nodes) = 1$$

$$ta(\text{“}\boldsymbol{Emitting_output}\text{”}, Nodes) = 0$$

The starting point for the model, also known as the initialization, is determined using the process outlined in equation 3.42.

$$s_{0d} = (\text{“}\boldsymbol{Emitting_output}\text{”}, Initial_nodes) \tag{3.42}$$

The input port X of the energy model named $enable$ can take values in $\{Connected, Disconnected\}^{N_{nodes}}$ with N_{nodes} the number of nodes in the system. The output ports Y are named p can take values in $\mathbb{R}^{N_{nodes}}$, q can take values in $\mathbb{R}^{N_{nodes}}$ and $State$ can take values in $\{Connected, Disconnected\}^{N_{nodes}}$. The states of the model, denoted as S, consist of a phase and a node variable. The phase, named $Phase$, can have two possible values: **"Waiting_input"** and **"Emitting_output"**. The node variable, named $Nodes$, can take values in $S^{N_{nodes} \times N_{variables}}$. The external transition function, denoted as δ_{ext}, uses the $enable$ variable to calculate the new $Nodes$ value using the abstract function $Changing_nodes_state$ and keep the phase unchanged. Applying δ_{int} when the phase is **"Waiting_input"**, recalculate the $Nodes$ value using the function $PyPSA_exec$ and changes the phase from **"Waiting_input"** to **"Emitting_output"**. Applying δ_{int} when the phase is **"Emitting_output"**, changes the phase from **"Emitting_output"** to **"Waiting_input"**. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(\text{“}\boldsymbol{Emitting_output}\text{”}, Nodes)$, sends the value of $Get_p(Nodes)$ through the p_output port, the value of $Get_q(Nodes)$ through the q_output port and the value of $Get_State(Nodes)$ through the $State_output$ port to the connected component. The phases are linked to specific time-advances, with **"Waiting_input"** linked to an infinite time-advance and **"Emitting_output"** linked to a time-advance of 0.

Financial Model

The financial model allows matrix calculation based cash flow to be encapsulated so that the results of the financial exchanges can be retrieved and passed on to other models during the co-simulation. Its formal representation is provided in equation 3.43 and imaged with the figure 3.66

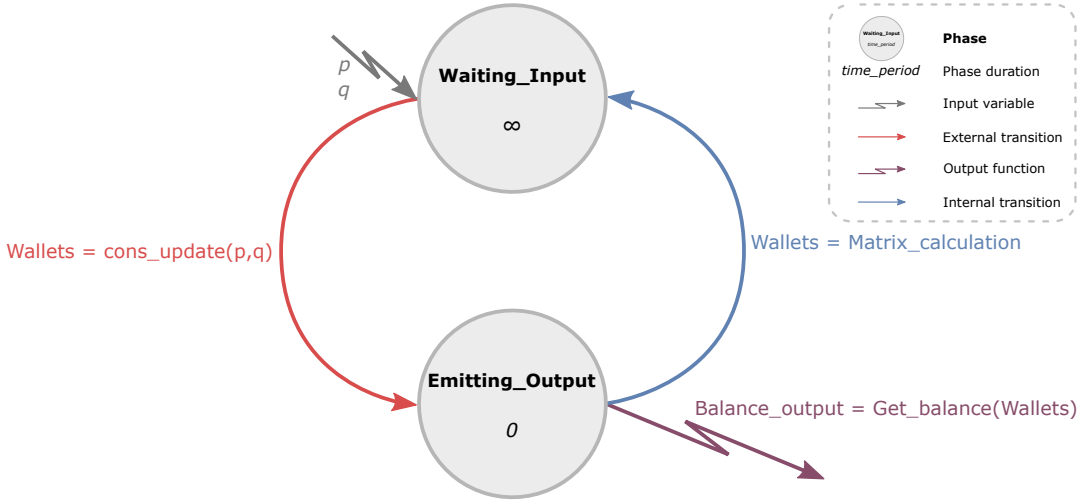


Figure 3.66: Graphical representation of the social model states.

The proposed financial model is composed of two main variables: $Phase$ and $Wallets$. The first variable, $Phase$, can take on two values: "$Waiting_input$" or "$Emitting_output$", and is used to model the system as a state machine. The second variable, $Wallet$, is a numerical variable used to store all the data related to the financial values. The system is initially set to the "$Waiting_input$" phase, i.e. the ta value is set to ∞ and the $Wallet$ is initialised to a determined value by the user. From the "$Waiting_input$" state, the system waits for an input variable to be received in the input p and q, triggering an external transition. This transition is performed by the $Cons_update$ to update the state of each agent depending on the power consumption and production in input variable and updates the phase to "$Emitting_output$". This triggers the output function, which outputs the value of $Balance$ in $Balance_outpout$ output. After the output function is complete, the internal transition is executed, which uses the $Matrix_calculation$ function to recalculate the currency balance in the market depending on consumption and production and updates the phase to "$Waiting_input$". This process repeats indefinitely. The mathematical representation of this model composed of a number N_{agent} electrical node in the microgrid is given by the equation 3.43.

$$PDEVS_{Financial_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.43}$$

Where :

$X = \{(p, v) \mid v \in \mathbb{R}^{N_{agent}}\} \cup \{(q, v) \mid v \in \mathbb{R}^{N_{agent}}\}$

$Y = \{(Balance_output, v) \mid v \in \mathbb{R}^{N_{agent}}\}$

$S = \{(\boldsymbol{Phase}, Wallets) \mid \boldsymbol{Phase} \in \{\text{“}\boldsymbol{Waiting_input}\text{”}, \text{“}\boldsymbol{Emitting_output}\text{”}\}, Wallets \in \mathscr{W}\}$

With $\mathscr{W}$ the set of possible states for a wallet.

$\delta_{ext}\Big(\big(\text{“}\boldsymbol{Waiting_input}\text{”}, Wallets\big), e, \big((p, v_1), (q, v_2)\big)\Big) = (\text{“}\boldsymbol{Emitting_output}\text{”}, cons_update(v_1, v_2))$

$\delta_{int}(\text{“}\boldsymbol{Emitting_output}\text{”}, Wallets) = (\text{“}\boldsymbol{Waiting_input}\text{”}, Matrix_calculation(Wallets))$

$\lambda(\text{“}\boldsymbol{Emitting_output}\text{”}, Wallets) = (Balance_output, Get_balance(Wallets))$

$ta(\text{“}\boldsymbol{Waiting_input}\text{”}, Wallets) = \infty$

$ta(\text{“}\boldsymbol{Emitting_output}\text{”}, Wallets) = 0$

The starting point for the model, also known as the initialization, is determined using the process outlined in equation 3.44.

$$s_{0d} = (\text{“}\boldsymbol{Emitting_output}\text{”}, Initial_wallets) \tag{3.44}$$

The input ports X of the financial model named p can take values in $\mathbb{R}^{N_{agent}}$, q can take values in $\mathbb{R}^{N_{agent}}$ with N_{agent} the number of agent in the system. The output port Y is named

balance can take values in $\mathbb{R}^{N_{agent}}$. The states of the model, denoted as S, consist of a phase and a wallet variable. The phase, named *Phase*, can have two possible values: **"Waiting_input"** and **"Emitting_output"**. The wallet variable, named $Wallet$, can take values in $\mathscr{W}$. The external transition function, denoted as δ_{ext}, uses the p and q variables to calculate the new $Wallet$ value using the abstract function *cons_update* and changes the phase from **"Waiting_input"** to **"Emitting_output"**. The internal transition function, denoted as δ_{int}, calculate the new $Wallet$ value using the abstract function $Matrix_calculation$ and changes the phase from **"Emitting_output"** to **"Waiting_input"**. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda($"***Emitting_output***"$, Wallets)$, sends the value of *Balance* through the $Financial_output$ port to the connected component. The phases are linked to specific time-advances, with **"Waiting_input"** linked to an infinite time-advance and "***Emitting_output***" linked to a time-advance of 0.

Social Model

The social model allows Bayesian network based TAM to be encapsulated so that the results of the actual use results can be retrieved and passed on to other models during the co-simulation. Its formal representation is provided in equation 3.45 and imaged with the figure 3.67

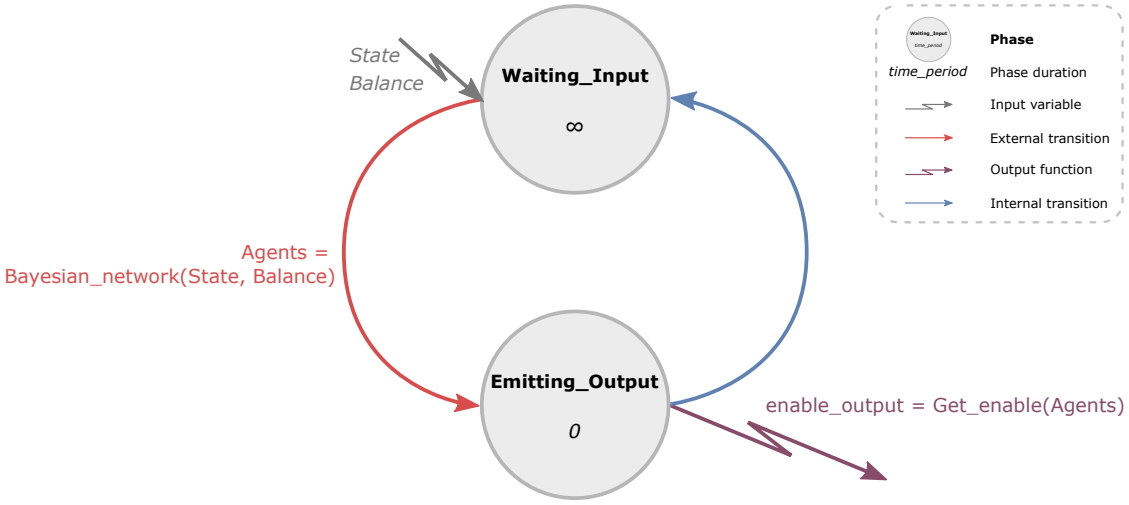


Figure 3.67: Graphical representation of the social model states.

The proposed social model is composed of two main variables: *Phase* and *Agents*. The first variable, *Phase*, can take on two values: "$Waiting_input$" or "$Emitting_output$", and is used to model the system as a state machine. The second variable, *Agents*, is a numerical variable used to store all the data related to the social acceptance. The system is initially set to the "$Waiting_input$" phase, i.e. the *ta* value is set to ∞ and the *Agents* is initialised to a determined value by the user. From the "$Waiting_input$" state, the system waits for an input variable to be received in the inputs *State* and *Balance*, triggering an external transition. This transition is performed by the $Bayesian_network$ to update the state of each agent depending on the power consumption and production and account balance in input variable and updates the phase to "$Emitting_output$". This triggers the output function, which outputs the value of $Get_enable(Agents)$ in $enable_output$ output. After the output function is complete, the internal transition is executed, which updates the phase to "$Waiting_input$". This process repeats indefinitely. The mathematical representation of this model composed of a number N_{agent} electrical node in the microgrid is given by the equation 3.45.

$$PDEVS_{Social_model} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{3.45}$$

Where :

$$X = \{(State, v) \mid v \in \mathbb{R}^{N_{agent}}\} \cup \{(Balance, v) \mid v \in \mathbb{R}^{N_{agent}}\}$$
$$Y = \{(enable_output, v) \mid \in (\mathbb{R}^{N_{agent}})\}$$

$S = \{(\textbf{\textit{Phase}}, Agents) \mid \textbf{\textit{Phase}} \in \{\text{“}\textbf{\textit{Waiting_input}}\text{”}, \text{“}\textbf{\textit{Emitting_output}}\text{”}\}), Agents \in \mathscr{A}\}$

With $\mathscr{A}$ the set of possible states for an agent.

$\delta_{ext}\Big(\big(\text{“}\textbf{\textit{Waiting_input}}\text{”}, Agents\big), e, \big((State, v_1), (Balance, v_2)\big)\Big) =$

$\big(\text{“}\textbf{\textit{Emitting_output}}\text{”}, Bayesian_network(v_1, v_2)\big)$

$\delta_{int}(\text{“}\textbf{\textit{Emitting_output}}\text{”}, Agents) = (\text{“}\textbf{\textit{Waiting_input}}\text{”}, Agents)$

$\lambda(\text{“}\textbf{\textit{Emitting_output}}\text{”}, Agents) = (enable_output, Get_enable(Agents)$

$ta(\text{“}\textbf{\textit{Waiting_input}}\text{”}, Agents) = \infty$

$ta(\text{“}\textbf{\textit{Emitting_output}}\text{”}, Agents) = 0$

The starting point for the model, also known as the initialization, is determined using the process outlined in equation 3.46.

$$s_{0d} = (\text{“}\textbf{\textit{Emitting_output}}\text{”}, Initial_agents) \tag{3.46}$$

The input ports X of the social model named *State* can take values in $\mathbb{R}^{N_{agent}}$, *Balance* can take values in $\mathbb{R}^{N_{agent}}$ with N_{agent} the number of agent in the system. The output port Y is named *enable* can take values in $\mathbb{R}^{N_{agent}}$. The states of the model, denoted as S, consist of a phase and an agents variable. The phase, named *Phase*, can have two possible values: **“Waiting_input”** and **“Emitting_output”**. The agent variable, named *Agents*, can take values in $\mathscr{A}$. The external transition function, denoted as δ_{ext}, uses the *State* and *Balance* variables to calculate the new *Agents* value using the abstract function *Bayesian_network* and changes the phase from **“Waiting_input”** to **“Emitting_output”**. The internal transition function, denoted as δ_{int}, changes the phase from **“Emitting_output”** to **“Waiting_input”**. In the case of confluence between δ_{int} and δ_{ext} functions, the δ_{con} function will trigger the δ_{int}. The output function, denoted as $\lambda(\text{“}\textbf{\textit{Emitting_output}}\text{”}, Agents)$, sends the value of *enable* through the *enable_output* port to the connected component. The phases are linked to specific time-advances, with **“Waiting_input”** linked to an infinite time-advance and “***Emitting_output***” linked to a time-advance of 0.

Final Coupled model

The root model, or final coupled model, of our co-simulation modeling of a microgrid in PDEVS allows us to connect all of the previously presented models of the different components of a microgrid. This coupled model represents the completion of our structural replicative modeling, which integrates the microgrid's fields of energy, finance, and social with a deeper level of modeling possibilities for a more comprehensive understanding of its operation. The root model serves as the central hub for the microgrid model, allowing for the simulation of interactions between the fields models. This can provide insight into the microgrid's operation and the interconnections between its various fields.

$$PDEVS_{root_model} = (X, Y, D, \{M_d\}, EIC, EOC, IC) \tag{3.47}$$

Where :

$X = \{\}$

$Y = \{\}$

$D = \{Energy, Financial, Social\}$

$\{M_d\} = \{PDEVS_{Energy_model}, PDEVS_{Financial_model}, PDEVS_{Social_model}\}$

$EIC = \{\}$

$EOC = \{\}$

$IC = \Big\{\Big((Energy, p), (Financial, p)\Big)\ ,\ \Big((Energy, q), (Financial, q)\Big)\ ,$

$\Big((Energy, state), (Social, state)\Big)\ ,\ \Big((Financial, balance), (Social, balance)\Big)\ ,$

$\Big((Social, state), (Energy, state)\Big)\Big\}$

Model Summary and Overview

In conclusion, the proposed replicative model for microgrid systemic modeling is a powerful tool for understanding the complex interactions between the energy, financial, information and social domains of a microgrid. By utilizing co-simulation, the model allows for the integration of models from different fields, each with its own unique formalism and dynamics, into a single simulation, thereby providing a more holistic and comprehensive understanding of the system being modeled. The use of DEVS as a wrapper enables the synchronization of the different models, ensuring that the co-simulation runs smoothly and accurately.

One of the key challenges in microgrid modeling is the accurate representation of the interactions between the various components of a microgrid, such as loads, sources, batteries, and buses. The proposed model addresses this challenge by utilizing the PyPSA library for power flow modeling. This library is particularly well-suited for microgrid modeling due to its ability to accurately represent these interactions.

Another challenge in microgrid modeling is the accurate representation of the financial behavior of the system. The proposed model addresses this challenge by utilizing a matrix-based cash flow model. This approach allows for a more comprehensive understanding of the financial behavior of the microgrid, taking into account the various factors that can affect the cash flow, such as energy consumption and production, costs and revenues.

Finally, the proposed model also addresses the challenge of understanding the social dynamics of microgrids by utilizing the Technology Acceptance Model (TAM) with Bayesian networks. This approach provides a simple yet effective framework for understanding user acceptance of microgrids. The incorporation of Bayesian networks as a modeling tool improves the accuracy of the TAM by providing a probabilistic framework for understanding the relationships between variables in the context of community microgrids.

Overall, the proposed replicative model for microgrid systemic modeling is a powerful tool for understanding the complex interactions and dynamics of microgrid systems. It effectively addresses the challenges of representing the interactions between various components, financial behavior, and social dynamics of microgrids. The use of DEVS as a wrapper ensures the co-simulation runs smoothly and accurately, resulting in a more comprehensive understanding of the microgrid system.

Chapter 4

Simulator: Formalism and their application for the proposed models

In Chapter 3, a study of the joint modeling of a micro-grid was carried out using three separate models: a structural model, a generative model, and a replicative model. The structural model has been designed to examine the replication of the generic structure presented in the experimental framework of the previous chapter. The generic model has been developed to investigate how to create an interoperable model based on input/output functions that can exchange data despite belonging to different fields of expertise. Finally, the replicative model explored the ability to couple different simulators based on real-world data to study its ability to reproduce the behavior of models from distinct domains of expertise, which were initially designed to work separately.

This chapter will focus on transforming our PDEVS formal models into executable models that simulators can understand, as well as developing the concept of simulation, as described in figure 4.1. The simulations of these models will be carried out using a spreadsheet for the first two models and a PyPDEVS kernel to run the last co-simulation model. The notion of simulation and the internal functioning of these simulators will be developed to understand their formalization better and adequately execute the developed models. Within this framework, we will approach our micro-grid modeling approach, using an iterative method between each model and its simulation to provide an additional layer of abstraction.

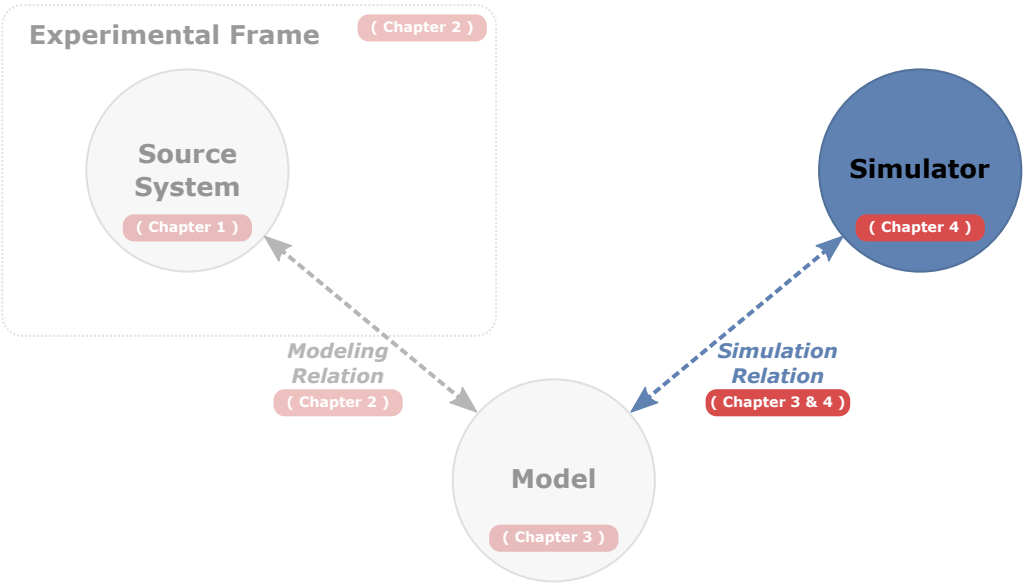


Figure 4.1: Scope of the Chapter 4

4.1 M&S Theory: Simulator formalism

Previous chapters have shown the process of creating models from experimental frames applied to source systems. The last step in the process is M&S theory is using this model to generate the expected behavior. This is done by a simulator.

4.1.1 What is a simulator?

In theory any computational system (including humans) is capable of executing a model by interpreting its semantics to generate its behavior. In a more abstract way, a simulator can be seen as an algorithm, a sequence of actions that respects the semantics of the model. Thus, a simulator operates at a high level of knowledge. In other words, it is a system of high specification hierarchy level being usually built by the association of several components.

In this section, the basics of how simulators work will be described. These concepts will be used in this chapter to explain the simulator associated with each of the models built previously.

4.1.2 General simulator concepts

Simulation is the transformation of high level system structure knowledge into the I/O relations and observation level [5]. This is achieved by taking a given system specification together with its initial state value and generating the corresponding state and output trajectories. Thus the simulator must comply with the conventions of the modeling formalism adopted by the modeler to create the model. This means the simulator must comply with the *time base*, *sequences*, *segments* and *input trajectories*.

At its core, a simulator has two elements that interact with their model counterparts. Models are built from atomic (in the sense of indivisible) components and coupled components. Simulators will associate a simulator to each atomic component and a coordinator with each coupled component. This association is done following the hierarchy of the model itself, with a root-coordinator at the top. A comprehensive and generic messaging protocol is used by simulators and coordinators to execute the simulation.

In order to simulate the microgrid with precision and accuracy, we propose a three-layer modelling approach that leverages the developed models and simulations. The first layer serves as the abstraction layer, representing the microgrid model and identifying all the elements and their respective characteristics. To refine this model, a transformation is carried out to define it in the PDEVS formalism. This step facilitates the separation of the internal elements of the microgrid into a structured model, which allows for better analysis of the different internal fields and the individual components that make up the microgrid. The second transformation transforms this structured model into an executable model, which can be run within the simulator. This allows for the simulation of the built model. The three different layers, from the microgrid model input by the user to the PDEVS executable model, can be identified as seen in Figure 4.2. By following this approach, we can ensure a high level of originality and creativity in our simulation expertise, providing a fresh perspective on the topic. The logical flow of ideas and clear organization also contribute to a technical and academic style, using specific terms and definitions related to strong simulation expertise.

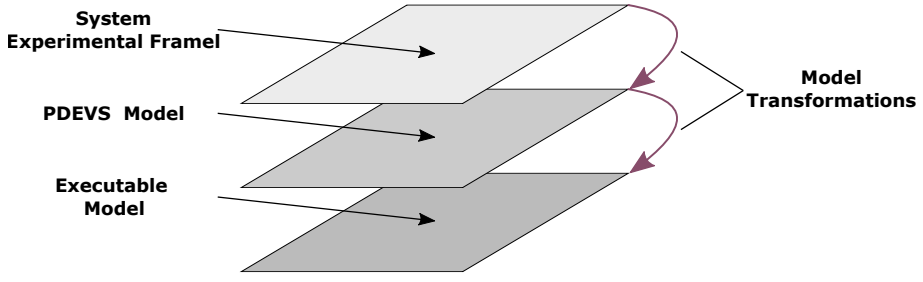


Figure 4.2: Model transformation layers within the microgrid modeling process

During the modeling process, we developed three different models that were based on the Modeling and Simulation theory using the PDEVS formalism. Each model was designed to delve deeper into the modeling structure, providing us with a more comprehensive understanding of the system being modeled.

The first model, the structural model, was designed as a diagnosis model. This model focused on understanding the overall structure of the system and identifying potential issues or areas for improvement. It was essential for our understanding of the system as a whole and for identifying the key components and their interactions. The second model, the generative model, was a component-based model. This model focused on understanding the behavior of individual components and how they interacted with one another. It provided us with detailed insights into the behavior of each component and how they contributed to the overall functioning of the system. Finally, the third model, the predictive model, was a co-simulation model. This model was designed to simulate the interactions between different fields and model the outcome of these interactions. It was an essential tool for understanding how different systems interacted and give an important framework for multidisciplinary modeling of microgrid.

For each model, a simulator was used to help us simulate and analyze the model. A spreadsheet was used for the first two models, which were relatively simple, while the python library PyPDEVS was used for the last model, which was more complex. These tools allowed us to effectively simulate and analyze the models, providing us with valuable insights and helping us to make more informed decisions. As represented in Figure 4.3, the models and simulators were developed in a sequential manner, with each model building on the knowledge gained from the previous one. This approach allowed us to gain a deeper understanding of the system over time and make more informed decisions.

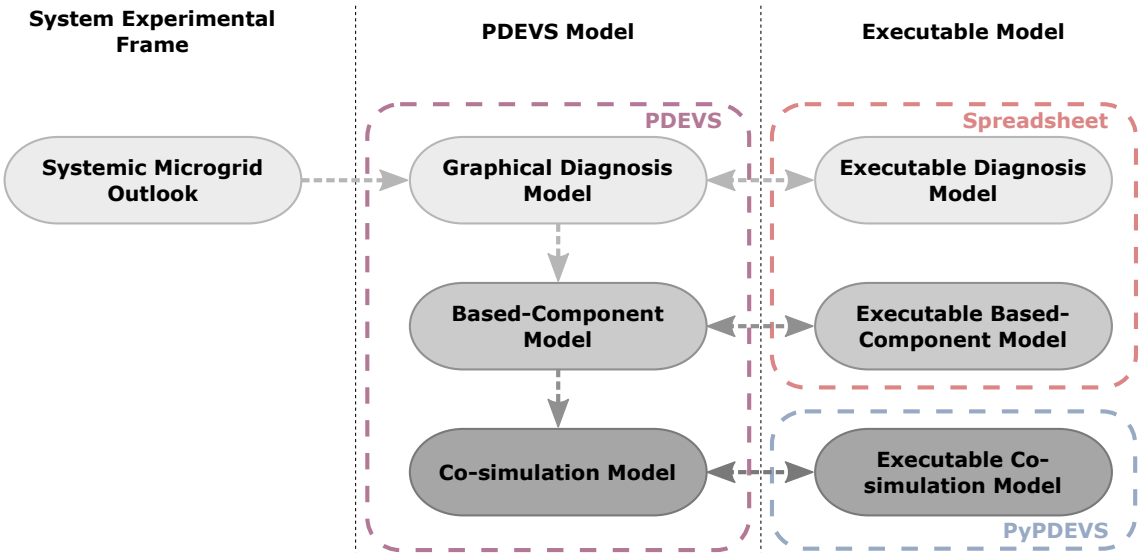


Figure 4.3: Evolution of the modelling approach during the PhD

4.1.3 Spreadsheet-based simulator

During our work, a spreadsheet (in our case, Excel) was used to simulate the evolution of our first models over time. In order to get some results out of these models presented earlier in this manuscript, the requirement of a simple tool and quick implementation of the model was required. Spreadsheets provides a powerfull framework in order to implement complex calculations and algorithms. To implement a DEVS based model, a clarification of the operation of the spreadsheet calculation process is required. Spreadsheets does not calculate the cells in a definite order (for example by rows or columns first) but instead will dynamically determine the order of the calculation of each cell referring from a dependency tree comprising all of the cells that require a calculation. To better understand an example is given in equation 4.1 as follow:

$$\begin{aligned} A1 &= B1 + B1/2 \\ B1 &= 4 \\ C1 &= SUM(A1 : B1) \end{aligned} \tag{4.1}$$

In this case, the dependency tree that will be defined by the software will calculate the cells in the order B1 $\rightarrow$ A1 $\rightarrow$ C1.

Once this is understood, the PDEVS formalism framework needs to be implemented and transformed into an executable model that can be calculated by a spreadsheet. In our case, columns will be treated as the simulation of one atomic component, the lines as the evolution of the time step of the simulation and the cell as the result of one atomic model on a defined time step. The first line will represent the initial state of each atomic models. The PDEVS root model will be transformed in a dependency tree linking all the atomic models of our modeling.

To illustrate, we will consider an example model. The PDEVS models formalized in equations 4.2, 4.3 and 4.4 will be implemented for simulation respectively in columns A, B and C.

$$PDEVS_{Model_A} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{4.2}$$

Where :

$X = \{(Result_B, v) \mid v \in \mathbb{N}\}$
$Y = \{(Result_output, v) \mid v \in \mathbb{N}\}$
$S = \{(\boldsymbol{Phase}, Result) \mid \boldsymbol{Phase} \in$ "$\boldsymbol{Wait_second_model}$", "$\boldsymbol{Send_result}$", $Result \in \mathbb{N}\}$
$q_{init}(Result) =$ ("$\boldsymbol{Wait_second_model}$", 6)
$\delta_{ext}\big(($"$\boldsymbol{Wait_second_model}$", $Result), e, (Result_B, v)\big) =$ ("$\boldsymbol{Send_result}$", $v + \frac{v}{2}$)
$\delta_{int}($"$\boldsymbol{Send_result}$", $Result) =$ ("$\boldsymbol{Wait_second_model}$", $Results$)
$\lambda(Phase, Result) = (Result_output, Result)$
$ta($"$\boldsymbol{Wait_second_model}$", $Result) = \infty$
$ta($"$\boldsymbol{Send_result}$", $Result) = 0$

$$PDEVS_{Model_B} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{4.3}$$

Where :

$X = \{\}$
$Y = \{(Result_output, v) \mid v \in \mathbb{N}\}$
$S = \{Result \mid Result \in \mathbb{N}\}$
$q_{init}(Result) = 4$
$\delta_{ext} = \{\}$
$\delta_{int}(Result) = Result + 2$
$\delta_{con} = \delta_{int}$
$\lambda(Result) = (Result_output, Result)$
$ta(Result) = 1$

$$PDEVS_{Model_C} = (X, Y, S, \delta_{ext}, \delta_{int}, \delta_{con}, \lambda, ta) \tag{4.4}$$

Where :

$X = \{(Result_A, v) \mid v \in \mathbb{N}\} \cup \{(Result_B, v) \mid v \in \mathbb{N}\}$
$Y = \{\}$
$S = \{(\boldsymbol{Phase}, Result) \mid \boldsymbol{Phase} \in$ "$\boldsymbol{Wait_models}$", "$\boldsymbol{Send_result}$", $Result \in \mathbb{N}$

$$q_{init}(Result) = (\text{“}\boldsymbol{Wait_models}\text{”}, 10)$$
$$\delta_{ext}\Big((\text{“}\boldsymbol{Wait_models}\text{”}, Result), e, \big((Result_A, v_1), (Result_B, v_2)\big)\Big) = (\text{“}\boldsymbol{Send_result}\text{”}, v_1 + v_2)$$
$$\delta_{int}(\text{“}\boldsymbol{Send_result}\text{”}, Result) = (\text{“}\boldsymbol{Wait_models}\text{”}, Results)$$
$$\lambda(Phase, Result) = \{\}$$
$$ta(\text{“}\boldsymbol{Wait_models}\text{”}, Result) = \infty$$
$$ta(\text{“}\boldsymbol{Send_result}\text{”}, Result) = 0$$

The model's starting point, or initialization, is determined based on the process outlined in equation 4.5.

$$\begin{aligned} s_{0\,Model_A} &= \{\text{“}\boldsymbol{Wait_second_model}\text{”}, 6\} \\ s_{0\,Model_B} &= \{4\} \\ s_{0\,Model_C} &= \{\text{“}\boldsymbol{Wait_models}\text{”}, 10\} \end{aligned} \tag{4.5}$$

The coupled model formalized in equation 4.6 allows to link the three atomic models together. This is from the link between atomic models inputs and outputs that the executable model implemented in Excel will take its dependencies.

$$PDEVS_{root_model} = (X, Y, D, \{M_d\}, EIC, EOC, IC) \tag{4.6}$$

Where :

$$\begin{aligned} X &= \{\} \\ Y &= \{\} \\ D &= \{Model_A,\ Model_B,\ Model_C\} \\ \{M_d\} &= \{PDEVS_{Model_A}, PDEVS_{Model_B}, PDEVS_{Model_C}\} \\ EIC &= \{\} \\ EOC &= \{\} \\ IC &= \Big\{\Big((Model_A, \text{”}Result_output\text{”}), (Model_C, \text{”}Result_A\text{”})\Big), \\ &\qquad \Big((Model_B, \text{”}Result_output\text{”}), (Model_A, \text{”}Result_B\text{”})\Big), \\ &\qquad \Big((Model_B, \text{”}Result_output\text{”}), (Model_C, \text{”}Result_B\text{”})\Big)\Big\} \end{aligned}$$

In order to implement this executable model in Excel, model A will be implemented in column A , model B in column B and model C in column C as following in equation 4.7 with a three step simulation:

$$\begin{array}{lll} A1 = 6 & B1 = 4 & C1 = 10 \\ A2 = B2 + B2/2 & B2 = B1 + 2 & C2 = SUM(A2 : B2) \\ A3 = B3 + B3/2 & B3 = B2 + 2 & C3 = SUM(A3 : B3) \end{array} \tag{4.7}$$

As one can see, the separation in phases that was required in PDEVS does not need to be implemented here, as the dependency graph and separation in various lines for different simulated times will naturally produce this phase effect. The first line contain all the s_0 of the models and is the initialisation of the simulation. Then, the simulation can start with the following lines that include the different equations to calculate the intern state of each model in each time steps. From this, the spreadsheet software will automatically build a dependency tree resulting from the equations implemented in each of the cells. The dependency tree of one step of simulation (i.e. one line of the spreadsheet) can be represented graphically in figure 4.4 This dependency diagram will be the base of the representation of the dependency tree of one step of the simulation.

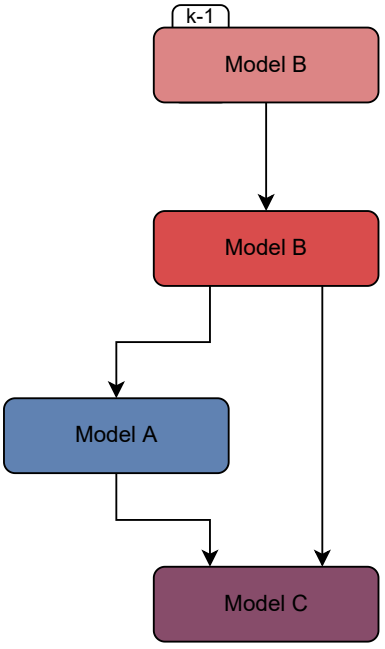


Figure 4.4: Graphical representation of the dependency diagram generated by Excel for one step of simulation

The full dependency tree of this simulation will then be B1 → A1 → C1 → B2 → A2 → C2 → B3 → A3 → C3. Once the software has determined the full dependency tree it can calculate each cells one at a time with the result expressed in equation 4.8.

$$\begin{array}{lll} A1 = 6 & B1 = 4 & C1 = 10 \\ A2 = 9 & B2 = 6 & C2 = 15 \\ A3 = 12 & B3 = 8 & C3 = 20 \end{array} \tag{4.8}$$

The first two models will be simulated using this approach, as will be detailed in section 4.2.

4.1.4 PDEVS-based simulator: The choice of PyPDEVS

As shown in figure 4.5, a DEVS model has a set of five functions: internal transition, external transition, confluence, output and time advance.

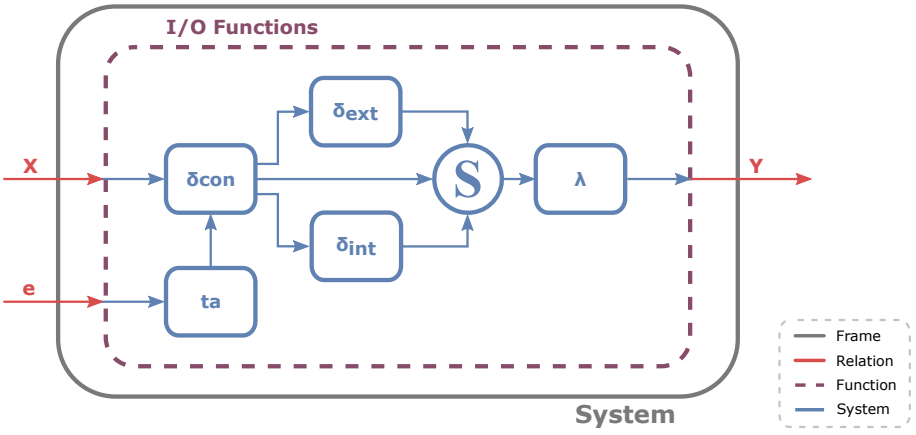


Figure 4.5: A PDEVS-based systems

The dynamics of a PDEVS simulator is shown in figure 4.6. For a specific component, the simulator tests if the elapsed time is equal to the time to the next event. If so, the output is

triggered. After what, the simulator checks for any inputs. If some is found, it will update the state via the confluent transition function, otherwise it will execute the internal transition function. If the time is not elapsed, the simulator only checks for inputs and in their presence executes the external transition function. The simulator then jumps to the next scheduled event time, which is the minimal time of all components in the system.

The PDEVS simulator uses a standardized set of messages, allowing it to be interfaced with all types of model formalisms. These are initialization (*i-message*), event scheduling (**-message*), output (*y-message*) and input (*x-message*) messages.

The initialization message is sent from the root-coordinator to all of its subordinates, providing an initial time. Scheduling messages are sent from the coordinators to their children to trigger new events. Output messages are sent from the children to their parent coordinator whenever an output function is triggered. Input messages are sent from the coordinator to their children to trigger external (or sometimes confluent) transitions.

The paragraphs above describe what is called parallel DEVS approach where each child has its own simulator. The role of the coordinator in this approach is to synchronize all the children. This is done by using an even list that holds a pair of simulator/next time information. The root-coordinator implements the overall simulation loop.

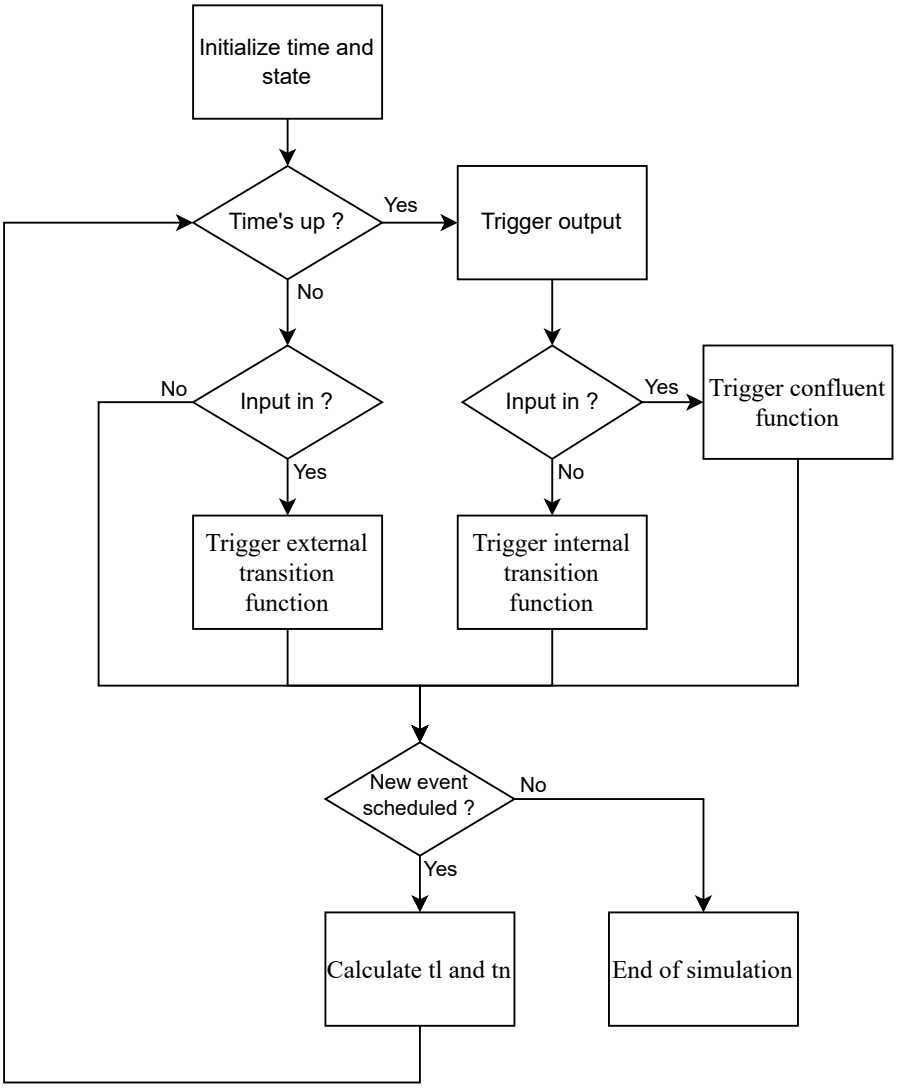


Figure 4.6: The basic PDEVS model sequence

The PDEVS formalism provides a highly elegant and simplistic approach to modeling complex

systems, but its implementation can be a challenging and time-consuming task. This is where the use of a library or simulator that provides a pre-built DEVS simulation kernel comes in.

When we began our project, we knew that we needed a library that would meet a number of specific characteristics. First, it had to be compatible with a wide range of scientific inputs, which is why it had to be based on a language that is already well-known and widely used. Second, it had to be scalable and able to evolve as our models and simulations developed. Third, it had to have high performance and be able to run simulations quickly. And finally, it had to be easy to use and implement.

After evaluating several options, we decided to use the Python library PyPDEVS. This library offers a variety of simulation kernels from the DEVS family, including PDEVS and DEVS with dynamic structure. Furthermore, it is based on the Python programming language, a widely-used and well-known language with a vast repository of available libraries.

The use of PyPDEVS allowed us to quickly and easily implement our models using pre-existing libraries, and its features facilitated the release of new models [254]. Additionally, this simulator has been proven to have good performance in several scientific papers [255]. It also provides a high level of flexibility, allowing us to easily adapt our models to different scenarios and to make changes as needed [256].

One of the key advantages of PyPDEVS is its ability to schedule multiple models within the executable model. This feature is especially useful for our multi-component systemic model, which requires communication between various components represented as formally separated models. The ability to handle simultaneous events is crucial in this type of simulation and PyPDEVS allows this seamlessly. This means that we are able to simulate the interactions between different components of our model in real-time, providing us with a more accurate representation of the system.

PyPDEVS is a DEVS library that provides a pre-built simulation kernel for modeling and simulating complex systems, including the PDEVS static simulation kernel which we will use for our co-simulation. The library offers a variety of simulation kernels from the DEVS family, including PDEVS and DEVS with dynamic structure.

In terms of modeling, PyPDEVS uses the hierarchical structure of DEVS models. Atomic models are the basic building blocks of the simulation, while coupled models connect and coordinate the atomic models. This structure allows for a clear and modular representation of the system being modeled, making it easy to understand and modify. In terms of implementation, PyPDEVS provides a set of Python classes and functions that can be used to define and execute DEVS models. These classes and functions abstract the complexity of the simulation algorithm, making it easy to implement and use. The library also provides support for visualization and data analysis, allowing us to easily analyze and interpret the results of our simulation.

One of the key advantages of PyPDEVS is its ability to handle multiple models within the executable model, which is crucial for our co-simulation that requires communication between various components represented as formally separated models. It also allows for dynamic structure changes during the simulation, which is useful in situations where the structure of the system under study is subject to changes over time, as can microgrids. However, due to a lack of time, this dynamic part was not explored in this work. Its parallel discrete event simulation algorithm, hierarchical structure of DEVS models, co-simulation capability, and easy implementation make PyPDEVS a suitable choice for our co-simulation. To better understand the role of a wrapper from a DEVS formalism perspective, figure 4.7 shows the class structure of a the PDEVS implementation within the PyPDEVS simulator.

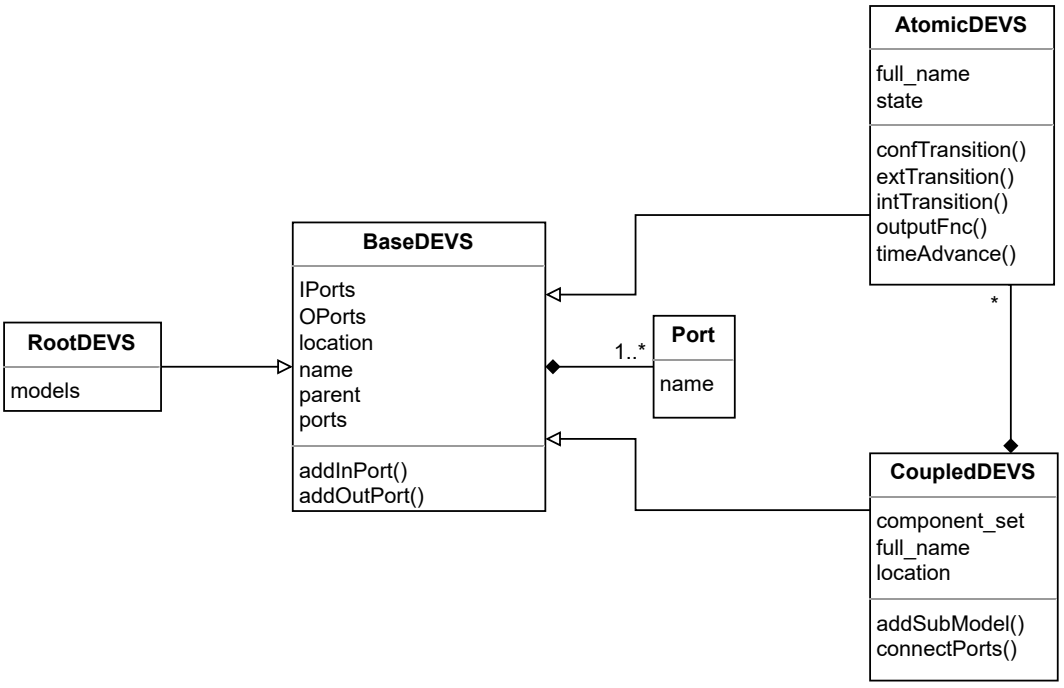


Figure 4.7: DEVS class structure from PyPDEVS

This figure shows the relations between typical PDEVS elements, with the BaseDEVS structure on top and AtomicDEVS, CoupledDEVS and RootDEVS inheriting its base structure. *BaseDEVS* is the generic I/O overview of a model, without its inner mechanics given by the transition functions. *AtomicDEVS* was previously explained in Figure 3.7 in page 50 and contains all inner mechanics missing in BaseDEVS. *CoupledDEVS* contains sub-models and represents here the Figure 3.5 in page 48. *RootDEVS* is a particular type of coupled model that has the same characteristics but is also can manage the execution of all its inner coupled or atomic models like the master coordinator. *RootDEVS* maintains a list of models all linked to a different simulator.

The class diagram in figure 4.8 for the executable model implemented in PyPDEVS for the example model in the PDEVS formalism framework is as follows:

- The atomic model classes (Model A, Model B, Model C) inherit from the atomicDEVS class and each have their own states, inputs, and outputs. They also have a Calcul() function to represent their inner dynamics and a variable for their phase and result.
- The root model class (PDEVS Root Model) inherits from the coupledDEVS class and contains the three atomic model classes as internal variables. It also has a ConnectAllPort() function to establish the connections between the models based on their inputs and outputs.

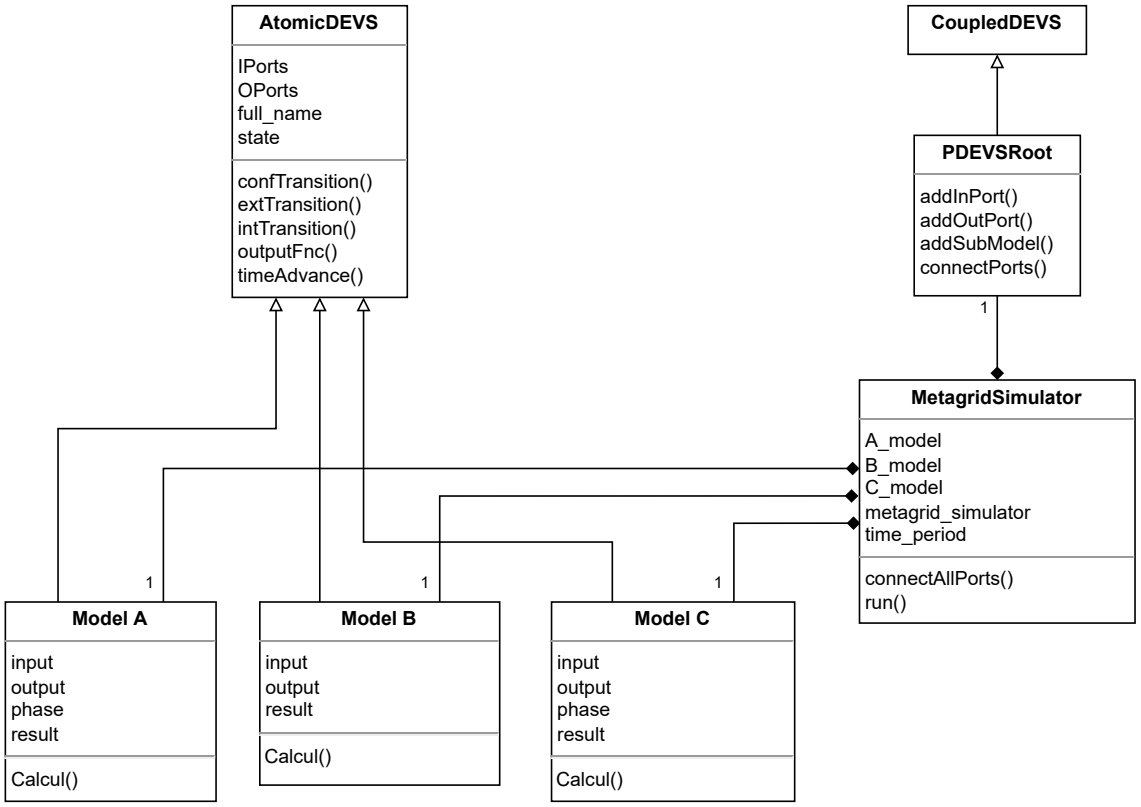


Figure 4.8: DEVS class structure from PyPDEVS

4.1.5 DEVS bus

The approach shown above can be generalized to any formalism via a DEVS-bus. This approach *wraps* the model in DEVS form, allowing coupling and their execution via a standard coordinator. Events-based models will be pre-scheduled by the coordinator. Discrete event models will be run when the simulation absolute time reaches their next step. Differential equations models require holding the value of their previous simulation steps in order to run their numerical integration.

We will use this notion of wrapping models that use different simulators into DEVS models in our cosimulation approach.

4.2 Microgrid: Simulating the proposed models

4.2.1 First simulator

The concept of microgrid equilibrium is central to understanding the functioning and sustainability of microgrid systems. The goal of a microgrid is to provide a stable source of energy that operates in harmony with its various components and exchange mechanisms. The dynamic interactions between the different fields of the microgrid can greatly impact its success or failure, and it is crucial to ensure that all fields receive as much as they give.

In order to assess the health and potential evolution of microgrid systems, a simulation model was developed. To gather the necessary information for the model, a report from the World Bank was utilized [24]. This report documents the experiences and challenges of microgrids in rural areas worldwide and provides a wealth of information on various themes related to microgrid governance, architecture, population, and more. The data from this report was extracted and used to rate each microgrid system through a questionnaire. The questionnaire results were then applied to the simulation model to give a visual representation of the strengths and weaknesses of the microgrid and to provide insights into its potential evolution.

The simulation results were presented in two stages. Firstly, the general health of the microgrid was recorded through the questionnaire and the results were visualized by the microgrid graphical model presented in figure 3.18 in page 59, using a color code ranging from red to dark green to indicate the strengths and weaknesses of the microgrid. Secondly, the questionnaire results were used as an initial state for the simulation model, which was developed using the DEVS formalism and implemented on a spreadsheet. The simulation model represents the links between the flow and exchange components and allows for the simulation of the future dynamics of the microgrid.

It is important to note that this simulation is not meant to represent the actual evolution of the microgrid or to predict its behavior, but rather to provide a rough indication of the flows that should be monitored in order to maintain the microgrid in a satisfactory state of health. The simulation accumulates the advantages and disadvantages of the microgrid into the stock of each component flow, and a microgrid that does not have an average score over 1.5 points will see its stock decrease rapidly.

To better understand the implementation of the simulation model in the spreadsheet, a representation of the dependency tree was created in figure 4.9. The dependency tree shows the sequence of a simulation step for the calculation of exchanges and flows, and highlights the fact that the exchanges are calculated first using data from the previous simulation step, followed by the calculation of flows using the received exchanges.

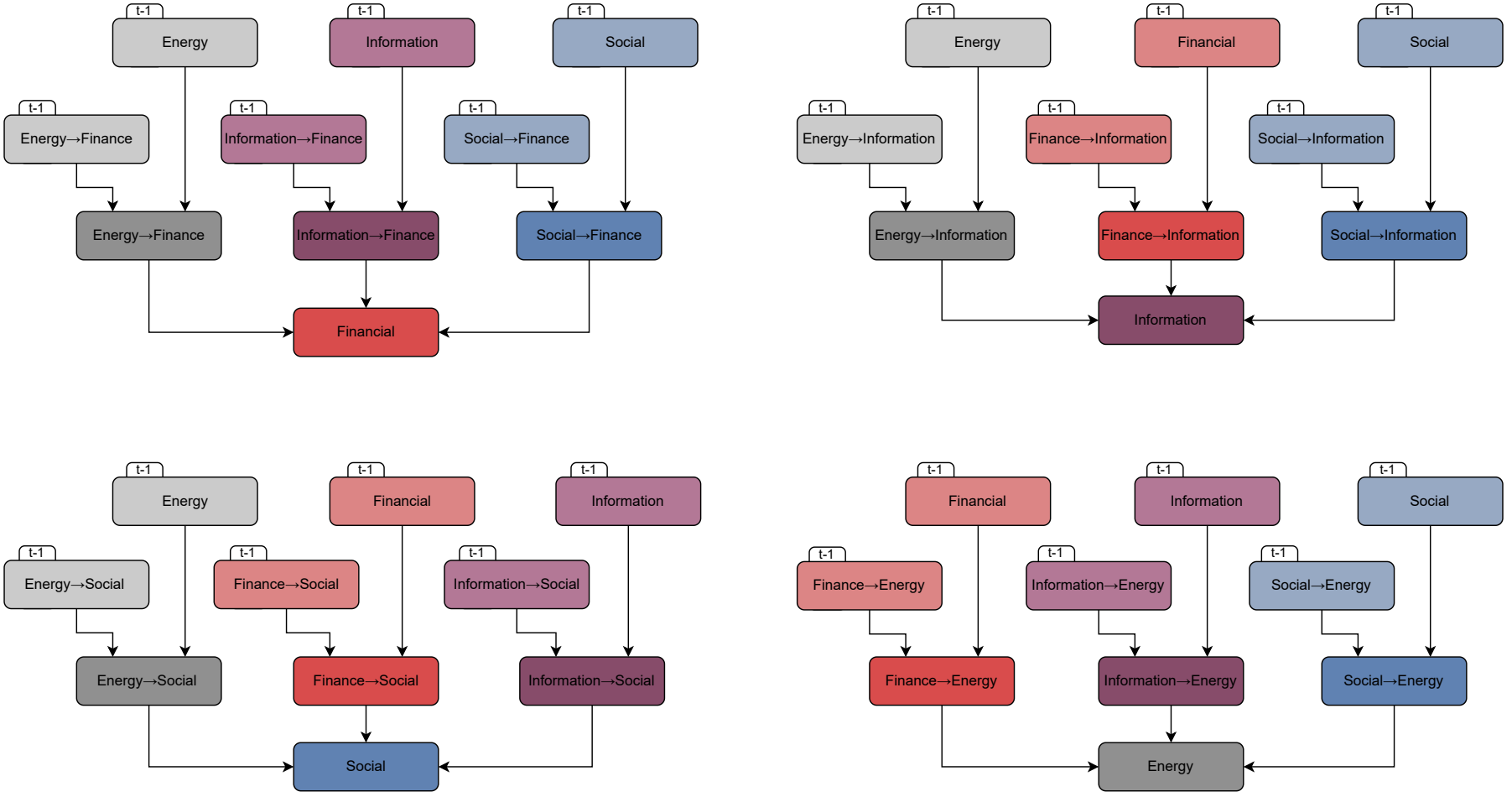
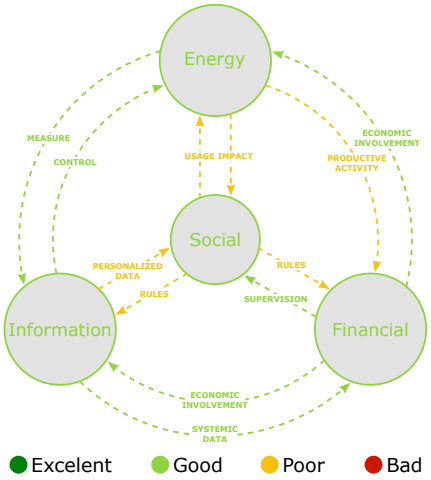
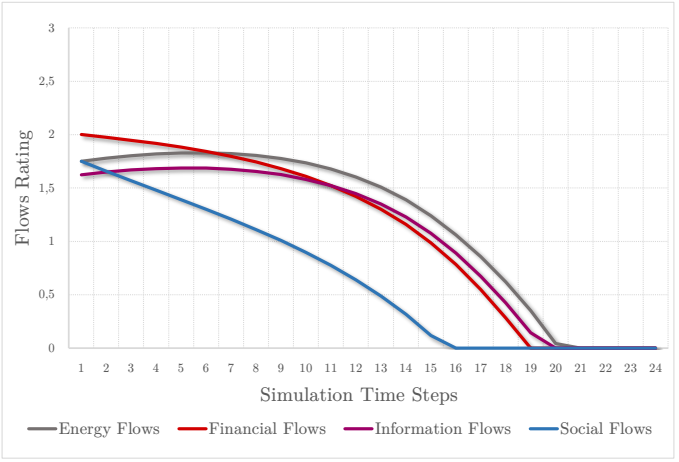


Figure 4.9: Representation of the dependency tree that is generated by the spreadsheet

CREDA



(a) Result based on the questionnaire



(b) Potential evolution of the microgrid based on the simulation

Figure 4.10: Result of the simulation of CREDA microgrid

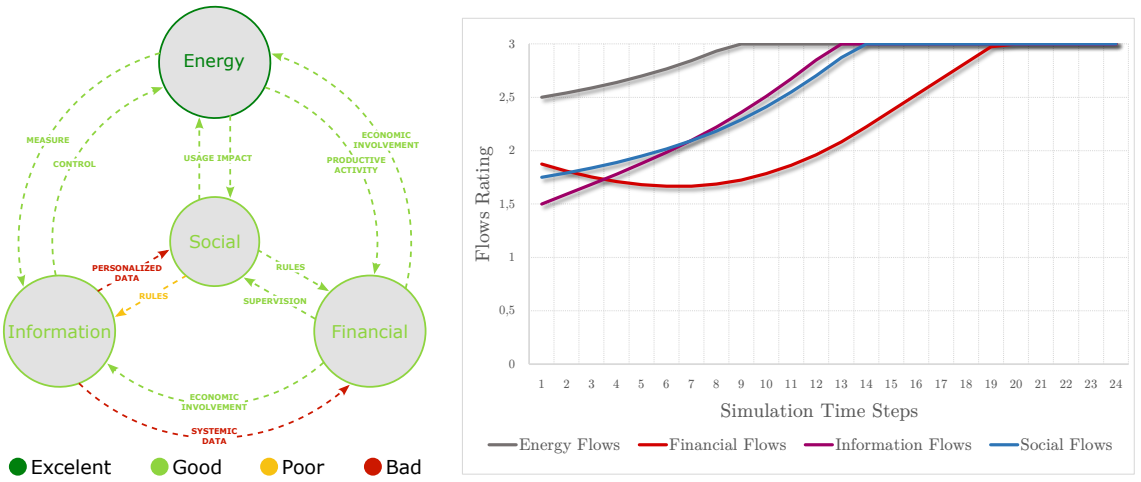
The CREDA microgrid is an innovative solution for providing rural communities in India with access to reliable and sustainable energy. The organization enables the establishment of microgrids in areas not served by the general Indian electricity grid, utilizing efficient and cost-effective solar panel technology. An advanced control and verification system tracks users' monthly consumption, allowing for efficient management and maintenance of the microgrid.

However, the microgrid faces several challenges. Despite the efficient collection system, the organization needs help with timely payment collection from users, which negatively impacts the stability and sustainability of the microgrid. Additionally, users are not satisfied with the limited amount of energy provided, often resorting to connecting unplanned loads, which overload the system. This, coupled with insufficient energy production, raises concerns about the long-term expansion and sustainability of the microgrid.

The CREDA microgrid is a commendable initiative for providing rural communities access to sustainable energy. However, the challenges faced by the organization in terms of timely payment collection and energy production raise concerns about the long-term systemic sustainability of the microgrid.

The general health of this microgrid can be seen in figure 4.10(a), where the grid appears to be in an excellent overall state with exchanges and flows that approximate a score of around 1.5. At first glance, the system has short-term sustainability, thanks partly to its four flows with a respectable score. However, we can observe in the simulation that the grid loses its sustainability in the long term. Indeed, the dynamics of its decline shown in figure 4.10(b) is relatively slow. Therefore, this quick simulation could be interpreted as a system that is relatively sustainable in the short term with its good overall health but that requires some work to reflect on the long-term and improve several exchanges and flows in order to be able to confidently consider this microgrid in good health for many years to come.

DESI



(a) The cells of the PV plant

(b) Potential evolution of the microgrid based on the simulation

Figure 4.11: Result of the simulation of DESI microgrid

The DESI Power microgrid is a decentralized energy system serving communities in rural areas. The company conducts extensive surveys to identify the most viable and profitable markets and provides microgrid solutions using a combination of self-funding and grants from organizations such as the World Bank Development Market Prize and government subsidies.

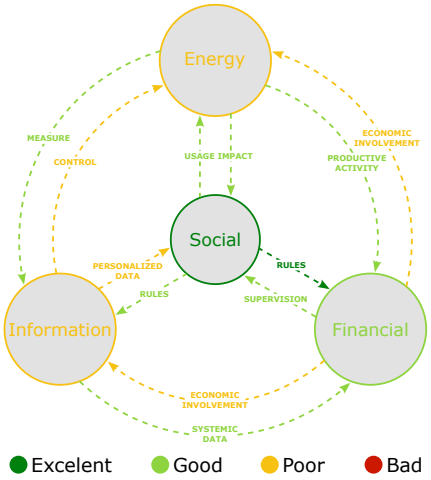
The microgrid operates at an approximate cost of 0.13 USD per kilowatt and has a well-established tariff collection system, with active collectors visiting daily to collect payments. The microgrid offers flexible consumption options to customers based on their spending power and employs a fair penalty system to ensure discipline in the case of non-payment. DESI Power aims to provide a better service than the central grid, with lower energy prices and a higher grid quality, to attract and retain customers.

However, the microgrid faces several challenges, such as technical problems that may only sometimes be solved efficiently and issues of energy theft and overconsumption due to the lack of advanced information processing systems. The area's central grid operates poorly, creating a free market for DESI Power's microgrid solutions.

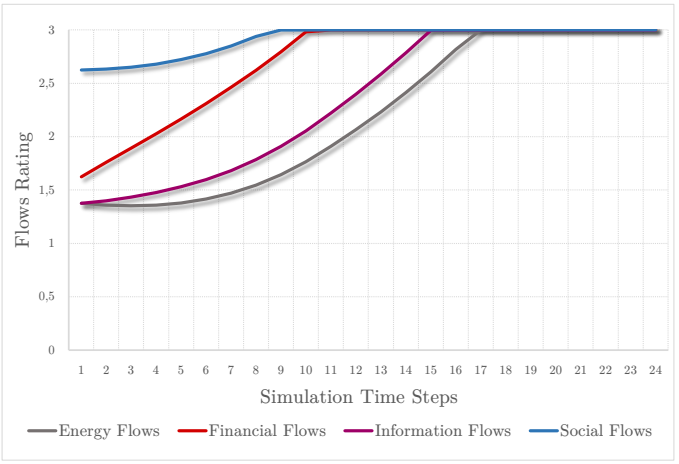
The DESI Power microgrid provides a sustainable energy solution for communities in rural areas, offering affordable energy with efficient payment collection systems. Despite the challenges, the microgrid has been successful in its operations, as evidenced by its active customer base. The microgrid also highlights the importance of proper information processing systems in ensuring the systemic sustainability of decentralized energy systems. To further improve its operation and address the challenges, DESI Power must prioritize implementing advanced information processing systems to monitor energy consumption and detect overuse.

As seen in figure 4.11(a), the overall health of this microgrid appears to be reasonably good, with four healthy flows and most of the exchanges having an honorable rating. Although the consideration of the information component for the community is lacking, the grid seems to have several assets to solidify its long-term sustainability. The simulation confirms this in figure 4.11(b), where we can see all the flows trending towards sustainability. However, it should be noted that the system's dynamics remain relatively slow, which would indicate that this microgrid could still be significantly affected by the occurrence of a major problem in its operation, such as climate change, technical issues, or unexpected problems in the community.

GE/T/P



(a) Result based on the questionnaire



(b) Potential evolution of the microgrid based on the simulation

Figure 4.12: Result of the simulation of GETP microgrid

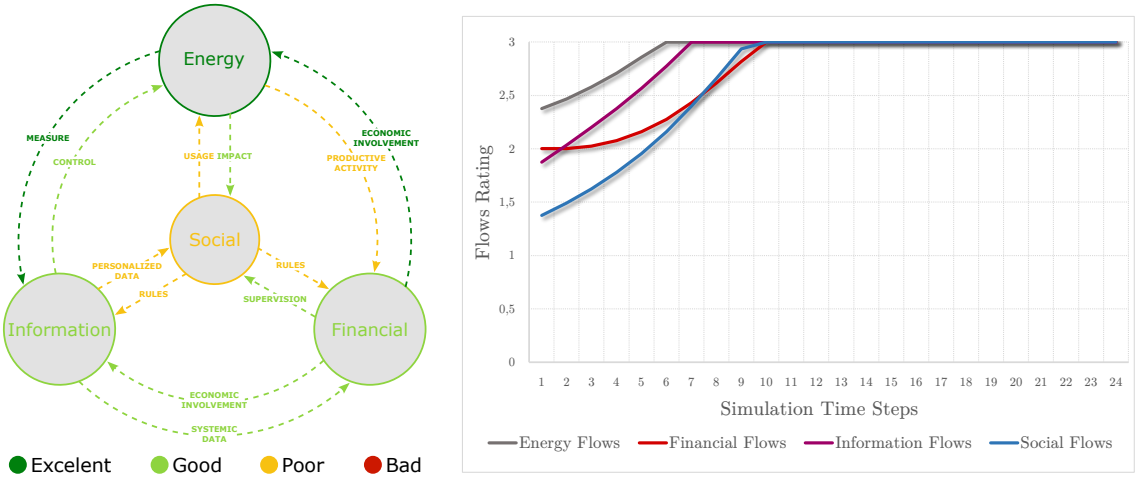
The GE/T/P microgrid is a community-driven project aiming to achieve the electricity grid's sustainability. It operates based on hydropower, providing an environmentally-friendly, low-carbon energy source. The community is at the center of this microgrid, and its users are empowered and trained to participate in the grid's maintenance, governance, and development.

However, the microgrid faces several challenges, including seasonal intermittencies in energy supply due to dependence on rainfall, rapidly increasing energy demands that can only sometimes be met and governance issues caused by community management. The microgrid is entirely financed by the community and lacks external financial support, which makes it vulnerable to equipment breakdowns and limits its resilience to significant events such as extreme weather conditions.

Despite these challenges, the social aspect of the microgrid remains strong and contributes to its stability. Integrating data from the microgrid into a systemic model will provide a clearer understanding of the interactions within the system. To enhance the systemic sustainability of the microgrid, greater financial valorization of the energy produced or seeking external organizations for support is necessary. Overall, the GE/T/P microgrid demonstrates a promising model for community-driven and sustainable electricity generation.

The figure 4.12(a) in our simulation shows that the microgrid seems to have a relatively balanced state of health, thanks partly to solid cohesion and social learning among its initiators. In terms of the simulation presented in figure 4.12(b), the overall sustainability of the grid is driven first by an increase in social and financial flows, which then pushes the entire grid towards long-term sustainability. Although the growth dynamic remains relatively slow, the microgrid displays a relatively high level of resilience.

EDH



(a) Result based on the questionnaire

(b) Potential evolution of the microgrid based on the simulation

Figure 4.13: Result of the simulation of EDH microgrid

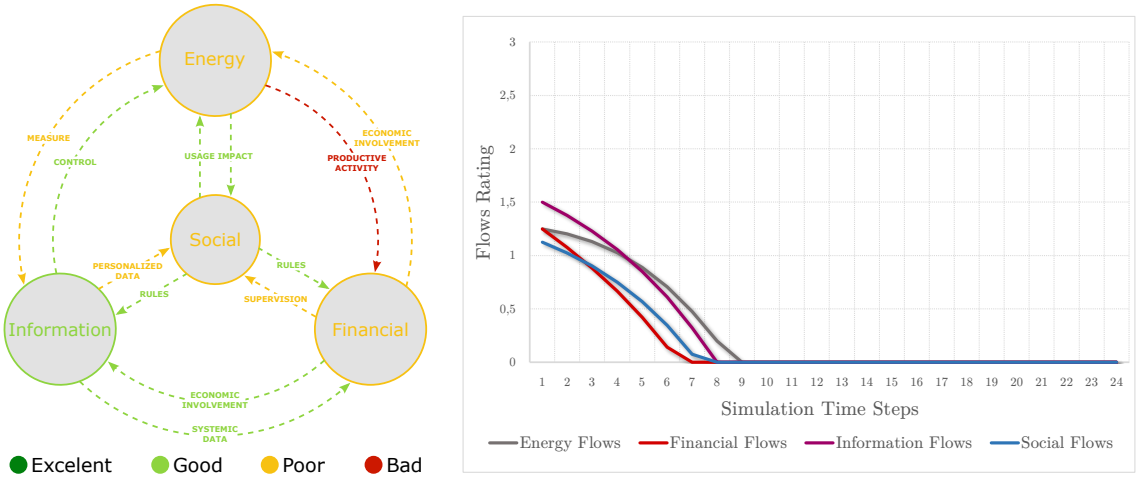
EDH is a government-owned utility company responsible for managing the main grid in Haiti and implementing microgrids in rural areas. The microgrid operates by utilizing generators to provide electricity to the community. However, the complexity of fuel management and operation has been a persistent challenge for EDH. Additionally, the electricity tariffs, currently set at around 0.35$/kWh, and the high theft rate, over 30%, could be better for the Haitian market to develop electricity grids sustainably.

Despite these challenges, EDH has experienced some successes in its microgrid operations. Customers have shown a willingness to make payments for electricity as long as the microgrid is providing power. However, the low-cost recovery levels and oversized generators, which operate at low efficiency and increase operational costs, have restricted the cost recovery capabilities of the microgrid operators. Inequitable decision-making, based on political connections, has also impacted the performance and maintenance of the microgrid.

The microgrid operation and development in Haiti are faced with numerous challenges, including weak cost recovery, oversized generators, political favoritism, and high operational costs due to fuel. These challenges, combined with the need for more access to project documents, slow down the development and sustainability of the microgrid. However, the willingness of customers to pay for electricity and the successes experienced by EDH demonstrate the potential for growth and improvement in the microgrid system. Overall, EDH's microgrid presents a unique opportunity for sustainable energy solutions in Haiti and serves as a case study for developing microgrids in challenging market environments.

The questionnaire results displayed in Figure 4.13(a) clearly demonstrate the energy success of this microgrid, funded by the State of Haiti. However, it is pretty evident that the social aspect of this system has been mostly overlooked. Although the simulation in Figure 4.13(b) appears to show a rapid growth towards the sustainability of this grid, it is still important to measure its results. The various problems within this microgrid do not necessarily guarantee the overall sustainability of the system. In fact, it is mainly unlimited state participation that keeps this microgrid afloat.

HPS



(a) Result based on the questionnaire

(b) Potential evolution of the microgrid based on the simulation

Figure 4.14: Result of the simulation of HPS microgrid

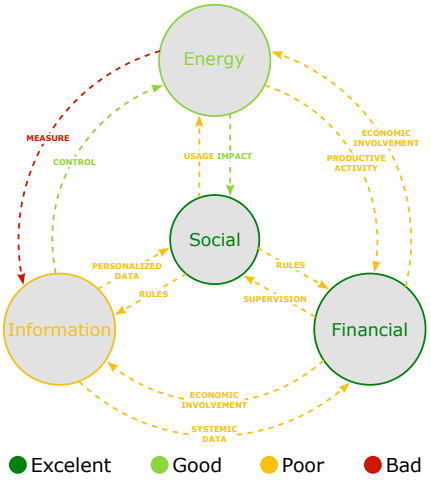
The HPS microgrid is a self-sustained electrical system that operates autonomously from the traditional power grid. Its primary objective is to provide reliable and cost-effective energy to remote communities. The microgrid operates based on a business model that integrates renewable energy sources, energy storage systems, and a local distribution grid. This microgrid community comprises stakeholders, such as local residents, businesses, and government organizations, who work together to ensure the microgrid's stability and growth.

The HPS microgrid faces several challenges, including limited energy resources, high energy demand, and a need for the technological infrastructure. Despite these challenges, the microgrid has accomplished several successes, such as reducing energy costs, improving energy efficiency, and promoting the use of renewable energy. Additionally, the microgrid has also helped to reduce the carbon footprint in the community, improve the quality of life of local residents, and support the development of local businesses.

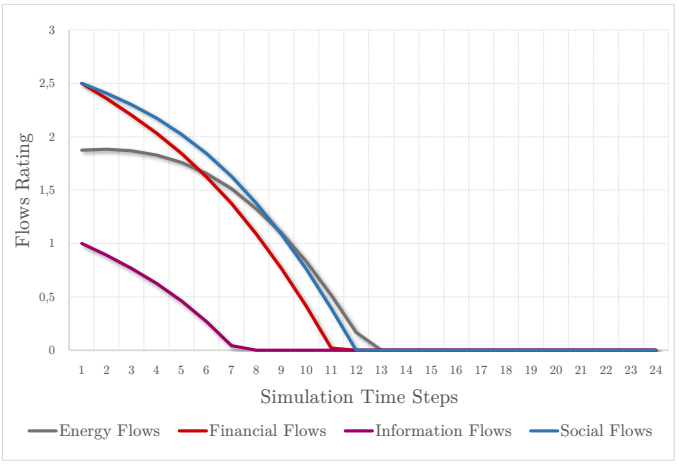
The HPS microgrid is a valuable asset to the community it serves. Despite facing several challenges, its successes have demonstrated the potential of sustainable microgrids to provide reliable and cost-effective energy to remote communities. However, this microgrid's overall lack of sustainability highlights the need for continued efforts to improve its energy efficiency, integrate more renewable energy sources, and enhance its technological infrastructure. These efforts will ensure that the HPS microgrid continues to play an essential role in promoting sustainability and providing clean energy to its community for years to come.

As we can see from the questionnaire result in figure 4.14(a), most of the system's flows are already in a bad state. Additionally, the lack of interaction between the energy and financial components may pose significant problems for the economic viability of the energy grid. Furthermore, in figure 4.14(b), the overall dynamics of the system seem to be heading towards a rapid grid collapse. This simulation result shows and depicts a fragile system that could fail quite quickly in the face of the first major challenge that presents itself. We can therefore conclude a general durability that appears to be fragile or even very fragile at the level of this system.

OREDA



(a) Result based on the questionnaire



(b) Potential evolution of the microgrid based on the simulation

Figure 4.15: Result of the simulation of OREDA microgrid

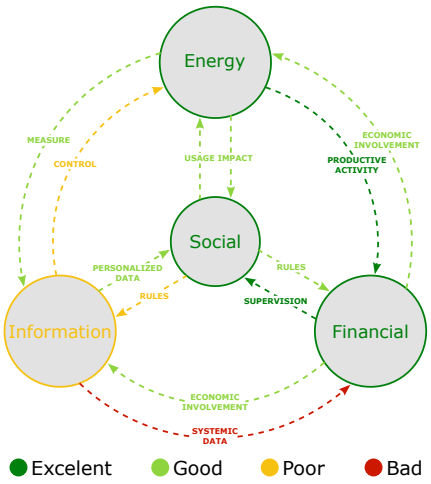
The OREDA microgrid is a decentralized energy system designed to bring electricity to remote populations that lack access to traditional grid infrastructure. The microgrid operates by harnessing renewable energy sources, such as solar and wind, to generate power for local communities. The business model is based on a pay-per-use system, where the community pays for the electricity they consume.

Despite its rapid growth, the OREDA microgrid has faced several challenges regarding reliability, maintenance, and community involvement. The microgrid is often less reliable than its peers due to poor maintenance practices, making it difficult to reach the maintenance company when issues arise. Additionally, disputes within villages and a lack of community involvement have contributed to the system's poor performance. On the other hand, OREDA has achieved notable successes in terms of providing basic lighting needs for remote populations and its well-thought-out energy, financial and social aspects. However, the maintenance aspect of the microgrid lacks sustainability in the long term, with the community needing more skills to maintain the systems on their own. The lack of cooperation within the community, theft, overuse, and rebel factions have also complicated the maintenance process.

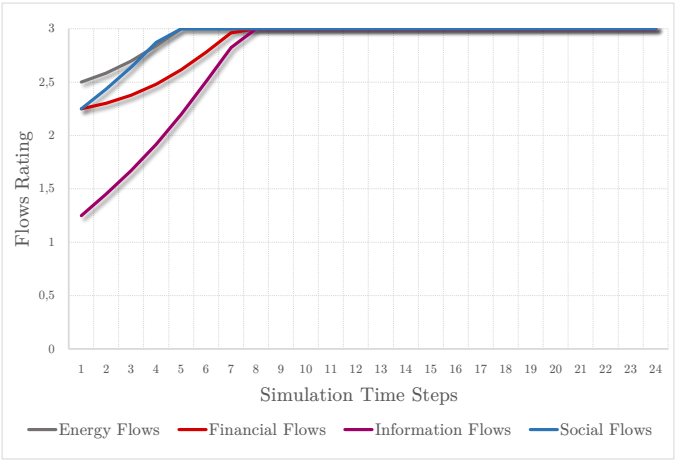
While the OREDA microgrid presents a promising solution for remote communities, it still needs to improve its sustainability and reliability. A more comprehensive approach with a stronger emphasis on maintenance and community involvement is necessary to overcome these challenges. The energy, financial, and social aspects of the microgrid are commendable, but to truly achieve success, all aspects of the system must be integrated and sustainably managed in the long term.

The results of the questionnaire, depicted in figure 4.15(a), show that while most of the microgrid flows are relatively healthy, the connections that link them are feebly managed and organized. This general trend of the microgrid is reflected in figure 4.15(b) by a relative collapse of each component of the microgrid. This shows that despite a solid social, financial, and energy base, the lack of cohesion within the microgrid will not allow it to maintain high overall health for this system.

WBREDA



(a) Result based on the questionnaire



(b) Potential evolution of the microgrid based on the simulation

Figure 4.16: A PV plant and its cells of WBREDA microgrid

The West Bengal Renewable Energy Development Agency (WBREDA) is a state agency that develops and operates microgrids in the West Bengal region of India. Their microgrids are designed to meet local communities' energy needs by integrating various energy sources to ensure energy security and stability.

The WBREDA microgrids are unique in their approach to community engagement and ownership. The agency creates a community-based business model where the energy consumers play an active role in the management and operation of the microgrid. This includes collecting payments and voting on decisions regarding the microgrid's development. The combination of government subsidies, community involvement, and a well-designed economic framework has enabled WBREDA to provide essential energy services with low power and limited availability to meet the primary needs of local communities.

Despite its successes, the WBREDA microgrid is not without challenges. To improve the sustainability of the microgrid, further engagement with the community in the maintenance of the electrical grid is necessary. Additionally, monitoring and analyzing the microgrid's operation and energy consumption habits would provide valuable insights for optimizing the microgrid's performance.

In conclusion, the WBREDA microgrid represents a successful community-based renewable energy development model. The combination of technical expertise, community engagement, and government support has created a sustainable energy solution that meets the needs of local communities in West Bengal.

The figure 4.16(a) depicts a relatively balanced microgrid with components in good condition, and most of the exchanges are well planned and integrated within the microgrid. This overall good rating in the figure 4.16(a) gives a promising simulation of the microgrid's future state in the figure 4.16(b). Although the microgrid has a relatively basic energy consumption, the overall business model seems well suited to the community it serves by avoiding overbuilding the microgrid too quickly, which could harm its sustainability. This is why the simulation results show a rapid rise towards the microgrid's sustainability but should be punctuated by different energy level expansions to evolve the community towards new energy tiers.

Discussion

In this section, we have explored a diverse range of microgrids, each with unique strengths and weaknesses. These microgrids had vastly different overall objectives, ranging from social cohesion for community development to financial gain through providing energy to rural communities. Our systemic questionnaire allowed us to assess each microgrid and visualize the results using a graphical model. Additionally, simulating the evolution of the health of these microgrids yielded varied results depending on the initial data input into the model.

Our systemic approach provided a comprehensive understanding of the areas impacted by a microgrid, particularly in the case of a grid established in a rural area where all components are closely intertwined. With this questionnaire and visualization tool, we can efficiently synthesize a general understanding of the functioning and quality of the microgrid, where all the diversity of this system can be viewed at a glance. Furthermore, the model developed to predict the future health of the microgrid is quite interesting, particularly regarding the dynamics of its evolution. We can quickly visualize which component will be impacted first during the aging of the microgrid, or which component will stabilize the grid in the long term. Additionally, the speed at which the system will converge towards an equilibrium state is an indicator of the general strength or weakness of the system. It is another form of visualization of the graphical model where a color represents each score.

However, this model and its simulation do reveal some limitations. The questionnaire is quite complex and requires a significant amount of information and expertise to fill out. This questionnaire should be simplified to make it more accessible and user-friendly. Additionally, feedback should be gathered from microgrid users, managers, operators, and scientists to improve, correct, and further validate the relevance and accuracy of each question. As for the simulation, the model remains quite simplistic. It does not truly represent the evolution of the health of the grid but rather an image of its resilience. There is a saturation issue within the model, where the notation of flows will necessarily converge towards zero or three, making the simulation results unrealistic for the actual evolution of the microgrid. To more accurately represent the evolution of the health of these flows, a saturation that covers the full range of possibilities between zero and three would be needed. It should also be noted that it would have been interesting to implement constraints when assessing the microgrid, with a different dynamic of our model, to resemble real-world scenarios more closely.

This model and simulation are the first of three in our study. It is crucial as it aims to link the four components of energy, information, finance, and social elements within a microgrid. While this model may not serve as a tool for predicting the future health of a microgrid, it highlights the importance of understanding and visualizing the interactions and relationships between the various components. The interdependence of all these elements within a microgrid is crucial to understanding and improving the system's overall performance.

4.2.2 Component Based Simulator

A simple example to illustrate this model: peer-to-peer energy selling

The generative model, as discussed in chapter 3, provides a comprehensive understanding of the internal components within each domain of a microgrid by modeling and comprehending the interactions between these domains. This is achieved by consolidating the vastly different and varied energy, information, financial, and social aspects into a single morphism, which greatly simplifies the simulation.

To illustrate our model, we have chosen a simple microgrid system for the purpose of clarity. As shown in Figure 4.17, the microgrid consists of two members, Member A and Member B, who engage in a peer-to-peer energy transaction. Member A has a source, energy storage, and sells energy to Member B, who has a load.

The electrical grid is the network connecting Member A's source and battery to Member B's electrical load and is controlled by a power electronics system. The information network performs two critical tasks: measuring the amount of energy consumed to calculate payment made by the bank and measuring the amount of money available in Member B's wallet to ensure payment if energy is consumed. If Member B does not have sufficient funds, their load will be disconnected. The financial network links the wallets of both members and is facilitated by a bank that transfers funds between the wallets. Lastly, the social aspect is modeled through a storage of user acceptance, with different levels of influence based on the user. For example, Member A's acceptance increases when a payment is made to their account, while Member B's acceptance increases when they use their load.

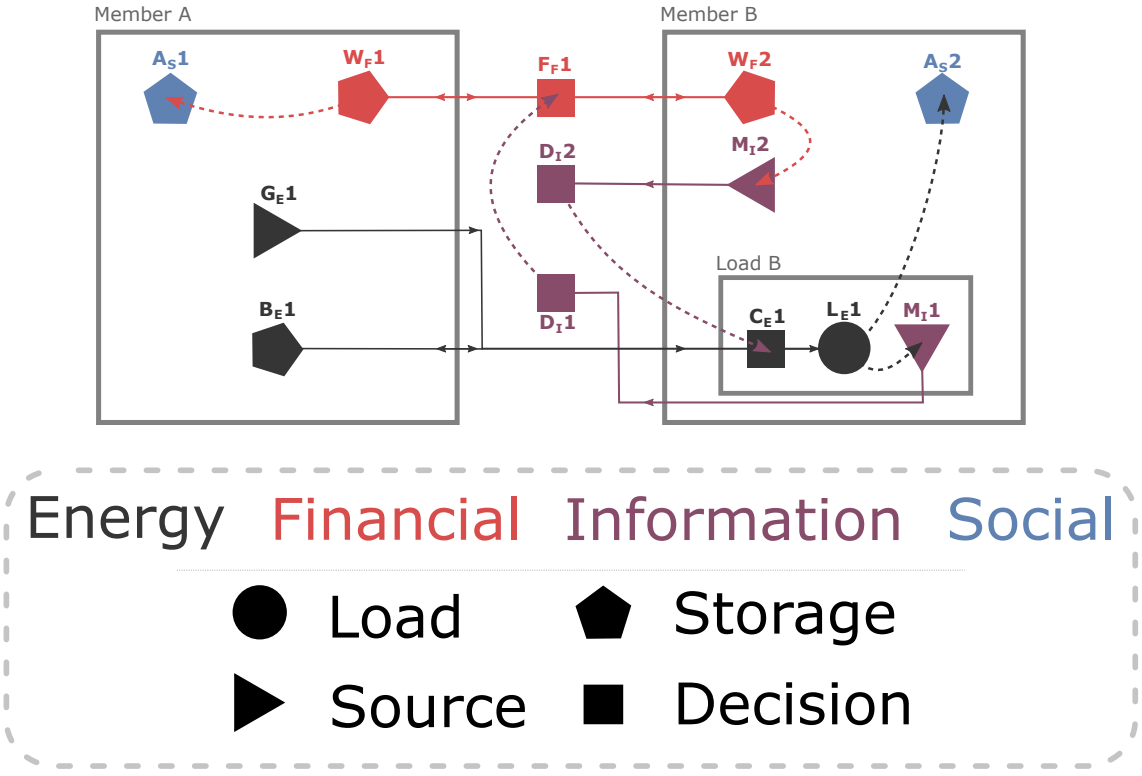


Figure 4.17: Graphical representation of the component-based microgrid model

The various models presented in chapter 3 must then be transformed into an executable model that can be used by our simulator. Here, as we previously mentioned, the simulator used is a spreadsheet, so it is necessary to transform all the models into a list of equations that can transcribe the dynamics of each component. Thus, in our executable model, each column of our spreadsheet will correspond to a component of the system, and each row of that same spreadsheet will represent an additional time step of our simulation. Figure 4.18 provides a graphical representation of the dependency tree of our spreadsheet.

Once our microgrid model is transformed, the simulation dynamics will be as follows: at each time step, given that the source does not depend on any other components, it will be calculated first. The other components will then be calculated one by one according to their dependence on the results of another component. The component that sees all these dependencies already calculated will be the next to be calculated by the spreadsheet, and so on until all the cells of our simulation are calculated. It should also be noted that some of the components will depend on results belonging to the previous time step of the simulation, that is, they depend on a calculation of the previous line.

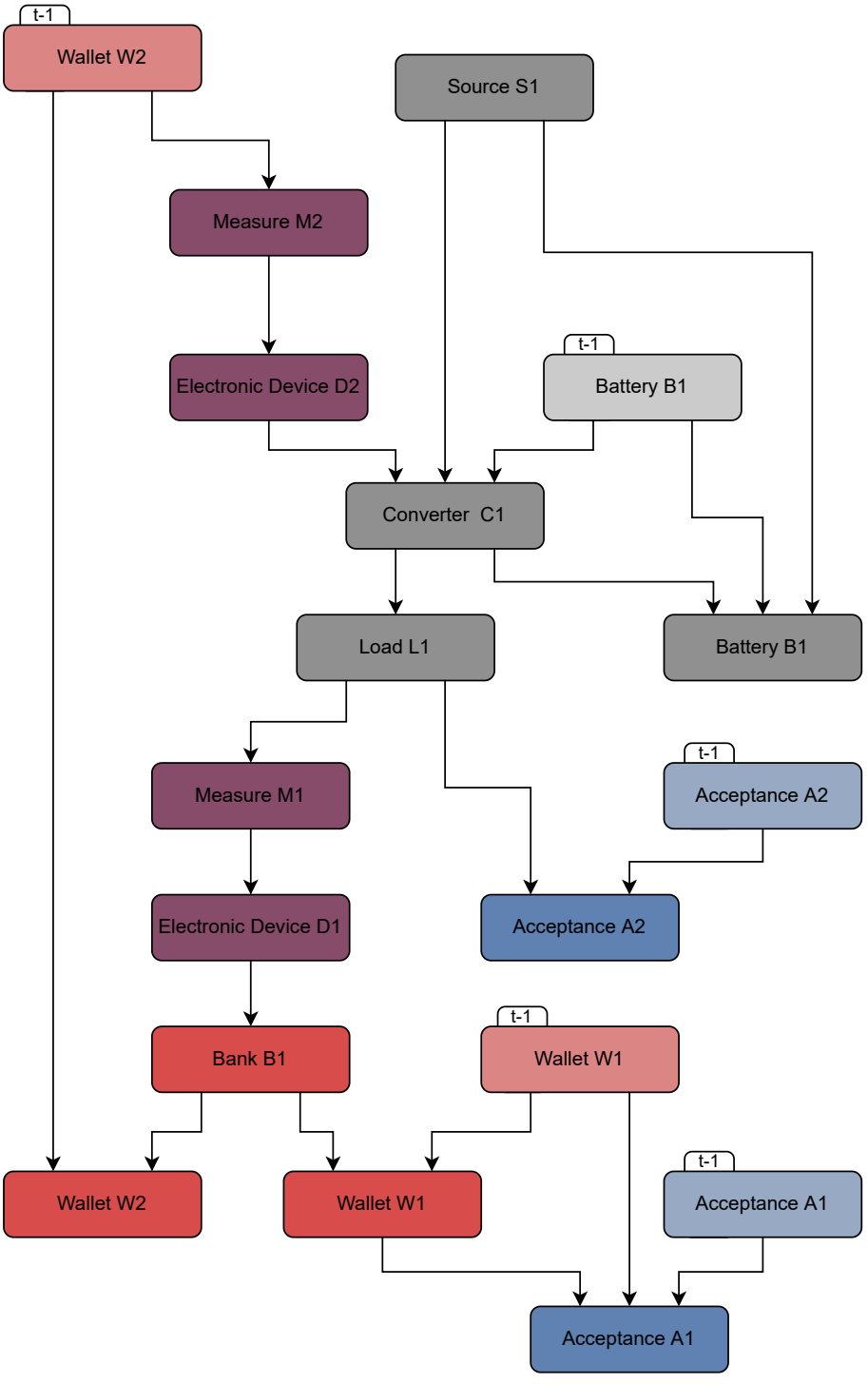
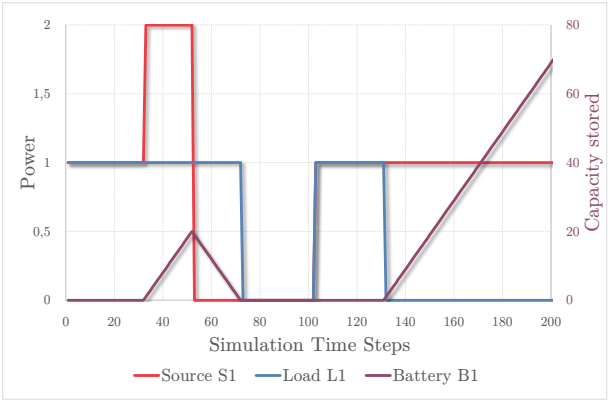
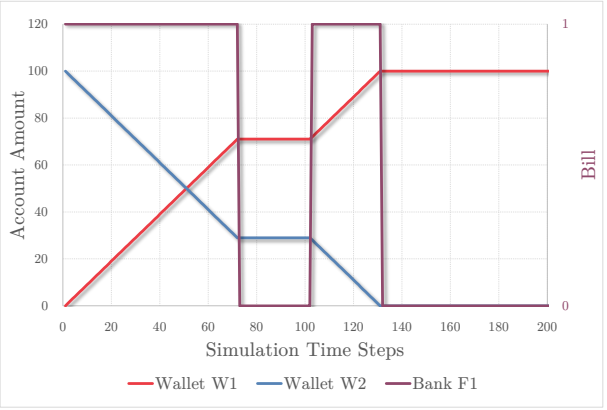


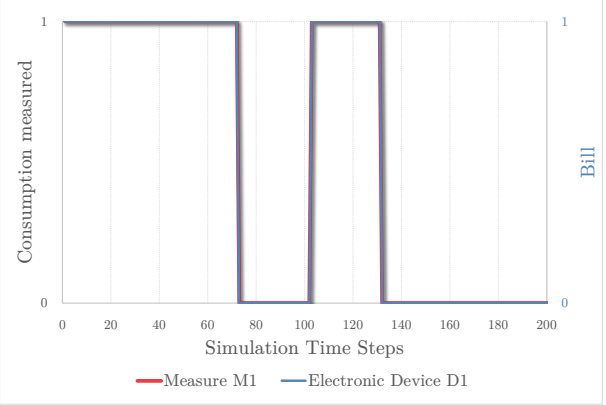
Figure 4.18: Representation of the dependency tree that is generated by the spreadsheet



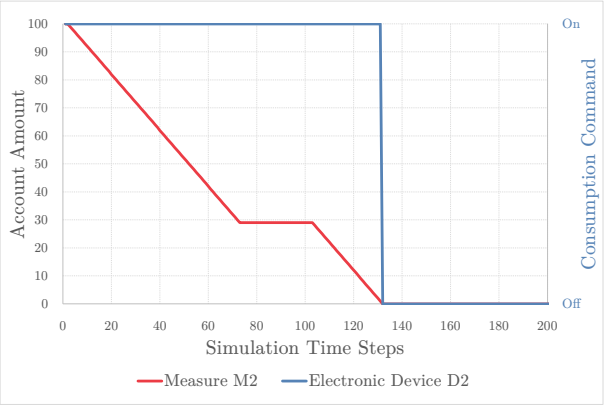
(a) Energy component variation during the simulation



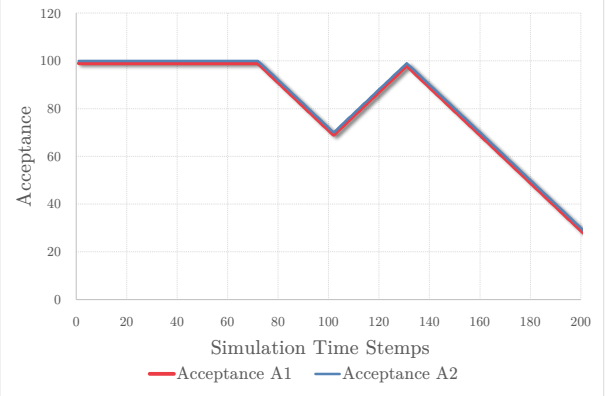
(b) Financial components variation during the simulation



(c) Information components variation during the simulation



(d) Information components variation during the simulation



(e) Social components variation during the simulation

Figure 4.19: Components evolution during the simulation

After conducting the simulation in its entirety, we obtain the following results as detailed in Figure 4.19. We clearly see the functioning of the electrical network in Figure 4.19(a), where the load effectively consumes when there is available energy and drops to zero when the source stops producing and the battery is empty. Similarly, we easily notice that the battery fills up when there is excess energy in the network. In terms of financial flows illustrated in Figure 4.19(b), the bank acts as a two-thirds trust for financial exchanges that can only take place when the bank certifies that a payment should be made. It can also be observed in Figure 4.19 that Wallet 2 fills up as Wallet 1 empties. In terms of the information network, we can see in Figure 4.19(c) that the

measured consumption corresponds well to the note that will be rented to the user. Furthermore, in Figure 4.19(d), electronic tools are disabled when the measurement of the money remaining in the user's wallet falls to zero, thus deactivating the user's load from the electrical network. The figure 4.19(e) shows the acceptance variation of the two members of the microgrid. We can see that these two acceptance variations follow the same size, given that, even if they are not influenced by the same variable, the two influencing variables are extremely linked: if the user does not consume, he obviously does not pay. Moreover, it can be noted that user acceptance increases when the microgrid energy system is functioning correctly.

Difficulties encountered

In the systemic simulation of microgrids, the modeling of each component presented specific challenges that required a change of paradigm in the PDEVS model and executable model. We adopted an incremental development approach, starting with the straightforward electrical analogy and gradually incorporating new characteristics and interactions between components. This allowed us to understand the unique difficulties associated with each component. Thus, this component-based modelling allows highlighting in an atomic way all the microgrids' internal dynamics to represent as close as possible to reality the microgrids' blocking points or failure factors. To go further in the simulation of this model, representing the hierarchical links of each component using an ontology would be of help in future works.

The electrical layer presented the most obvious analogy, as it closely mirrors the microgrid system itself. It was the first layer of abstraction that we attempted to model and served as the foundation for the simulator, as it required an accurate representation of electrical exchanges between components to validate simulation results. However, the large variety of existing electrical components greatly complicates the modeling process, making it necessary to revisit and recheck existing electrical flow simulators with only slight differences in approach.

The primary added value of this simulator lies in the interconnection of all components and the inclusion of often overlooked components such as social, financial, and information. This holistic approach allows for a more comprehensive and realistic simulation of microgrid systems.

In the systemic simulation of microgrids, the modeling of the economic component presents a unique set of challenges. The component-based approach, while useful in some areas, makes calculations of financial exchanges between actors in the microgrid overly complex without adding significant value. The separation of portfolios and financial coordinators into separate components multiplies the number of connections within the simulator, making it difficult to evolve the model and add new members to the grid. It would be more efficient to simplify the financial exchanges between actors and model them as a single entity.

Additionally, the information layer presents difficulties in modeling data measurements and the components that house control, management, and monitoring algorithms. This component makes the model overly complex without providing significant value in diagnosis or finalizing the model. It would be more appropriate to make this component almost transparent at the level of data exchanges and measurements, and model only the algorithms or behaviors that manage the microgrid.

The social component, while important in measuring the social health of the microgrid, presents challenges in modeling human psychology and behavior. Social acceptance is an interesting factor to consider, but many characteristics and behaviors can influence it within the microgrid. Modeling acceptance as a gauge that is filled or emptied over time allows for straightforward modeling, but this approach is limited as it is not based on realistic behaviors of human psychology. It would be beneficial to develop this model by relying on existing models of acceptance that are more complex.

4.2.3 Co-simulation based simulator

In the preceding chapter, we delineated the overarching architecture of our microgrid simulation framework, incorporating energy, financial, and social models to encapsulate the respective domains. The resulting executable model, depicted in Figure 4.20, establishes interconnections between the three modeled domains. Initially represented in PDEVS modeling form, we are required to transform these models into an executable format for integration into PyPDEVS.

The executable model must maintain congruence with the originally developed PDEVS model. The existing literature tends to focus on determining the input responsible for a failed output, rather than formally defining the internal dynamics of microgrids. Our interdisciplinary modeling approach provides the foundation for a systematic formalization of the microgrid's internal functioning. By merging energy, financial, and social models, we achieve a deeper understanding of the root causes of microgrid failures. Our model acts as a diagnostic tool by determining the microgrid's operational state through a balance of system equilibrium. Further formalization of component functions is imperative for future progress in this area.

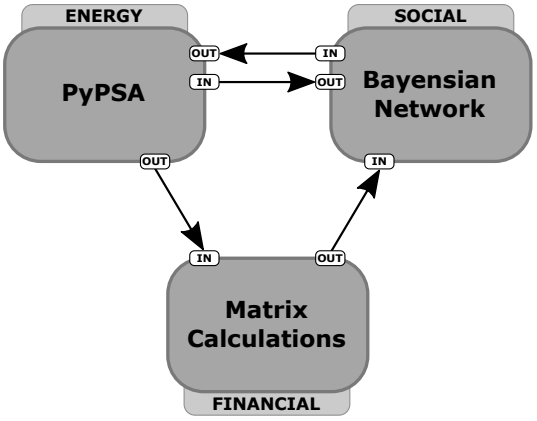


Figure 4.20: Architecture of the microgrid co-simulation executable model

Our simulation tool PyPDEVS is grounded in the Theory of modeling and simulation, as put forth by Ziegler. This foundation allows for a more efficient and seamless transformation of the theoretical model into an executable one. The model integrated into the simulator follows the same architecture as depicted in Figure 4.21, with additional tools and features that enhance its functionality. To effectively execute the model, it must be linked to the co-simulation coordinator, which aggregates all atomic models into a root model that can be run by PyPDEVS with the necessary parameters.

To enhance user experience, we have designed a simple, user-friendly interface that enables the construction of a microgrid for simulation purposes. The object-oriented interface allows the user to add electrical and energy components, connect them, and add agents (consumers or producers) that are linked to each component. The user only needs to enter three parameters for the simulation to run, with all other connections made transparently. After the simulation is completed, we have integrated a data visualization tool to help extract relevant information. PyPDEVS provides an easy integration of elements into the executable model, which we have combined with the Python library matplotlib to create easily interpretable graphs.

The simulator operates within a co-simulation architecture, where the user inputs all microgrid elements and actors, which are then distributed and communicated through the co-simulation interface. The interface, based on the egg-parallel formalism, reads all inputs and outputs and schedules each model. The scope is available to retrieve all information passing through the simulation for display purposes.

To go further, we will present in this part the input and output connections of each of the components so that all the internal models have access to the variables necessary for their execution and the visualisation of the simulation results.

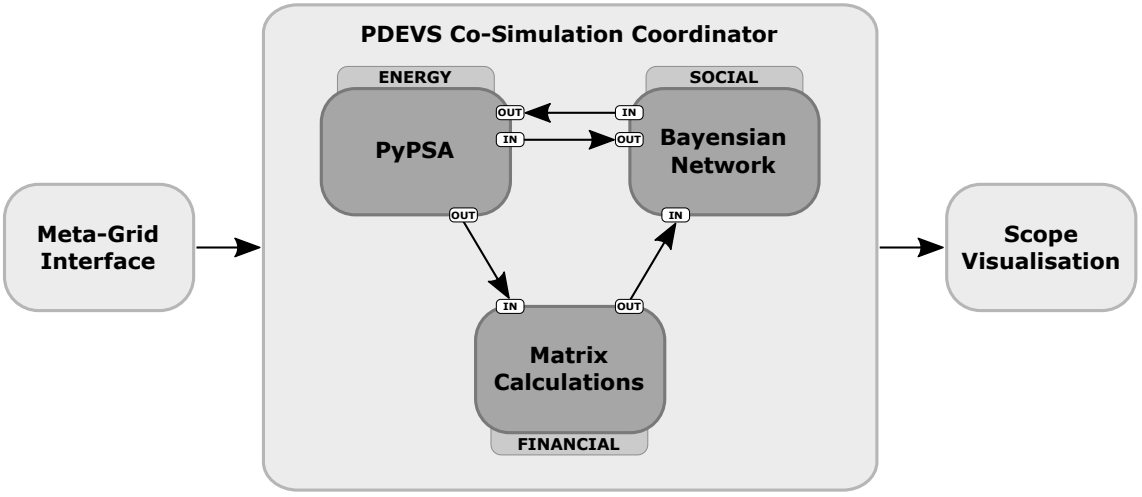


Figure 4.21: Archtitecture of the microgrid co-simulation

The PyPDEVS simulation platform is equipped with an integrated executable model that enables the simulation of microgrids. The simulation process is executed in a well-defined sequence, as illustrated in Figure 4.22, which ensures a systematic and dynamic simulation outcome.

The first step in the simulation process is the calculation of the energy network. This is the most critical and time-sensitive part of the simulation and must be performed before any other calculations. The updated energy flow information is then used to calculate the financial model, which distributes payments among users based on their energy consumption. The final step in the simulation process is the calculation of the social atomic model. This step is crucial in determining the acceptance level of each user based on their energy consumption or perception of energy prices. The simulation then collects the desired data for visualization in the graph, and the process is repeated until the end of the simulation.

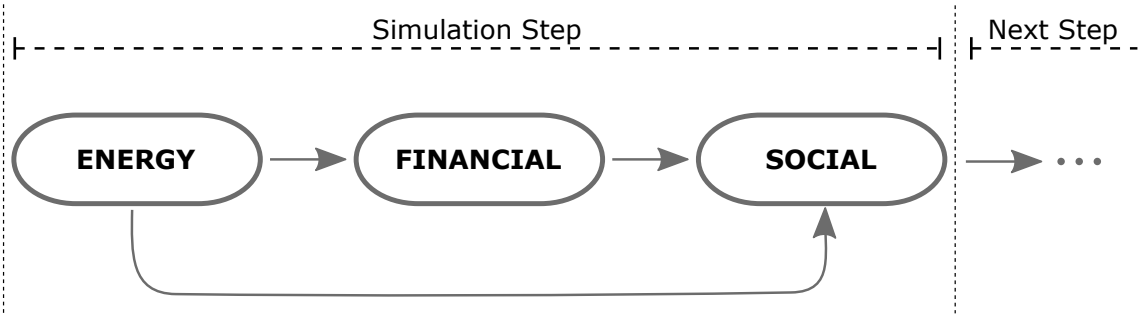


Figure 4.22: Scheduling a simulation step of the executable model

Root Model: Building the co-simulation model

The proposed class diagram for the co-simulation model, described in Figure 4.23, is aimed at gaining a deeper understanding of its internal structure. The *Element* class, at the center of the diagram, is derived from the *AtomicDEVS* class and augmented with two additional specifications: roles and domains. Every modeled element in the microgrid is linked to a specific domain and falls under the energy, financial, or social class. Energy elements also have an *ElementRole* in the electrical network as a source, storage, load, or converter.

The models for each domain inherit from the *Element* class and describe their internal variables that are linked to their specific model. The *MetagridSimulator*, which brings all models together and launches the simulation, inherits from the *CoupledDEVS* class and represents an input and output network.

Each of the three domain models are implemented using PyPDEVS, as described in the previous chapter. The *EnergyModel* integrates the PyPSA network for energy flow calculations and takes

user-entered input states through the *EnergyModelState* class, linking the user interface to the executable model in the meta-Grid simulator. Load shedding, where the network cuts the load consuming the most energy when there is not enough power to supply all consumers, is also implemented in the model.

The *FinancialModel* performs its calculations and cash-flow using classical python matrix calculation libraries, but incorporates a random number generator through the *FinancialModelState* class to add unexpected events to the financial data entered by the user for each agent. This results in salary and total cost variations over time or between simulations.

The *SocialModel*, on the other hand, uses Bayesian network calculation libraries to predict user acceptance, all users having the same TAM Bayesian network. The *SocialModelState* integrates all the data given by the user for the proper simulation of the *SocialModel*

The model developed in this thesis is available for download here in https://gitlab.laas.fr/metagrid/metagrid-simulator, with a step-by-step tutorial here https://gitlab.laas.fr/metagrid/metagrid-simulator/-/wikis/home.

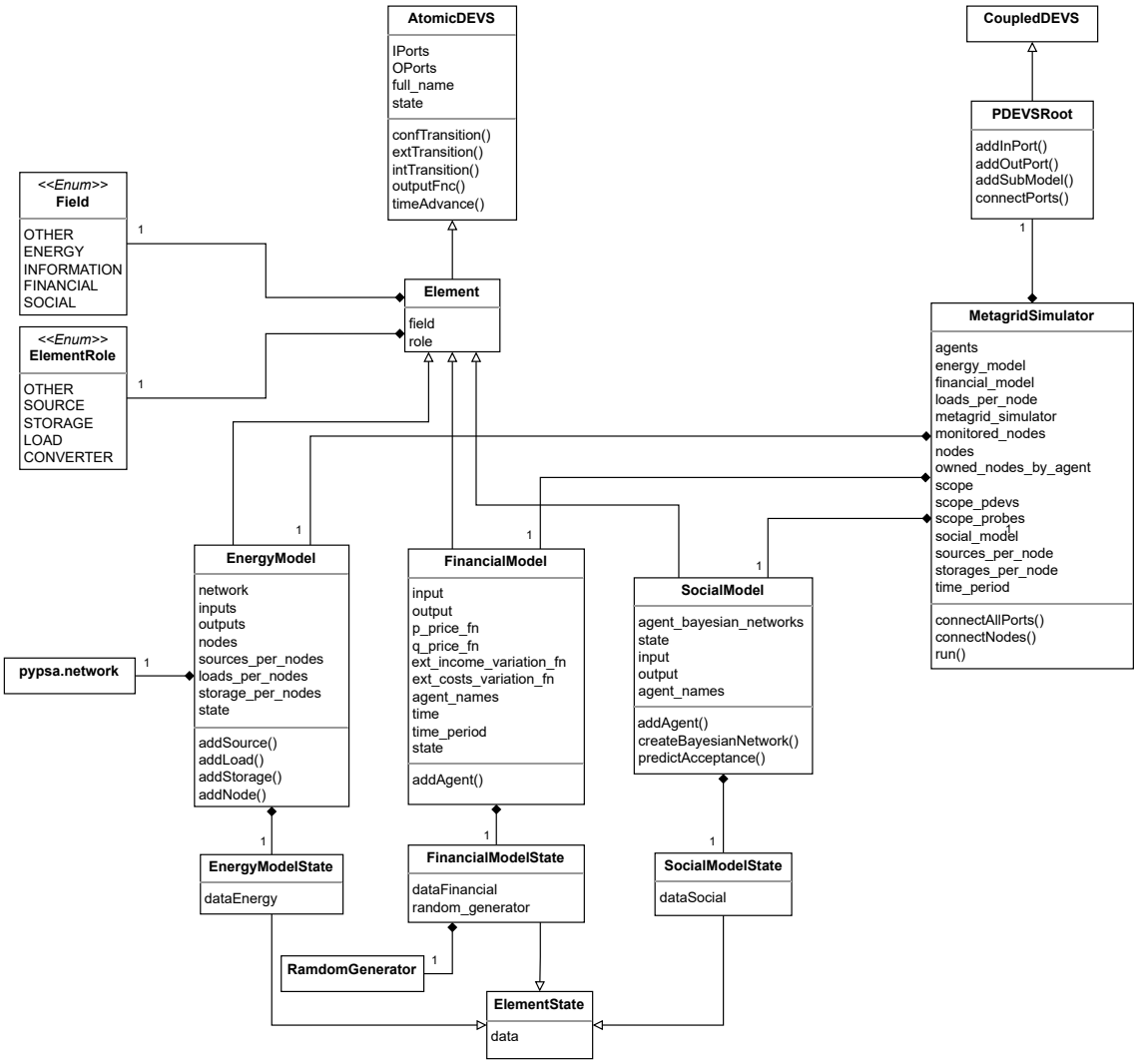


Figure 4.23: Proposed co-simulation model

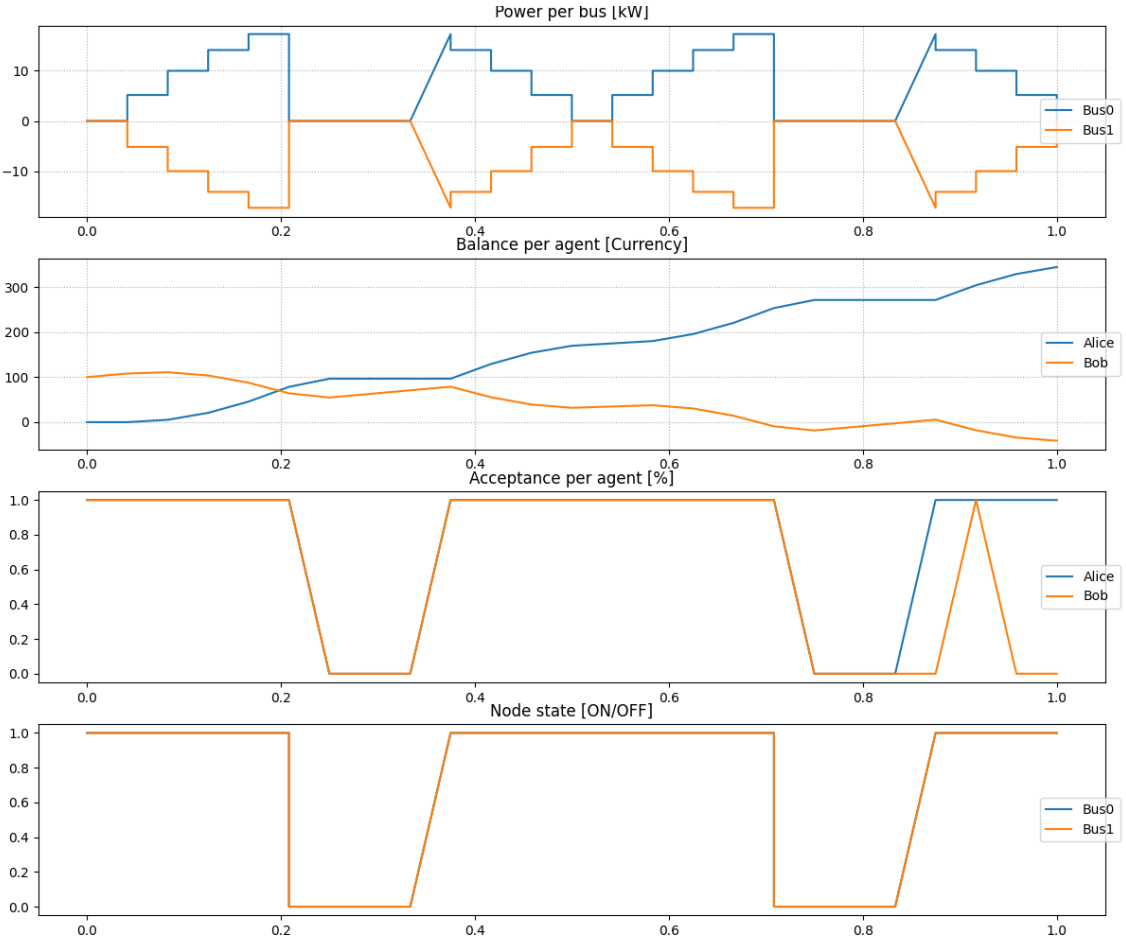


Figure 4.24: Simulation of a simple microgrid between two user

The simulator was tested by modeling a simple microgrid system, comprising of two entities, Alice and Bob. Alice had a controllable energy source with a maximum power capacity of 15 kW, while Bob was the energy consumer, operating a load at his residence. The electrical network in the simulation was straightforward, with two buses, where bus zero was connected to Alice and her energy source, and bus one was connected to Bob and his load. The result of the simulation of this microgrid is depicted in figure 4.24. The model developed in this example is available for download in https://gitlab.laas.fr/metagrid/metagrid-simulator/-/tree/simu_simple

In terms of finance, Bob started the simulation with a bank account balance of 200, while Alice started at zero. Bob was assigned a small salary that was regularly added to his account, while Alice could only earn money from Bob's energy consumption payments.

The simulation showed that energy production and consumption were closely matched. In instances where Bob's load tried to consume more than 15 kW, it was offloaded from the network, and consumption ceased until it requested less than 15 kW from Alice's source.

In terms of monetary exchanges, Alice's bank account gradually increased as Bob consumed energy, and remained stagnant when there was no consumption. On the other hand, Bob's bank account decreased as he consumed more energy, with a more pronounced decrease as his consumption increased. When Bob's load was offloaded from the network, his bank account gradually filled up due to his salary, and when Alice did not receive payments, her account remained stagnant.

The simulation also indicated that towards the end of the simulation, Alice's acceptance of energy dropped to zero as Bob's negative bank account impacted his acceptance of the energy as too expensive at that time. The simulation results showed that Bob's and Alice's bank accounts followed each other almost perfectly, with Alice's account stagnant when Bob was offloaded, and Bob's account filling up with his salary when offloaded.

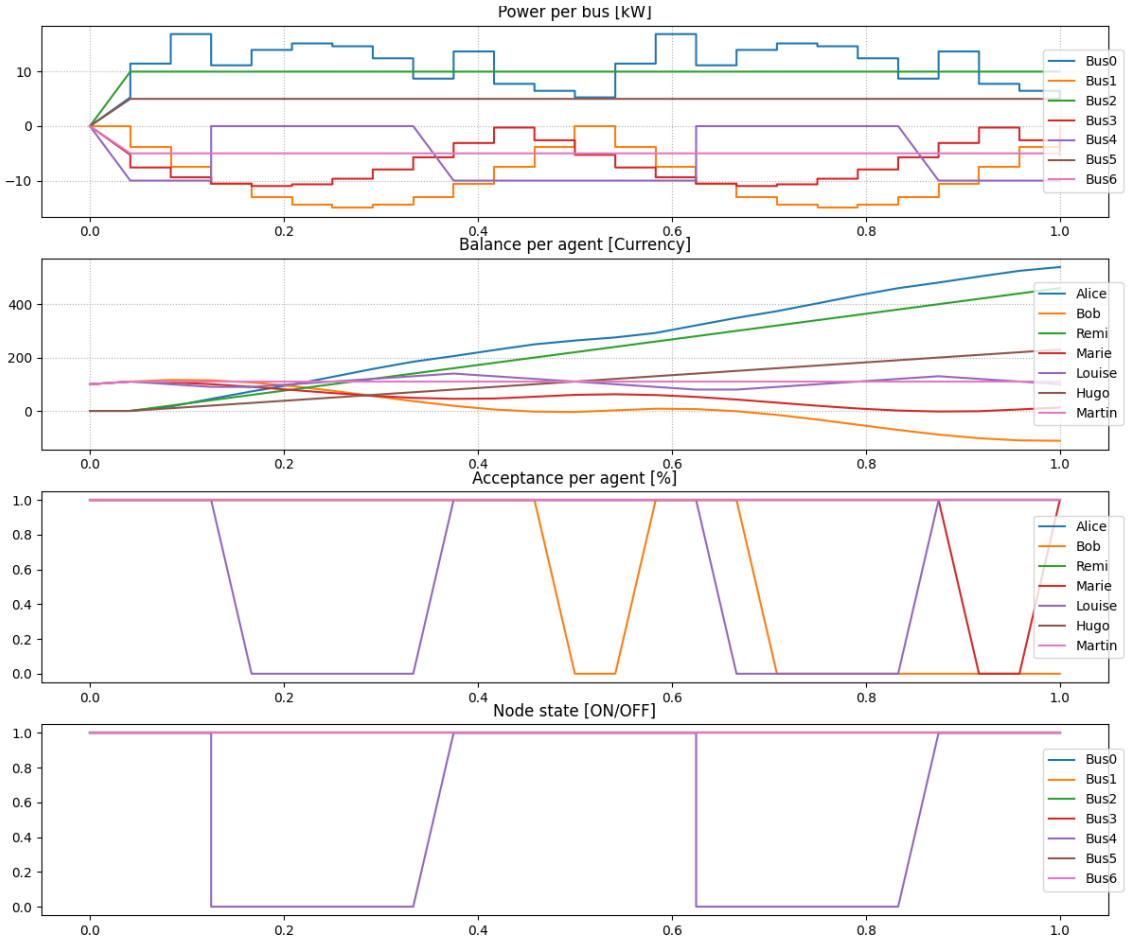


Figure 4.25: Simulation of a More complex microgrid with 7 users

In the field of microgrid simulation, scalability is of utmost importance in order to effectively model and analyze the behavior of a system under various conditions. A scalable simulation model is capable of accommodating the increasing number of agents and components in the system, thereby providing a comprehensive and accurate representation of the system. The model developed in this example is available for download in https://gitlab.laas.fr/metagrid/metagrid-simulator/-/tree/simu_complex

The present simulation depicted in figure 4.25 focuses on demonstrating the scalability of an executable model. The simulation scenario involves 7 different agents consisting of both producers and consumers, all connected to a central bus which enables energy exchange. There are three different energy sources in bus0, bus2, and bus5, with the first source being variable and the others being constant. The other four buses consist of loads with dynamics that are either constant or sinusoidal.

The shedding function, which disconnects the bus when there is an excessive energy demand from consumers, is observed to always work as expected. The acceptance of Louise, which is linked to the bus, is noted to vary correctly in response to changes in the system. Additionally, the results of the simulation demonstrate that energy price continues to play a key role in influencing the acceptance of the agents.

The results of this simulation are highly promising for the development of this executable model. The model's ability to easily accommodate the increasing number of agents and components in the system, as compared to the second model, highlights the value of scalability in simulation models. This is particularly important in microgrid systems, where the behavior of the system can change rapidly in response to changes in energy demand and production, making the need for a scalable model even more crucial.

Discussion

In conclusion, the proposed model leverages PyPDEVS to perform a systemic simulation of microgrids by linking three separate models. This executable model demonstrates a capacity for handling increased component complexity in microgrid modeling, presenting a pathway for future improvements. In particular, incorporating information pertaining to microgrids and establishing connections between relevant domains could enhance the representation of logic and dynamics introduced in the diagnostic model.

One of the primary challenges in the development of systemic models is the coupling of flows and exchanges. The preceding models addressed this issue through a multi-component approach, as illustrated in Figure 3.4 on page 48, resulting in the generation of the coupling generative model (Figure 3.30). Although useful, the generative model was found to be challenging to use and expand.

The co-simulation approach, on the other hand, reduces complexity by utilizing network-level coupling and abstracting lower-level models, thus avoiding state-level coupling between fields. While this approach has several benefits, it necessitates the use of a more sophisticated simulator, as will be discussed in the subsequent chapter.

In summary, the proposed model represents a simple yet effective solution for systemic simulation of microgrids, offering opportunities for improvement through the incorporation of information and links between relevant domains. The co-simulation approach reduces complexity through network-level coupling, though it requires the use of a more advanced simulator.

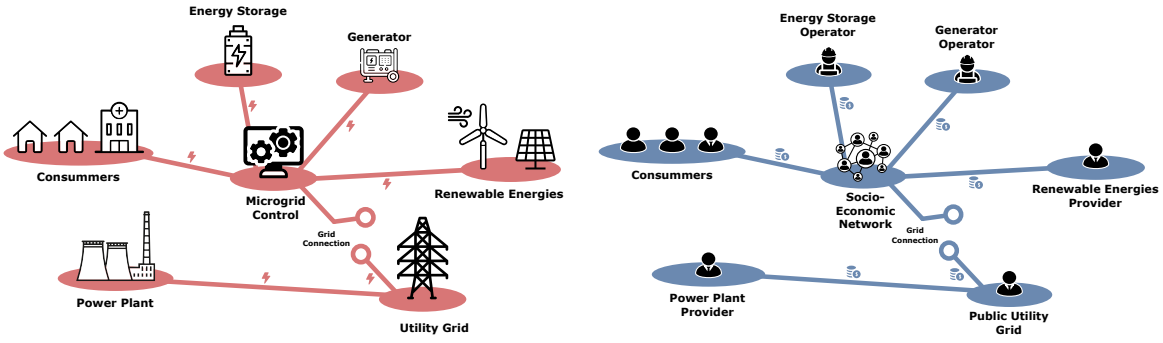
Conclusion and Future Work

Conclusions

In order to elucidate the reasons for the failure of microgrids, we have chosen first to formally understand the functioning of microgrids at the level of their internal structure. This work has made it possible to put the formal definition of modeling and simulation back at the center of the study of microgrids. A significant amount of popularisation and formalization work has been carried out to combine Ziegler's theory of modeling and simulation with a systemic study of the functioning of microgrids. Through this manuscript, we have provided the first building blocks of understanding to clearly explain a microgrid system and how it works in its internal structure. It is posited in this we that this effort of formalization is an essential step in understanding and studying the failures of these fragile systems.

The manuscript has walked the reader through the formalism of model building and simulation developed from Ziegler's theory of modeling and simulation. The four key points of this method are: choosing a source system, specifying an experimental frame, building a model, and developing a simulator. These points have been the guiding thread throughout the work.

The first chapter provided an understanding of what a source system is at a formal level and gave the technical requirements to describe this system clearly. It showed the importance of the source data to know how a source system works. As explained in this chapter, our study system is extremely complex and diverse, interconnecting many domains in a powerful way, which makes understanding and analyzing the data more difficult. The methods for dealing with these problems are varied, sometimes relying on energy aspects depicted in figure 4.26(a), sometimes on community aspects depicted in figure 4.26(b), and sometimes partly on both. However, it is clear that the system needs long-term sustainability, which has led the literature to study and propose solutions to improve the resilience and sustainability of microgrids. Combining this problem with the definition of a source system, we concluded that this system needed a framed systemic formalism for its study.



(a) The energy microgrid

(b) The community microgrid

Figure 4.26: The two visions of microgrid in the literature

In the second chapter, we established an experimental framework for studying microgrids. Starting from a divergence visible in the scientific literature, we identified a deficit point for the understanding of this source system: two opposing visions, that of an energy system requiring

many technical and technological feats to provide quality energy permanently and sustainably and that of a community system connecting people and considering energy as a mean and a necessity rather than as a complex balance of electron flows. To combine these two visions, we identified four main fields from the literature imaged in figure 4.27: energy, information, financial and social.

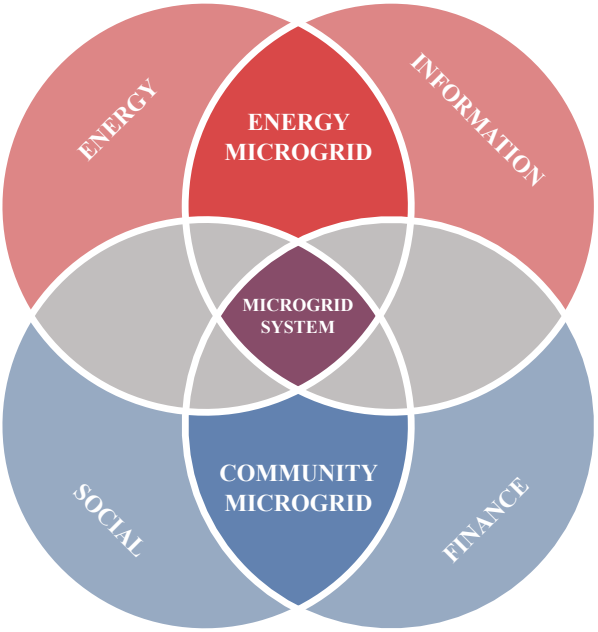


Figure 4.27: The experimental frames related to the intersection of *energy* and *community* microgrids

In order to facilitate our transition to the modeling of our source system, we sought to identify a form of morphism in the operation and definition of these domains. This led to the identification of three components present in each of the domains: several elements that make them up, a particular architecture linking these elements, and major problems that may hinder their long-term sustainability summarized in figure 4.28.

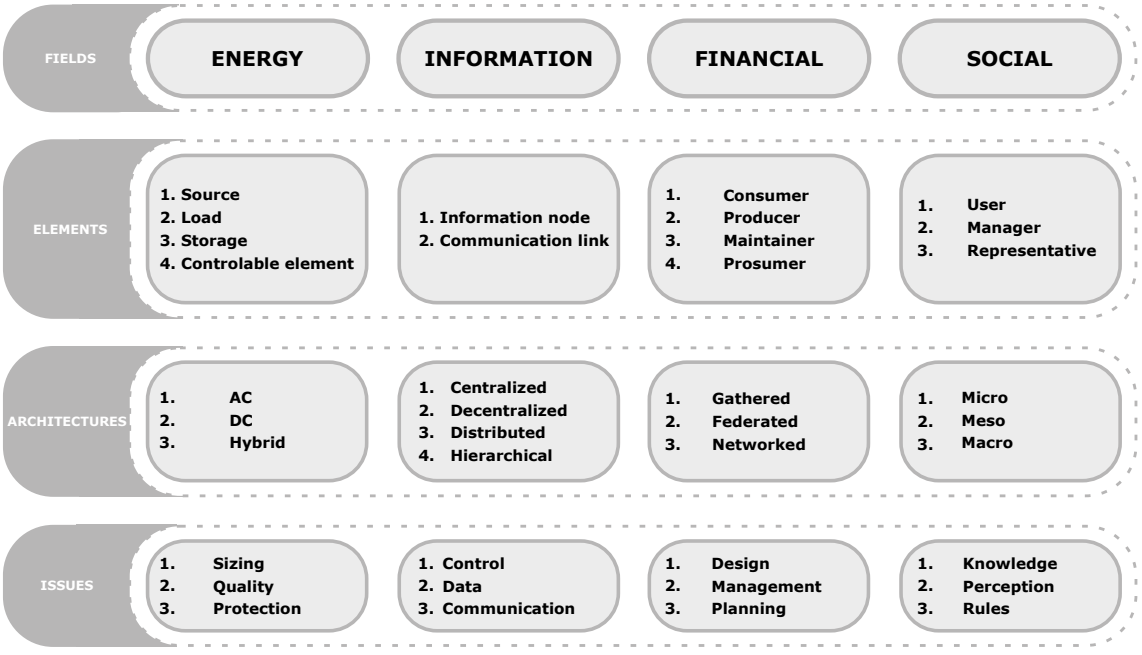


Figure 4.28: Microgrid systemic homomorphic review

This work of literature review and domain organization rigorously frames our experimental framework in order to approach the third stage of microgrid formalization in chapter 3. Given the lack of systemic data on microgrids, we have had to approach the modeling of this system from another angle, moving from a structural model to a generative model and then to a replicative model as our understanding of the theory of modeling and simulation applied systemically to microgrids grew.

The structural model imaged in figure 4.29 gives us a clear understanding of the importance of identifying and separating these four main fields of microgrids. Through this interplay of balances and interactions between these domains, we can easily understand the interest in formalizing them more strictly. This also provide an innovative vision for rating the health and sustainability of a microgrid.

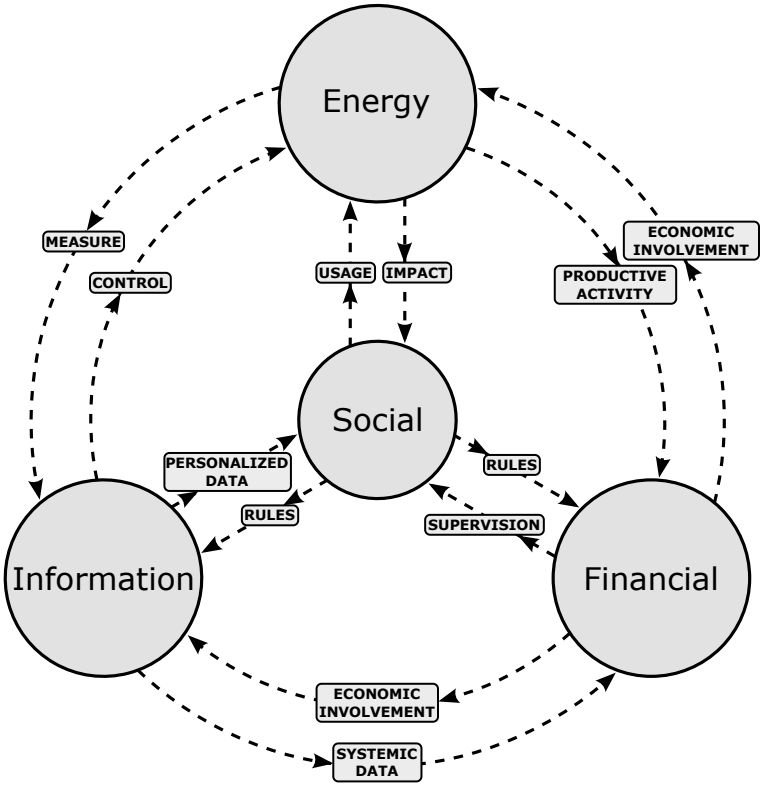


Figure 4.29: The structural graphical model

The generative model imaged in figure 4.30 is an essential step in this process of understanding the theory of modeling and simulation applied to a systemic view of microgrids. In trying to go deeper into the formalization of all the elements present in a microgrid, we came up against the complexity of these systems. However, it is an effective way to understand the internal mechanisms and complex dynamics of microgrids.

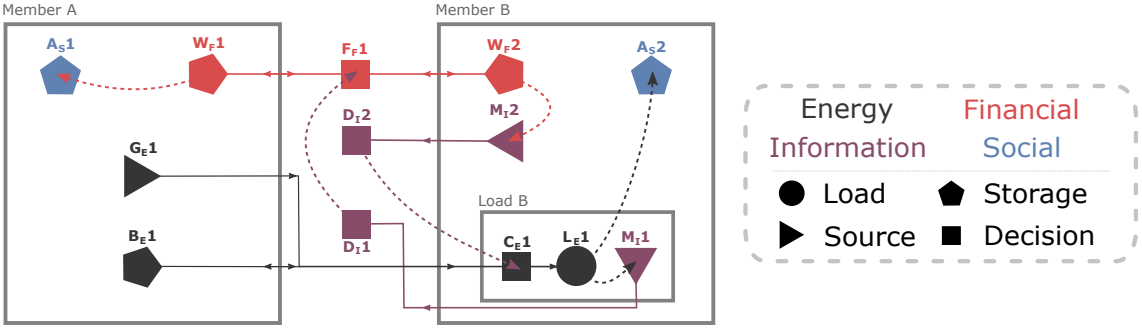


Figure 4.30: The generative graphical model

The replicative model imaged in figure 4.31 is a consequence of this. It represents the systemic understanding of our experimental framework related to the sustainability of microgrids, the attempt to link the equilibrium of its different domains, and the representation of the whole internal complexity of each domain. By adopting a co-simulation methodology, we can address the interconnections between the four domains while maintaining an understandable, intelligible structure conducive to multidisciplinary work and exchange.

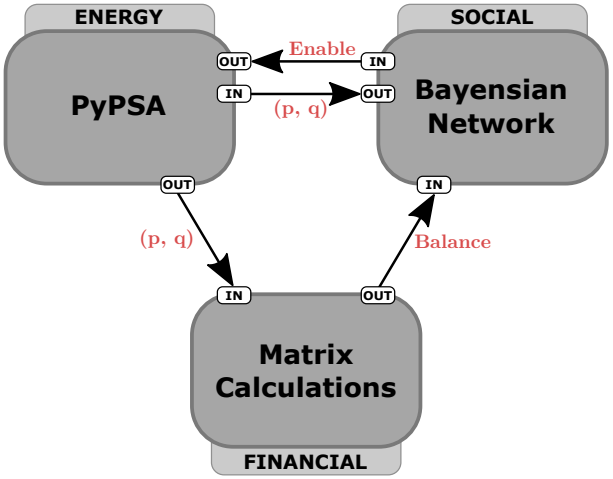


Figure 4.31: The replicative graphical model

Chapter four discusses the transformation of our models into executable models that a simulator can understand and the concept of simulation. Simulators can take many forms, ranging from a simple spreadsheet to a formal application of modeling and simulation theory through a DEVS simulation kernel. It is essential to understand their formalization in order to execute our developed model properly. This is how we have proceeded with our microgrid modeling approach, using an iterative method between each model and its simulation to provide an additional layer of abstraction as presented in figure 4.32.

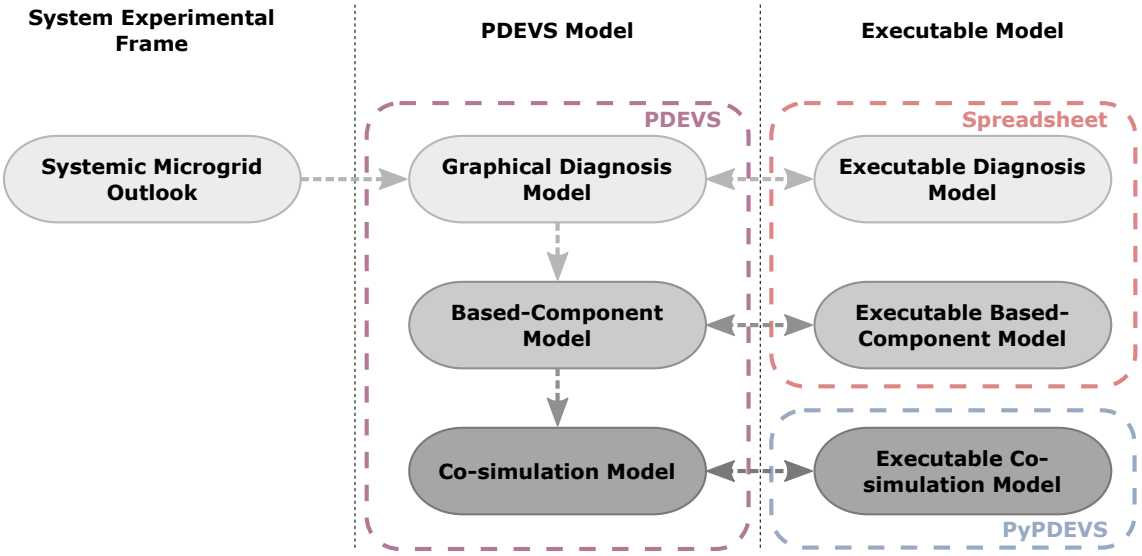


Figure 4.32: Evolution of the modelling approach during the PhD

This model-building process and innovative approach to the study of microgrids are the essential aspects to be retained from this work. From identifying the different key domains of microgrids to the formalization of their interconnection, we have provided a solid framework for modeling these systems. By rigorously following the framework defined by Ziegler's theory of modeling and

simulation, we have provided a more formal understanding of the coupling of the different existing models. This exploratory work provides a first approach for modeling microgrids, hoping to inspire many researchers to deepen the first elements of abstraction.

Future Work

This thesis paves the way to multiple research topics and tasks.

It is clear that the first task to be developed and continued (although not the simplest) is to finalize the interconnection of the four domains in the co-simulation model. Firstly, it will be necessary to add an information model to handle the control and data management aspects, including the complexities of these domains. Secondly, it will be essential to develop each model to add a large number of inputs and outputs that will be used for interconnection in the co-simulation. The goal here is to build a systemic co-simulation tool, as depicted in Figure 4.33 to simulate the diversity of sustainability indicators.

To take the model further, once it is fully formalised, we will be able to confront it with real case studies to identify the internal problems of microgrids that may affect their sustainability. A field study will have to be carried out to validate and improve the systemic notation of the microgrid sustainability factors. An attempt to formalize the rating of these factors was made during this work using a generic questionnaire developed. This questionnaire can be used as a basis for evaluating and understanding a real case study or for monitoring the health of a microgrid over time to assess its evolution.

Once this model and its simulation reach a sufficient degree of formalization and validation to represent the four domains and their interconnections, research can be carried out focusing on microgrid sustainability issues. With more accurate experience data and a complete model, the simulation can be subjected to various scenarios to accurately identify the weak points of the system or the chains of internal events leading to its failure. Given the difficulty of collecting systematic data on a microgrid, hybrid experiments can also be carried out by combining physical components, such as a power grid, with simulated components, such as the social domain. Thanks to the ability of the co-simulation model to integrate connected domains regardless of their temporal pattern, this type of hybrid simulation can be carried out even closer to reality.

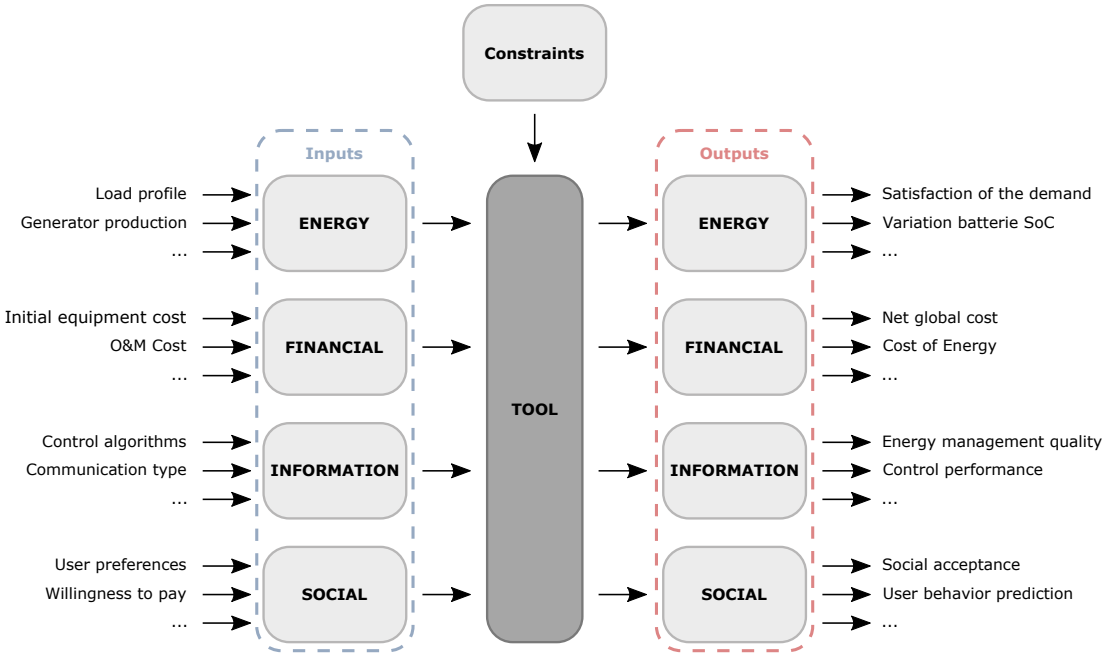


Figure 4.33: Systemic co-simulation tool adapted from [4]

On the other hand, although the use of the PDEVS formalism has been largely justified in this work and the final PyPDEVS simulator is a consequence of it, one aspect of the interest of

this formalism has been little developed in this work: the dynamic structure modeling of systems. This is an approach to the simulation of complex systems that allows the evolution of system processes over time to be modeled, including the creation and deletion of system components, and modification of their behavior. In the system studied, a process fundamental to understanding the sustainability of microgrids deserves to be better understood: expansion.

The expansion of microgrids is a poorly understood and studied process, yet it is part of the normal evolution of the system. Since these systems are extremely small and often designed with a bottom-up approach, expansion is part of the daily life of microgrids. In order to explain in detail the scientific considerations of this thesis, it is essential to know that the study of microgrid expansion is at the heart of the work done.

A first intuition regarding the sustainability of microgrids was that, as these systems are subject to almost daily expansion with new users, increasing energy demand, new local investments, or the installation of new and more efficient generators, their daily instability makes it difficult to visualize and perceive their current state of health, which may inevitably contribute toward their failure. Therefore, we started to look for study cases on formalizing the functioning of microgrids to be able to model and study the expansion process within these models. Nevertheless, as we have seen before, the expansion can affect several fields of the microgrid and sometimes several fields simultaneously with interconnected impacts, which are difficult to define. This is why we need a model based on an experimental systems approach, considering all related domains within microgrids. While studying the complexity of the expansion, we realized that there need to be more microgrid modeling in the literature. Therefore, we approached the problem differently by building this experimental framework, better defining the source system, and proposing several system models. Unfortunately, due to the extensive work required to develop and explore this first step, we could not address the modeling of microgrid expansion, which requires an additional step to integrate dynamic modeling and simulation formalisms.

We hope this exploratory work can resonate with other work, inspire other scientists to apply this method to their system of study, encourage other researchers to evolve systemic microgrid modeling and co-simulation and help many research fields work together around a strict formalism without sacrificing their domain-specific foundations.

References

[1] USAID, "Scaling up commercial off grid solutions in senegal," *USAID*.

[2] "Mini-grid policy toolkit : Policy and business framework for successful minigrid roll-outs." Accessed: 2022-07-26.

[3] M. F. Zia, E. Elbouchikhi, and M. Benbouzid, "Microgrids energy management systems: A critical review on methods, solutions, and prospects," *Applied Energy*, vol. 222, pp. 1033–1055, July 2018.

[4] M. A. Cuesta, T. Castillo-Calzadilla, and C. E. Borges, "A critical analysis on hybrid renewable energy modeling tools: An emerging opportunity to include social indicators to optimise systems in small communities," *Renewable and Sustainable Energy Reviews*, vol. 122, p. 109691, Apr. 2020.

[5] B. P. Zeigler, A. Muzy, and E. Kofman, *Theory of modeling and simulation: discrete event & iterative system computational foundations*. Academic press, 2018.

[6] S. Sen and V. Kumar, "Microgrid control: A comprehensive survey," *Annual Reviews in Control*, vol. 45, pp. 118–151, Jan. 2018.

[7] P. Upham, C. Oltra, and [U+FFFD] Boso, "Towards a cross-paradigmatic framework of the social acceptance of energy systems," *Energy Research & Social Science*, vol. 8, pp. 100–112, July 2015.

[8] S. Auer, J. Liße, S. R. Mandha, and C. Horn, "Power-flow-constrained asset optimization for off-grid power systems using selected open-source frameworks," in *Proceedings of the 4th International Hybrid Power Systems Workshop, Crete, Greece*, pp. 22–23, 2019.

[9] G. J. Klir, "Architecture of systems complexity," *Saunders, New York*, 1985.

[10] U. Nations, "Sustainable un. development goal, 7," 2019.

[11] W. Bank, "More People Have Access to Electricity Than Ever Before, but World Is Falling Short of Sustainable Energy Goals," 2019.

[12] M. Kanagawa and T. Nakata, "Assessment of access to electricity and the socio-economic impacts in rural areas of developing countries," *Energy Policy*, vol. 36, pp. 2016–2029, June 2008.

[13] "The evolution of distributedenergy resources," *Microgrid Knowledge*, 2020.

[14] A. Chatterjee, D. Burmester, A. Brent, and R. Rayudu, "Research Insights and Knowledge Headways for Developing Remote, Off-Grid Microgrids in Developing Countries," *Energies*, vol. 12, p. 2008, Jan. 2019.

[15] A. G. Tsikalakis and N. D. Hatziargyriou, "Centralized control for optimizing microgrids operation," in *2011 IEEE Power and Energy Society General Meeting*, pp. 1–8, July 2011. ISSN: 1944-9925.

[16] E. M. Gui, M. Diesendorf, and I. MacGill, "Distributed energy infrastructure paradigm: Community microgrids in a new institutional economics context," *Renewable and Sustainable Energy Reviews*, vol. 72, pp. 1355–1365, May 2017.

[17] N. Narayan, V. Vega-Garita, Z. Qin, J. Popovic-Gerber, P. Bauer, and M. Zeman, "The Long Road to Universal Electrification: A Critical Look at Present Pathways and Challenges," *Energies*, vol. 13, p. 508, Jan. 2020.

[18] R. Hanna, M. Ghonima, J. Kleissl, G. Tynan, and D. G. Victor, "Evaluating business models for microgrids: Interactions of technology and policy," *Energy Policy*, vol. 103, pp. 47–61, Apr. 2017.

[19] I. Gunnarsdottir, B. Davidsdottir, E. Worrell, and S. Sigurgeirsdottir, "Sustainable energy development: History of the concept and emerging themes," *Renewable and Sustainable Energy Reviews*, vol. 141, p. 110770, May 2021.

[20] M. S. Shah Danish, T. Senjyu, T. Funabashia, M. Ahmadi, A. M. Ibrahimi, R. Ohta, H. O. Rashid Howlader, H. Zaheb, N. R. Sabory, and M. M. Sediqi, "A sustainable microgrid: A sustainability and management-oriented approach," *Energy Procedia*, vol. 159, pp. 160–167, Feb. 2019.

[21] E. E. Gaona, C. L. Trujillo, and J. A. Guacaneme, "Rural microgrids and its potential application in Colombia," *Renewable and Sustainable Energy Reviews*, vol. 51, pp. 125–137, Nov. 2015.

[22] F. Martin-Martínez, A. Sánchez-Miralles, and M. Rivier, "A literature review of Microgrids: A functional layer based classification," *Renewable and Sustainable Energy Reviews*, vol. 62, pp. 1133–1153, Sept. 2016.

[23] J. Cust, A. Singh, and K. Neuhoff, "Rural Electrification in India: Economic and Institutional Aspects of Renewables," Tech. Rep. ID 2760810, Rochester, NY, Dec. 2007.

[24] D. Schnitzer, D. S. Lounsbury, J. P. Carvallo, R. Deshmukh, J. Apt, and D. M. Kammen, "Microgrids for rural electrification: A critical review of best practices based on seven case studies," *United Nations Foundation*, 2014.

[25] F. Almeshqab and T. S. Ustun, "Lessons learned from rural electrification initiatives in developing countries: Insights for technical, social, financial and public policy aspects," *Renewable and Sustainable Energy Reviews*, vol. 102, pp. 35–53, Mar. 2019.

[26] E. Hartvigsson, *To Be or Not to Be: On System Dynamics and the Viability of Mini-Grids in Rural Electrification.* Chalmers Tekniska Hogskola (Sweden), 2018.

[27] K. Ubilla, G. A. Jiménez-Estévez, R. Hernádez, L. Reyes-Chamorro, C. Hernández Irigoyen, B. Severino, and R. Palma-Behnke, "Smart Microgrids as a Solution for Rural Electrification: Ensuring Long-Term Sustainability Through Cadastre and Business Models," *IEEE Transactions on Sustainable Energy*, vol. 5, pp. 1310–1318, Oct. 2014.

[28] M. E. Khodayar, "Rural electrification and expansion planning of off-grid microgrids," *The Electricity Journal*, vol. 30, no. 4, pp. 68–74, 2017.

[29] A. Q. Santos, Z. Ma, C. G. Olsen, and B. N. Jørgensen, "Framework for Microgrid Design Using Social, Economic, and Technical Analysis," *Energies*, vol. 11, p. 2832, Oct. 2018.

[30] H. Suk, A. Yadav, and J. Hall, "Scalability considerations in the design of microgrids to support socioeconomic development in rural communities," in *ASME International Mechanical Engineering Congress and Exposition*, vol. 52187, p. V013T05A068, American Society of Mechanical Engineers, 2018.

[31] A. Boche, C. Foucher, and L. F. L. Villa, "Understanding Microgrid Sustainability: A Systemic and Comprehensive Review," *Energies*, vol. 15, p. 2906, Jan. 2022.

[32] S. Sen and V. Kumar, "Microgrid modelling: A comprehensive survey," *Annual Reviews in Control*, vol. 46, pp. 216–250, Jan. 2018.

[33] D. Akinyele, J. Belikov, and Y. Levron, "Challenges of Microgrids in Remote Communities: A STEEP Model Application," *Energies*, vol. 11, p. 432, Feb. 2018.

[34] M. Carpintero-Rentería, D. Santos-Martín, and J. M. Guerrero, "Microgrids Literature Review through a Layers Structure," *Energies*, vol. 12, p. 4381, Jan. 2019.

[35] T. Sachs, A. Gründler, M. Rusic, and G. Fridgen, "Framing Microgrid Design from a Business and Information Systems Engineering Perspective," *Business & Information Systems Engineering*, vol. 61, pp. 729–744, Dec. 2019.

[36] S. Alrashed, "Key performance indicators for Smart Campus and Microgrid," *Sustainable Cities and Society*, vol. 60, p. 102264, Sept. 2020.

[37] C. Rahmann, O. Núñez, F. Valencia, S. Arrechea, J. Sager, and D. Kammen, "Methodology for Monitoring Sustainable Development of Isolated Microgrids in Rural Communities," *Sustainability*, vol. 8, p. 1163, Nov. 2016.

[38] J. González-Jiménez, J. I. Asencio-Yace, M. Pérez-Lugo, and A. Irizarry-Rivera, "Compound index: Reliability, resilience, and social forces for the sustainability of isolated community microgrids (icmgs) after catastrophic weather events (cwe)," 2017.

[39] J. Hicks and N. Ison, "An exploration of the boundaries of 'community' in community renewable energy projects: Navigating between motivations and context," *Energy Policy*, vol. 113, pp. 523–534, Feb. 2018.

[40] A. Hirsch, Y. Parag, and J. Guerrero, "Microgrids: A review of technologies, key drivers, and outstanding issues," *Renewable and Sustainable Energy Reviews*, vol. 90, pp. 402–411, July 2018.

[41] L. Mariam, M. Basu, and M. F. Conlon, "Microgrid: Architecture, policy and future trends," *Renewable and Sustainable Energy Reviews*, vol. 64, pp. 477–489, Oct. 2016.

[42] R. Bayindir, E. Hossain, E. Kabalci, and R. Perez, "A Comprehensive Study on Microgrid Technology," *International Journal of Renewable Energy Research (IJRER)*, vol. 4, pp. 1094–1107, Dec. 2014.

[43] M. F. Akorede, H. Hizam, and E. Pouresmaeil, "Distributed energy resources and benefits to the environment," *Renewable and Sustainable Energy Reviews*, vol. 14, pp. 724–734, Feb. 2010.

[44] M. Gunasekaran, H. Mohamed Ismail, B. Chokkalingam, L. Mihet-Popa, and S. Padmanaban, "Energy Management Strategy for Rural Communities' DC Micro Grid Power System Structure with Maximum Penetration of Renewable Energy Sources," *Applied Sciences*, vol. 8, p. 585, Apr. 2018.

[45] H. F. Habib, C. R. Lashway, and O. A. Mohammed, "A Review of Communication Failure Impacts on Adaptive Microgrid Protection Schemes and the Use of Energy Storage as a Contingency," *IEEE Transactions on Industry Applications*, vol. 54, pp. 1194–1207, Mar. 2018.

[46] S. Hajiaghasi, A. Salemnia, and M. Hamzeh, "Hybrid energy storage system for microgrids applications: A review," *Journal of Energy Storage*, vol. 21, pp. 543–570, 2019.

[47] S. Parhizi, H. Lotfi, A. Khodaei, and S. Bahramirad, "State of the art in research on microgrids: A review," *Ieee Access*, vol. 3, pp. 890–925, 2015.

[48] R. Takalani and B. Bekker, "Load and load growth models for rural microgrids, and how to future-proof designs," in *2020 International SAUPEC/RobMech/PRASA Conference*, pp. 1–6, Jan. 2020.

[49] X. Wang, J. M. Guerrero, F. Blaabjerg, and Z. Chen, "A Review of Power Electronics Based Microgrids," *International Journal of Power Electronics*, vol. 12, no. 1, pp. 181–192, 2012.

[50] Z. A. Arfeen, A. B. Khairuddin, R. M. Larik, and M. S. Saeed, "Control of distributed generation systems for microgrid applications: A technological review," *International Transactions on Electrical Energy Systems*, vol. 29, no. 9, p. e12072, 2019.

[51] K. A. Jadav, H. M. Karkar, and I. N. Trivedi, "A Review of Microgrid Architectures and Control Strategy," *Journal of The Institution of Engineers (India): Series B*, vol. 98, pp. 591–598, Dec. 2017.

[52] A. M. R. Lede, M. G. Molina, M. Martinez, and P. E. Mercado, "Microgrid architectures for distributed generation: A brief review," in *2017 IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America)*, pp. 1–6, Sept. 2017.

[53] J. J. Justo, F. Mwasilu, J. Lee, and J.-W. Jung, "AC-microgrids versus DC-microgrids with distributed energy resources: A review," *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 387–405, Aug. 2013.

[54] T. Dragicevic, X. Lu, J. C. Vasquez, and J. M. Guerrero, "DC Microgrids—Part II: A Review of Power Architectures, Applications, and Standardization Issues," *IEEE Transactions on Power Electronics*, vol. 31, pp. 3528–3549, May 2016.

[55] A. Gupta, S. Doolla, and K. Chatterjee, "Hybrid AC–DC Microgrid: Systematic Evaluation of Control Strategies," *IEEE Transactions on Smart Grid*, vol. 9, pp. 3830–3843, July 2018.

[56] L. Petrucci, C. Boccaletti, B. Francois, and P. Di Felice, "Hybrid trigeneration system management with a double DC-bus configuration on the electrical side," in *2009 8th International Symposium on Advanced Electromechanical Motion Systems Electric Drives Joint Symposium*, pp. 1–6, July 2009.

[57] X. She, S. Lukic, and Q. H. Alex, "DC zonal micro-grid architecture and control," in *IECON 2010 - 36th Annual Conference on IEEE Industrial Electronics Society*, pp. 2988–2993, Nov. 2010. ISSN: 1553-572X.

[58] H. De Silva, D. Jayamaha, and N. Lidula, "Review on design and control of solid state transformer based microgrids," *AIMS Energy*, vol. 7, no. 6, pp. 901–923, 2019.

[59] M. Chalbi and G. Boukettaya, "Review on Solid State Transformer Based on Microgrids Architectures," in *2021 18th International Multi-Conference on Systems, Signals Devices (SSD)*, pp. 1120–1126, Mar. 2021. ISSN: 2474-0446.

[60] I. Patrao, E. Figueres, G. Garcerá, and R. González-Medina, "Microgrid architectures for low voltage distributed generation," *Renewable and Sustainable Energy Reviews*, vol. 43, pp. 415–424, 2015.

[61] E. Bullich-Massagué, F. Díaz-González, M. Aragüés-Peñalba, F. Girbau-Llistuella, P. Olivella-Rosell, and A. Sumper, "Microgrid clustering architectures," *Applied Energy*, vol. 212, pp. 340–361, Feb. 2018.

[62] S. Groh, D. Philipp, B. E. Lasch, and H. Kirchhoff, "Swarm Electrification: Investigating a Paradigm Shift Through the Building of Microgrids Bottom-up," in *Decentralized Solutions for Developing Economies* (S. Groh, J. van der Straeten, B. Edlefsen Lasch, D. Gershenson, W. Leal Filho, and D. M. Kammen, eds.), Springer Proceedings in Energy, (Cham), pp. 3–22, Springer International Publishing, 2015.

[63] V. Motjoadi, P. N. Bokoro, and M. O. Onibonoje, "A Review of Microgrid-Based Approach to Rural Electrification in South Africa: Architecture and Policy Framework," *Energies*, vol. 13, p. 2193, Jan. 2020.

[64] S. Sarangi, B. K. Sahu, and P. K. Rout, "Review of distributed generator integrated AC microgrid protection: issues, strategies, and future trends," *International Journal of Energy Research*, vol. 45, no. 10, pp. 14117–14144, 2021.

[65] S. Sarangi, B. K. Sahu, and P. K. Rout, "A comprehensive review of distribution generation integrated DC microgrid protection: issues, strategies, and future direction," *International Journal of Energy Research*, vol. 45, no. 4, pp. 5006–5031, 2021.

[66] K. Moslehi and R. Kumar, "Smart grid-a reliability perspective," in *2010 Innovative Smart Grid Technologies (ISGT)*, pp. 1–8, IEEE, 2010.

[67] K. Moslehi and R. Kumar, "A Reliability Perspective of the Smart Grid," *IEEE Transactions on Smart Grid*, vol. 1, pp. 57–64, June 2010.

[68] J. R. Aguero, E. Takayesu, D. Novosel, and R. Masiello, "Modernizing the Grid: Challenges and Opportunities for a Sustainable Future," *IEEE Power and Energy Magazine*, vol. 15, pp. 74–83, May 2017.

[69] C. Marnay, S. Chatzivasileiadis, C. Abbey, R. Iravani, G. Joos, P. Lombardi, P. Mancarella, and J. von Appen, "Microgrid Evolution Roadmap," in *2015 International Symposium on Smart Electric Distribution Systems and Technologies (EDST)*, pp. 139–144, Sept. 2015.

[70] F. S. Al-Ismail, "DC Microgrid Planning, Operation, and Control: A Comprehensive Review," *IEEE Access*, vol. 9, pp. 36154–36172, 2021.

[71] M. Sandelic, S. Peyghami, A. Sangwongwanich, and F. Blaabjerg, "Reliability aspects in microgrid design and planning: Status and power electronics-induced challenges," *Renewable and Sustainable Energy Reviews*, vol. 159, p. 112127, May 2022.

[72] S. Chandak and P. K. Rout, "The implementation framework of a microgrid: A review," *International Journal of Energy Research*, vol. 45, no. 3, pp. 3523–3547, 2021.

[73] M. Garba, M. A. Tankari, and G. Lefebvre, "Using of distributed energy ressources for microgrid resilience achieving," in *2017 IEEE 6th International Conference on Renewable Energy Research and Applications (ICRERA)*, pp. 659–663, 2017.

[74] A. Hussain, V.-H. Bui, and H.-M. Kim, "Microgrids as a resilience resource and strategies used by microgrids for enhancing resilience," *Applied Energy*, vol. 240, pp. 56–72, Apr. 2019.

[75] I. Alsaidan, A. Alanazi, W. Gao, H. Wu, and A. Khodaei, "State-Of-The-Art in Microgrid-Integrated Distributed Energy Storage Sizing," *Energies*, vol. 10, p. 1421, Sept. 2017.

[76] A. Jhunjhunwala and P. Kaur, "Solar Energy, dc Distribution, and Microgrids: Ensuring Quality Power in Rural India," *IEEE Electrification Magazine*, vol. 6, pp. 32–39, Dec. 2018.

[77] A.-K. K. Seppälä, *Sustainable design and implementation of renewable energy systems in rural areas*. Dissertation, Technische Universität München, München, 2022.

[78] R. K. Akikur, R. Saidur, H. W. Ping, and K. R. Ullah, "Comparative study of stand-alone and hybrid solar energy systems suitable for off-grid rural electrification: A review," *Renewable and Sustainable Energy Reviews*, vol. 27, pp. 738–752, Nov. 2013.

[79] E. L. V. Eriksson and E. M. Gray, "Optimization and integration of hybrid renewable energy hydrogen fuel cell energy systems – A critical review," *Applied Energy*, vol. 202, pp. 348–364, Sept. 2017.

[80] L. Gacitua, P. Gallegos, R. Henriquez-Auba, [U+FFFD] Lorca, M. Negrete-Pincetic, D. Olivares, A. Valenzuela, and G. Wenzel, "A comprehensive review on expansion planning: Models and tools for energy policy analysis," *Renewable and Sustainable Energy Reviews*, vol. 98, pp. 346–360, Dec. 2018.

[81] NJBPU, "New jersey board of public utilities microgrid report," 2016.

[82] D. Suri, J. Shekhar, A. Mukherjee, and A. Singh Bajaj, "Designing Microgrids for Rural Communities: A Practitioner Focused Mini-Review," in *2020 IEEE International Conference on Environment and Electrical Engineering and 2020 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I CPS Europe)*, pp. 1–6, June 2020.

[83] S. Sharma and Y. R. Sood, "Microgrids: A Review of Status, Technologies, Software Tools, and Issues in Indian Power Market," *IETE Technical Review*, vol. 0, pp. 1–22, Dec. 2020.

[84] X. Wei, X. Xiangning, and C. Pengwei, "Overview of key microgrid technologies," *International Transactions on Electrical Energy Systems*, vol. 28, no. 7, p. e2566, 2018.

[85] R. Pinto, S. Mariano, M. D. R. Calado, and J. F. De Souza, "Impact of Rural Grid-Connected Photovoltaic Generation Systems on Power Quality," *Energies*, vol. 9, p. 739, Sept. 2016.

[86] A. A. Bajwa, H. Mokhlis, S. Mekhilef, and M. Mubin, "Enhancing power system resilience leveraging microgrids: A review," *Journal of Renewable and Sustainable Energy*, vol. 11, p. 035503, May 2019.

[87] A. Hooshyar and R. Iravani, "Microgrid Protection," *Proceedings of the IEEE*, vol. 105, pp. 1332–1353, July 2017.

[88] A. Dagar, P. Gupta, and V. Niranjan, "Microgrid protection: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 149, p. 111401, Oct. 2021.

[89] K. H. Sirviö, H. Laaksonen, K. Kauhaniemi, and N. Hatziargyriou, "Evolution of the Electricity Distribution Networks—Active Management Architecture Schemes and Microgrid Control Functionalities," *Applied Sciences*, vol. 11, p. 2793, Jan. 2021.

[90] D. E. Olivares, A. Mehrizi-Sani, A. H. Etemadi, C. A. Cañizares, R. Iravani, M. Kazerani, A. H. Hajimiragha, O. Gomis-Bellmunt, M. Saeedifard, R. Palma-Behnke, *et al.*, "Trends in microgrid control," *IEEE Transactions on smart grid*, vol. 5, no. 4, pp. 1905–1919, 2014.

[91] H. Farhangi, "The path of the smart grid," *IEEE Power and Energy Magazine*, vol. 8, pp. 18–28, Jan. 2010.

[92] M. Faheem, S. B. H. Shah, R. A. Butt, B. Raza, M. Anwar, M. W. Ashraf, M. A. Ngadi, and V. C. Gungor, "Smart grid communication and information technologies in the perspective of Industry 4.0: Opportunities and challenges," *Computer Science Review*, vol. 30, pp. 1–30, Nov. 2018.

[93] X. Fang, S. Misra, G. Xue, and D. Yang, "Smart Grid — The New and Improved Power Grid: A Survey," *IEEE Communications Surveys Tutorials*, vol. 14, no. 4, pp. 944–980, 2012.

[94] R. Ma, H.-H. Chen, Y.-R. Huang, and W. Meng, "Smart Grid Communication: Its Challenges and Opportunities," *IEEE Transactions on Smart Grid*, vol. 4, pp. 36–46, Mar. 2013.

[95] M. Kuzlu and M. Pipattanasomporn, "Assessment of communication technologies and network requirements for different smart grid applications," in *2013 IEEE PES Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–6, Feb. 2013.

[96] S. Marzal, R. Salas, R. González-Medina, G. Garcerá, and E. Figueres, "Current challenges and future trends in the field of communication architectures for microgrids," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 3610–3622, Feb. 2018.

[97] I. Serban, S. Céspedes, C. Marinescu, C. A. Azurdia-Meza, J. S. Gómez, and D. S. Hueichapan, "Communication Requirements in Microgrids: A Practical Survey," *IEEE Access*, vol. 8, pp. 47694–47712, 2020.

[98] M. Saleh, Y. Esa, M. E. Hariri, and A. Mohamed, "Impact of Information and Communication Technology Limitations on Microgrid Operation," *Energies*, vol. 12, p. 2926, Jan. 2019.

[99] M. Najafzadeh, R. Ahmadiahangar, O. Husev, I. Roasto, T. Jalakas, and A. Blinov, "Recent Contributions, Future Prospects and Limitations of Interlinking Converter Control in Hybrid AC/DC Microgrids," *IEEE Access*, vol. 9, pp. 7960–7984, 2021.

[100] Z. Cheng, J. Duan, and M.-Y. Chow, "To Centralize or to Distribute: That Is the Question: A Comparison of Advanced Microgrid Management Systems," *IEEE Industrial Electronics Magazine*, vol. 12, pp. 6–24, Mar. 2018.

[101] D. Y. Yamashita, I. Vechiu, and J.-P. Gaubert, "A review of hierarchical control for building microgrids," *Renewable and Sustainable Energy Reviews*, vol. 118, p. 109523, Feb. 2020.

[102] Q. Zhou, M. Shahidehpour, A. Paaso, S. Bahramirad, A. Alabdulwahab, and A. Abusorrah, "Distributed Control and Communication Strategies in Networked Microgrids," *IEEE Communications Surveys Tutorials*, vol. 22, no. 4, pp. 2586–2633, 2020.

[103] X. Feng, A. Shekhar, F. Yang, R. E. Hebner, and P. Bauer, "Comparison of Hierarchical Control and Distributed Control for Microgrid," *Electric Power Components and Systems*, vol. 45, pp. 1043–1056, June 2017.

[104] A. Bani-Ahmed, M. Rashidi, A. Nasiri, and H. Hosseini, "Reliability Analysis of a Decentralized Microgrid Control Architecture," *IEEE Transactions on Smart Grid*, vol. 10, pp. 3910–3918, July 2019.

[105] Z. Shuai, Y. Sun, Z. J. Shen, W. Tian, C. Tu, Y. Li, and X. Yin, "Microgrid stability: Classification and a review," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 167–179, May 2016.

[106] M. S. Mahmoud, N. M. Alyazidi, and M. I. Abouheaf, "Adaptive intelligent techniques for microgrid control systems: A survey," *International Journal of Electrical Power & Energy Systems*, vol. 90, pp. 292–305, Sept. 2017.

[107] P. Basak, S. Chowdhury, S. Halder nee Dey, and S. P. Chowdhury, "A literature review on integration of distributed energy resources in the perspective of control, protection and stability of microgrid," *Renewable and Sustainable Energy Reviews*, vol. 16, pp. 5545–5556, Oct. 2012.

[108] Z. Zeng, H. Yang, R. Zhao, and C. Cheng, "Topologies and control strategies of multi-functional grid-connected inverters for power quality enhancement: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 24, pp. 223–270, Aug. 2013.

[109] M.-S. Kim, R. Haider, G.-J. Cho, C.-H. Kim, C.-Y. Won, and J.-S. Chai, "Comprehensive Review of Islanding Detection Methods for Distributed Generation Systems," *Energies*, vol. 12, p. 837, Jan. 2019.

[110] X. Shen, D. Tan, Z. Shuai, and A. Luo, "Control Techniques for Bidirectional Interlinking Converters in Hybrid Microgrids: Leveraging the advantages of both ac and dc," *IEEE Power Electronics Magazine*, vol. 6, pp. 39–47, Sept. 2019.

[111] Y. Wang, A. O. Rousis, and G. Strbac, "On microgrids and resilience: A comprehensive review on modeling and operational strategies," *Renewable and Sustainable Energy Reviews*, vol. 134, p. 110313, Dec. 2020.

[112] S. Kakran and S. Chanana, "Smart operations of smart grids integrated with distributed generation: A review," *Renewable and Sustainable Energy Reviews*, vol. 81, pp. 524–535, Jan. 2018.

[113] M. F. Roslan, M. A. Hannan, P. J. Ker, and M. N. Uddin, "Microgrid control methods toward achieving sustainable energy management," *Applied Energy*, vol. 240, pp. 583–607, Apr. 2019.

[114] O. Boqtob, H. El Moussaoui, H. El Markhi, and T. Lamhamdi, "Microgrid energy management system: a state-of-the-art review," *Journal of Electrical Systems*, vol. 15, no. 1, pp. 53–67, 2019.

[115] M. El-Hendawi, H. A. Gabbar, G. El-Saady, and E.-N. A. Ibrahim, "Control and EMS of a Grid-Connected Microgrid with Economical Analysis," *Energies*, vol. 11, p. 129, Jan. 2018.

[116] L. Wen, K. Zhou, S. Yang, and X. Lu, "Optimal load dispatch of community microgrid with deep learning based solar power and load forecasting," *Energy*, vol. 171, pp. 1053–1065, Mar. 2019.

[117] E. Yoder and N. J. Williams, "Load Profile Prediction Using Customer Characteristics," in *2020 IEEE PES/IAS PowerAfrica*, pp. 1–5, Aug. 2020.

[118] J. Ma and X. Ma, "State-of-the-art forecasting algorithms for microgrids," in *2017 23rd International Conference on Automation and Computing (ICAC)*, pp. 1–6, Sept. 2017.

[119] K. Moharm, "State of the art in big data applications in microgrid: A review," *Advanced Engineering Informatics*, vol. 42, p. 100945, Oct. 2019.

[120] Y. Zhang, T. Huang, and E. F. Bompard, "Big data analytics in smart grids: a review," *Energy Informatics*, vol. 1, p. 8, Aug. 2018.

[121] S. Huaiying, L. Yan, Z. Angang, Q. Songfeng, W. Yong, and Z. Wulei, "Research on Technical Architecture and Application of Big Data Cloud Platform for Electric Power Measurement," *Journal of Physics: Conference Series*, vol. 1213, p. 042040, June 2019.

[122] M. A. Ruiguang, W. Haiyan, Z. Quanming, and L. Yuan, "Technical Research on the Electric Power Big Data Platform of Smart Grid," *MATEC Web of Conferences*, vol. 139, p. 00217, 2017.

[123] M. A.-u.-d. Khan, M. F. Uddin, and N. Gupta, "Seven V's of Big Data understanding Big Data to extract value," in *Proceedings of the 2014 Zone 1 Conference of the American Society for Engineering Education*, pp. 1–5, Apr. 2014.

[124] K. Ye, Y. Cao, F. Xiao, J. Bai, F. Ma, and Y. Hu, "Research on unified information model for big data analysis of power grid equipment monitoring," in *2017 3rd IEEE International Conference on Computer and Communications (ICCC)*, pp. 2334–2337, Dec. 2017.

[125] J. T. C. Neto, P. H. M. de Andrade, J. M. Vilanueva, and F. A. O. Santos, "Big Data Analytics of Smart Grids using Artificial Intelligence for the Outliers Correction at Demand Measurements," in *2018 3rd International Symposium on Instrumentation Systems, Circuits and Transducers (INSCIT)*, pp. 1–6, Aug. 2018.

[126] C. A. Marino and M. Marufuzzaman, "A microgrid energy management system based on chance-constrained stochastic optimization and big data analytics," *Computers & Industrial Engineering*, vol. 143, p. 106392, May 2020.

[127] Y. X. Sun and Q. Y. Yan, "The application mode of energy big data and its enlightenment for power grid enterprises," in *Advanced Materials Research*, vol. 1008, pp. 1452–1455, Trans Tech Publ, 2014.

[128] D. Liu, G. Li, R. Fan, and G. Guo, "Research About Big Data Platform of Electrical Power System," in *Industrial IoT Technologies and Applications* (J. Wan, I. Humar, and D. Zhang, eds.), Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, (Cham), pp. 36–43, Springer International Publishing, 2016.

[129] K. M. Paramkusem and R. S. Aygun, "Classifying Categories of SCADA Attacks in a Big Data Framework," *Annals of Data Science*, vol. 5, pp. 359–386, Sept. 2018.

[130] P.-C. Chen, T. Dokic, and M. Kezunovic, *THE USE OF BIG DATA FOR OUTAGE MANAGEMENT IN DISTRIBUTION SYSTEMS*. June 2014.

[131] T. Reber and S. Booth, "Tariff Structures to Encourage Micro-Grid Deployment in Sub-Saharan Africa: Review and Recent Trends," *Current Sustainable/Renewable Energy Reports*, vol. 5, pp. 199–204, Sept. 2018.

[132] D. Barrios-O'Neill and G. Schuitema, "Online engagement for sustainable energy projects: A systematic review and framework for integration," *Renewable and Sustainable Energy Reviews*, vol. 54, pp. 1611–1621, Feb. 2016.

[133] R. Wang, S.-C. Hsu, S. Zheng, J.-H. Chen, and X. I. Li, "Renewable energy microgrids: Economic evaluation and decision making for government policies to contribute to affordable and clean energy," *Applied Energy*, vol. 274, p. 115287, Sept. 2020.

[134] S. Zhao, F. Blaabjerg, and H. Wang, "An Overview of Artificial Intelligence Applications for Power Electronics," *IEEE Transactions on Power Electronics*, vol. 36, pp. 4633–4658, Apr. 2021.

[135] N. Saputro, K. Akkaya, and S. Uludag, "A survey of routing protocols for smart grid communications," *Computer Networks*, vol. 56, pp. 2742–2771, July 2012.

[136] Y. Kabalci, "A survey on smart metering and smart grid communication," *Renewable and Sustainable Energy Reviews*, vol. 57, pp. 302–318, May 2016.

[137] H. M. Jawad, R. Nordin, S. K. Gharghan, A. M. Jawad, and M. Ismail, "Energy-Efficient Wireless Sensor Networks for Precision Agriculture: A Review," *Sensors*, vol. 17, p. 1781, Aug. 2017.

[138] P. Fraga-Lamas, T. M. Fernández-Caramés, and L. Castedo, "Towards the Internet of Smart Trains: A Review on Industrial IoT-Connected Railways," *Sensors*, vol. 17, p. 1457, June 2017.

[139] T. Ramathulasi and M. Rajasekhara Babu, "Comprehensive Survey of IoT Communication Technologies," in *Emerging Research in Data Engineering Systems and Computer Communications* (P. Venkata Krishna and M. S. Obaidat, eds.), Advances in Intelligent Systems and Computing, (Singapore), pp. 303–311, Springer, 2020.

[140] G. Walker and P. Devine-Wright, "Community renewable energy: What should it mean?," *Energy Policy*, vol. 36, pp. 497–500, Feb. 2008.

[141] D. Andreini and C. Bettinelli, "Business Model Definition and Boundaries," International Series in Advanced Management Studies, pp. 25–53, Cham: Springer International Publishing, 2017.

[142] A. Piterou and A.-M. Coles, "A review of business models for decentralised renewable energy projects," *Business Strategy and the Environment*, vol. 30, no. 3, pp. 1468–1480, 2021.

[143] P. Asmus and M. Lawrence, "Emerging microgrid business models," *Navigant research brief*, 2016.

[144] H. Dibaba, E. Vanadzina, G. Mendes, A. Pinomaa, and S. Honkapuro, "Business Model Design for Rural Off-the-Grid Electrification and Digitalization Concept," in *2020 17th International Conference on the European Energy Market (EEM)*, pp. 1–5, Sept. 2020. ISSN: 2165-4093.

[145] J. Söderholm, "Powered by actors and business models: Analysing the potential for energy community development in new regions using the case of kökar island," *IIIEE Master Thesis*, 2020.

[146] E. Vanadzina, G. Mendes, S. Honkapuro, A. Pinomaa, and H. Melkas, "Business models for community microgrids," in *2019 16th International Conference on the European Energy Market (EEM)*, pp. 1–7, Sept. 2019. ISSN: 2165-4093.

[147] C. Zhang, J. Wu, Y. Zhou, M. Cheng, and C. Long, "Peer-to-Peer energy trading in a Microgrid," *Applied Energy*, vol. 220, pp. 1–12, June 2018.

[148] T. Sousa, T. Soares, P. Pinson, F. Moret, T. Baroche, and E. Sorin, "Peer-to-peer and community-based markets: A comprehensive review," *Renewable and Sustainable Energy Reviews*, vol. 104, pp. 367–378, Apr. 2019.

[149] J. Diego Jiménez, S. M. Vives, E. G. Jiménez, and A. P. Mendoza, "Development of a methodology for planning and design of microgrids for rural electrification," in *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, pp. 1–6, Oct. 2017.

[150] P. Weston, W. Kalhoro, E. Lockhart, T. J. Reber, and S. S. Booth, "Financial and Operational Bundling Strategies for Sustainable Micro-Grid Business Models," tech. rep., Dec. 2018.

[151] G. Veilleux, T. Potisat, D. Pezim, C. Ribback, J. Ling, A. Krysztofinski, A. Ahmed, J. Papenheim, A. M. Pineda, S. Sembian, and S. Chucherd, "Techno-economic analysis of microgrid projects for rural electrification: A systematic approach to the redesign of Koh Jik off-grid case study," *Energy for Sustainable Development*, vol. 54, pp. 1–13, Feb. 2020.

[152] K. Murugaperumal and P. Ajay D Vimal Raj, "Feasibility design and techno-economic analysis of hybrid renewable energy system for rural electrification," *Solar Energy*, vol. 188, pp. 1068–1083, Aug. 2019.

[153] S. Emmanouil, J. Philhower, S. Macdonald, F. K. Khadim, M. Yang, E. Atsbeha, H. Nagireddy, N. Roach, E. Holzer, and E. N. Anagnostou, "A Comprehensive Approach to the Design of a Renewable Energy Microgrid for Rural Ethiopia: The Technical and Social Perspectives," *Sustainability*, vol. 13, p. 3974, Jan. 2021.

[154] S. Mandelli, J. Barbieri, R. Mereu, and E. Colombo, "Off-grid systems for rural electrification in developing countries: Definitions, classification and a comprehensive literature review," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 1621–1646, May 2016.

[155] B. Domenech, L. Ferrer-Martí, P. Lillo, R. Pastor, and J. Chiroque, "A community electrification project: Combination of microgrids and household systems fed by wind, pv or micro-hydro energies according to micro-scale resource evaluation and social constraints," *Energy for Sustainable Development*, vol. 23, pp. 275–285, 2014.

[156] J. Leary, M. Czyrnek-Delêtre, A. Alsop, A. Eales, L. Marandin, M. Org, M. Craig, W. Ortiz, C. Casillas, J. Persson, C. Dienst, E. Brown, A. While, J. Cloke, and K. Latoufis, "Finding the niche: A review of market assessment methodologies for rural electrification with small scale wind power," *Renewable and Sustainable Energy Reviews*, vol. 133, p. 110240, Nov. 2020.

[157] P. Loka, S. Moola, K. Polsani, S. Reddy, S. Fulton, and A. Skumanich, "A case study for micro-grid PV: lessons learned from a rural electrification project in India," *Progress in Photovoltaics: Research and Applications*, vol. 22, no. 7, pp. 733–743, 2014.

[158] C. Kirubi, A. Jacobson, D. M. Kammen, and A. Mills, "Community-Based Electric Micro-Grids Can Contribute to Rural Development: Evidence from Kenya," *World Development*, vol. 37, pp. 1208–1221, July 2009.

[159] F. Ahmad and M. S. Alam, "Economic and ecological aspects for microgrids deployment in India," *Sustainable Cities and Society*, vol. 37, pp. 407–419, Feb. 2018.

[160] M. Juanpera, P. Blechinger, L. Ferrer-Martí, M. M. Hoffmann, and R. Pastor, "Multicriteria-based methodology for the design of rural electrification systems. A case study in Nigeria," *Renewable and Sustainable Energy Reviews*, vol. 133, p. 110243, Nov. 2020.

[161] S. Bracco, F. Delfino, F. Pampararo, M. Robba, and M. Rossi, "A pilot facility for analysis and simulation of smart microgrids feeding smart buildings," *Renewable and Sustainable Energy Reviews*, vol. 58, pp. 1247–1255, May 2016.

[162] D. L. Rodrigues, X. Ye, X. Xia, and B. Zhu, "Battery energy storage sizing optimisation for different ownership structures in a peer-to-peer energy sharing community," *Applied Energy*, vol. 262, p. 114498, Mar. 2020.

[163] A. A. Eras-Almeida, M. Fernández, J. Eisman, J. G. Martín, E. Caamaño, and M. A. Egido-Aguilera, "Lessons Learned from Rural Electrification Experiences with Third Generation Solar Home Systems in Latin America: Case Studies in Peru, Mexico, and Bolivia," *Sustainability*, vol. 11, p. 7139, Jan. 2019.

[164] N. J. Williams, P. Jaramillo, J. Taneja, and T. S. Ustun, "Enabling private sector investment in microgrid-based rural electrification in developing countries: A review," *Renewable and Sustainable Energy Reviews*, vol. 52, pp. 1268–1281, Dec. 2015.

[165] B. Cornélusse, I. Savelli, S. Paoletti, A. Giannitrapani, and A. Vicino, "A community microgrid architecture with an internal local market," *Applied Energy*, vol. 242, pp. 547–560, May 2019.

[166] T. Levin and V. M. Thomas, "Utility-maximizing financial contracts for distributed rural electrification," *Energy*, vol. 69, pp. 613–621, May 2014.

[167] S. R. Khandker, D. F. Barnes, and H. A. Samad, "Welfare Impacts of Rural Electrification: A Panel Data Analysis from Vietnam," *Economic Development and Cultural Change*, vol. 61, pp. 659–692, Apr. 2013.

[168] P. Lillo, L. Ferrer-Martí, A. Boni, and [U+FFFD] Fernández-Baldor, "Assessing management models for off-grid renewable energy electrification projects using the Human Development approach: Case study in Peru," *Energy for Sustainable Development*, vol. 25, pp. 17–26, Apr. 2015.

[169] P. Yadav, P. J. Davies, and D. Palit, "Distributed solar photovoltaics landscape in Uttar Pradesh, India: Lessons for transition to decentralised rural electrification," *Energy Strategy Reviews*, vol. 26, p. 100392, Nov. 2019.

[170] C. Gradl and C. Knobloch, *Energize the BoP!: Energy Business Model Generator for Low-income Markets; a Practitioners' Guide*. Endeva UG, 2011.

[171] M. Daneshvar, M. Pesaran, and B. Mohammadi-ivatloo, "Transactive energy integration in future smart rural network electrification," *Journal of Cleaner Production*, vol. 190, pp. 645–654, July 2018.

[172] A. Paudel and G. H. Beng, "A Hierarchical Peer-to-Peer Energy Trading in Community Microgrid Distribution Systems," in *2018 IEEE Power Energy Society General Meeting (PESGM)*, pp. 1–5, Aug. 2018. ISSN: 1944-9933.

[173] F. C. Robert, G. S. Sisodia, and S. Gopalan, "The critical role of anchor customers in rural microgrids: Impact of load factor on energy cost," in *2017 International Conference on Computation of Power, Energy Information and Commuincation (ICCPEIC)*, pp. 398–403, Mar. 2017.

[174] S. Booth, X. Li, I. Baring-Gould, D. Kollanyi, A. Bharadwaj, and P. Weston, "Productive Use of Energy in African Micro-Grids: Technical and Business Considerations," tech. rep., Aug. 2018.

[175] A. Moreno and A. Bareisaite, "Scaling up access to electricity: pay-as-you-go plans in off-grid energy services," 2015.

[176] H. K. Trabish, "Public purpose microgrids: Mixed-ownership models spur utility investment in growing sector," *available athttp://www. utilitydive. com/news/public-purpose-microgrids-mixedownership-models-spur-utility-investment-i/425296*, 2016.

[177] D. Hawkey, J. Webb, and M. Winskel, "Organisation and governance of urban energy systems: district heating and cooling in the UK," *Journal of Cleaner Production*, vol. 50, pp. 22–31, July 2013.

[178] S. Bird, C. Hotaling, A. Enayati, and T. Ortmeyer, "Resilient community microgrids: governance and operational challenges," ch. 4, pp. 65–95, Woodhead Publishing, Jan. 2019.

[179] D. A. S. says, "New Business Models for Microgrids: Energy-as-a-Services (EaaS) Leads in Popularity," Aug. 2018.

[180] A. Ingalalli and S. Kamalasadan, "Participation of Networked Microgrids in Energy-as-a-Service Model for Enhancing Grid Resiliency," in *2021 IEEE Power Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5, Feb. 2021. ISSN: 2472-8152.

[181] U. Paukstadt and J. Becker, "From Energy as a Commodity to Energy as a Service—A Morphological Analysis of Smart Energy Services," *Schmalenbach Journal of Business Research*, vol. 73, pp. 207–242, June 2021.

[182] B. P. Koirala, E. Koliou, J. Friege, R. A. Hakvoort, and P. M. Herder, "Energetic communities for community energy: A review of key issues and trends shaping integrated community energy systems," *Renewable and Sustainable Energy Reviews*, vol. 56, pp. 722–744, Apr. 2016.

[183] L. Yavuz, A. Önen, S. M. Muyeen, and I. Kamwa, "Transformation of microgrid to virtual power plant – a comprehensive review," *IET Generation, Transmission & Distribution*, vol. 13, pp. 1994–2005, Jan. 2019.

[184] A. D. Rathnayaka, V. M. Potdar, T. S. Dillon, and S. Kuruppu, "Formation of virtual community groups to manage prosumers in smart grids," *International Journal of Grid and Utility Computing*, vol. 6, pp. 47–56, Dec. 2014.

[185] S. Cui, Y.-W. Wang, J.-W. Xiao, and N. Liu, "A Two-Stage Robust Energy Sharing Management for Prosumer Microgrid," *IEEE Transactions on Industrial Informatics*, vol. 15, pp. 2741–2752, May 2019.

[186] J. Bowes, C. Booth, and S. Strachan, "System interconnection as a path to bottom up electrification," in *2017 52nd International Universities Power Engineering Conference (UPEC)*, pp. 1–5, Aug. 2017.

[187] A. Diaz-Valdivia and M. Poblet, "Connecting the Grids: A Review of Blockchain Governance in Distributed Energy Transitions," SSRN Scholarly Paper ID 3777293, Rochester, NY, Dec. 2020.

[188] V. Kulkarni and K. Kulkarni, "A Blockchain-based Smart Grid Model for Rural Electrification in India," in *2020 8th International Conference on Smart Grid (icSmartGrid)*, pp. 133–139, June 2020.

[189] A. Ahl, M. Yarime, K. Tanaka, and D. Sagawa, "Review of blockchain-based distributed energy: Implications for institutional development," *Renewable and Sustainable Energy Reviews*, vol. 107, pp. 200–211, June 2019.

[190] T. Chen, H. Pourbabak, and W. Su, "Electricity market reform," ch. 5, pp. 97–121, Woodhead Publishing, Jan. 2019.

[191] E. M. Gui, I. MacGill, and R. Betz, "Community Microgrid Investment Planning: A Conceptual Framework," in *2018 IEEE International Smart Cities Conference (ISC2)*, pp. 1–5, Sept. 2018.

[192] V. Vahidinasab, M. Tabarzadi, H. Arasteh, M. I. Alizadeh, M. Mohammad Beigi, H. R. Sheikhzadeh, K. Mehran, and M. S. Sepasian, "Overview of Electric Energy Distribution Networks Expansion Planning," *IEEE Access*, vol. 8, pp. 34750–34769, 2020.

[193] S. M. Wyse and C. E. Hoicka, ""By and for local people": assessing the connection between local energy plans and community energy," *Local Environment*, vol. 24, pp. 883–900, Sept. 2019.

[194] Y. Parag and M. Ainspan, "Sustainable microgrids: Economic, environmental and social costs and benefits of microgrid deployment," *Energy for Sustainable Development*, vol. 52, pp. 72–81, Oct. 2019.

[195] J. Kumar, B. V. Suryakiran, A. Verma, and T. S. Bhatti, "Analysis of techno-economic viability with demand response strategy of a grid-connected microgrid model for enhanced rural electrification in Uttar Pradesh state, India," *Energy*, vol. 178, pp. 176–185, July 2019.

[196] Z. Huang, H. Yu, Z. Peng, and M. Zhao, "Methods and tools for community energy planning: A review," *Renewable and Sustainable Energy Reviews*, vol. 42, pp. 1335–1348, Feb. 2015.

[197] R. K. Oueid, "Microgrid finance, revenue, and regulation considerations," *The Electricity Journal*, vol. 32, pp. 2–9, June 2019.

[198] A. Sharma, "'We Do Not Want Fake Energy': The Social Shaping of a Solar Micro-grid in Rural India," *Science, Technology and Society*, vol. 25, pp. 308–324, July 2020.

[199] V. Azarova, J. Cohen, C. Friedl, and J. Reichl, "Designing local renewable energy communities to increase social acceptance: Evidence from a choice experiment in Austria, Germany, Italy, and Switzerland," *Energy Policy*, vol. 132, pp. 1176–1183, Sept. 2019.

[200] M. Busse and R. Siebert, "Acceptance studies in the field of land use—A critical and systematic review to advance the conceptualization of acceptance and acceptability," *Land Use Policy*, vol. 76, pp. 235–245, July 2018.

[201] J. Gaede and I. H. Rowlands, "Visualizing social acceptance research: A bibliometric review of the social acceptance literature for energy technology and fuels," *Energy Research & Social Science*, vol. 40, pp. 142–158, June 2018.

[202] M. Wolsink, "The research agenda on social acceptance of distributed generation in smart grids: Renewable as common pool resources," *Renewable and Sustainable Energy Reviews*, vol. 16, pp. 822–835, Jan. 2012.

[203] M. Wolsink, "Social acceptance, lost objects, and obsession with the 'public'—The pressing need for enhanced conceptual and methodological rigor," *Energy Research & Social Science*, vol. 48, pp. 269–276, Feb. 2019.

[204] P. M. Blau, "A macrosociological theory of social structure," *American journal of sociology*, vol. 83, no. 1, pp. 26–54, 1977.

[205] E. Creamer, W. Eadson, B. v. Veelen, A. Pinker, M. Tingey, T. Braunholtz-Speight, M. Markantoni, M. Foden, and M. Lacey-Barnacle, "Community energy: Entanglements of community, state, and private sector," *Geography Compass*, vol. 12, no. 7, p. e12378, 2018.

[206] M. Wolsink, "Distributed energy systems as common goods: Socio-political acceptance of renewables in intelligent microgrids," *Renewable and Sustainable Energy Reviews*, vol. 127, p. 109841, July 2020.

[207] R. Wüstenhagen, M. Wolsink, and M. J. Bürer, "Social acceptance of renewable energy innovation: An introduction to the concept," *Energy Policy*, vol. 35, pp. 2683–2691, May 2007.

[208] V. Z. Gjorgievski, S. Cundeva, and G. E. Georghiou, "Social arrangements, technical designs and impacts of energy communities: A review," *Renewable Energy*, 2021.

[209] O. Muza and R. Debnath, "Disruptive innovation for inclusive renewable policy in sub-Saharan Africa: A social shaping of technology analysis of appliance uptake in Rwanda," *Renewable Energy*, vol. 168, pp. 896–912, May 2021.

[210] T. von Wirth, L. Gislason, and R. Seidl, "Distributed energy systems on a neighborhood scale: Reviewing drivers of and barriers to social acceptance," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 2618–2628, Feb. 2018.

[211] J. Moody and D. R. White, "Structural Cohesion and Embeddedness: A Hierarchical Concept of Social Groups," *American Sociological Review*, vol. 68, no. 1, pp. 103–127, 2003.

[212] M. Wolsink, "Social acceptance revisited: gaps, questionable trends, and an auspicious perspective," *Energy Research & Social Science*, vol. 46, pp. 287–295, Dec. 2018.

[213] P. C. Stern, "Individual and household interactions with energy systems: Toward integrated understanding," *Energy Research & Social Science*, vol. 1, pp. 41–48, Mar. 2014.

[214] J. J. Cohen, J. Reichl, and M. Schmidthaler, "Re-focussing research efforts on the public acceptance of energy infrastructure: A critical review," *Energy*, vol. 76, pp. 4–9, Nov. 2014.

[215] C. Friedl and J. Reichl, "Realizing energy infrastructure projects – A qualitative empirical analysis of local practices to address social acceptance," *Energy Policy*, vol. 89, pp. 184–193, Feb. 2016.

[216] P. Ponce, K. Polasko, and A. Molina, "End user perceptions toward smart grid technology: Acceptance, adoption, risks, and trust," *Renewable and Sustainable Energy Reviews*, vol. 60, pp. 587–598, July 2016.

[217] S. Karytsas, I. Vardopoulos, and E. Theodoropoulou, "Factors Affecting Sustainable Market Acceptance of Residential Microgeneration Technologies. A Two Time Period Comparative Analysis," *Energies*, vol. 12, p. 3298, Jan. 2019.

[218] E. Mengelkamp, P. Staudt, J. Gärttner, C. Weinhardt, and J. Huber, "Quantifying Factors for Participation in Local Electricity Markets," in *2018 15th International Conference on the European Energy Market (EEM)*, pp. 1–5, June 2018. ISSN: 2165-4093.

[219] N. Li, R. Hakvoort, and Z. Lukszo, "Cost Allocation in Integrated Community Energy Systems—Social Acceptance," *Sustainability*, vol. 13, p. 9951, Jan. 2021.

[220] D. Peters, J. Axsen, and A. Mallett, "The role of environmental framing in socio-political acceptance of smart grid: The case of British Columbia, Canada," *Renewable and Sustainable Energy Reviews*, vol. 82, pp. 1939–1951, Feb. 2018.

[221] C. Dermont, K. Ingold, L. Kammermann, and I. Stadelmann-Steffen, "Bringing the policy making perspective in: A political science approach to social acceptance," *Energy Policy*, vol. 108, pp. 359–368, Sept. 2017.

[222] B. J. Rygg, M. Ryghaug, and G. Yttri, "Is local always best? Social acceptance of small hydropower projects in Norway.," *International Journal of Sustainable Energy Planning and Management*, vol. 31, pp. 161–174, May 2021.

[223] G. Assefa and B. Frostell, "Social sustainability and social acceptance in technology assessment: A case study of energy technologies," *Technology in Society*, vol. 29, pp. 63–78, Jan. 2007.

[224] H. Suk and J. Hall, "Integrating Quality of Life in Sociotechnical Design: A Review of Microgrid Design Tools and Social Indicators," American Society of Mechanical Engineers Digital Collection, Nov. 2019.

[225] C. Alvial-Palavicino, N. Garrido-Echeverría, G. Jiménez-Estévez, L. Reyes, and R. Palma-Behnke, "A methodology for community engagement in the introduction of renewable based smart microgrid," *Energy for Sustainable Development*, vol. 15, pp. 314–323, Sept. 2011.

[226] C. Rosenthal-Sabroux and M. Grundstein, "A global vision of information management.," in *MoDISE-EUS*, pp. 55–66, 2008.

[227] Y. Zekaria and R. Chitchyan, "Qualitative study of skills needs for community energy projects," in *Presentación en la Conferencia sobre Comunidades Energéticas organizada por Eco-SESA de la Universidad de Grenoble Alpes. Energy Communities for Collective self-consumption: frameworks, practices and tools*, vol. 16, 2020.

[228] F. Girbau-Llistuella, A. Sumper, R. Gallart-Fernandez, and S. Martinez-Farrero, "Smart rural grid pilot in Spain," ch. 14, pp. 315–345, Woodhead Publishing, Jan. 2019.

[229] M. Torero, "The Impact of Rural Electrification: Challenges and Ways Forward," *Revue d'economie du developpement*, vol. 23, no. HS, pp. 49–75, 2015.

[230] S. Hirmer and P. Guthrie, "The benefits of energy appliances in the off-grid energy sector based on seven off-grid initiatives in rural Uganda," *Renewable and Sustainable Energy Reviews*, vol. 79, pp. 924–934, Nov. 2017.

[231] E. K. Stigka, J. A. Paravantis, and G. K. Mihalakakou, "Social acceptance of renewable energy sources: A review of contingent valuation applications," *Renewable and Sustainable Energy Reviews*, vol. 32, pp. 100–106, Apr. 2014.

[232] A. Ali, W. Li, R. Hussain, X. He, B. W. Williams, and A. H. Memon, "Overview of Current Microgrid Policies, Incentives and Barriers in the European Union, United States and China," *Sustainability*, vol. 9, p. 1146, July 2017.

[233] M. Derks and H. Romijn, "Sustainable performance challenges of rural microgrids: Analysis of incentives and policy framework in Indonesia," *Energy for Sustainable Development*, vol. 53, pp. 57–70, Dec. 2019.

[234] O. Ellabban and H. Abu-Rub, "Smart grid customers' acceptance and engagement: An overview," *Renewable and Sustainable Energy Reviews*, vol. 65, pp. 1285–1298, Nov. 2016.

[235] A. Anderson and S. Suryanarayanan, "An Enterprise Systems Engineering Approach to Electrification: Looking at the Bigger Picture Through Life-Cycle Analysis of Community Microgrids: A Case Study in Papua New Guinea," *IEEE Electrification Magazine*, vol. 6, pp. 18–31, Dec. 2018.

[236] F. P. Boon and C. Dieperink, "Local civil society based renewable energy organisations in the Netherlands: Exploring the factors that stimulate their emergence and development," *Energy Policy*, vol. 69, pp. 297–307, June 2014.

[237] D. Geelen, A. Reinders, and D. Keyson, "Empowering the end-user in smart grids: Recommendations for the design of products and services," *Energy Policy*, vol. 61, pp. 151–161, Oct. 2013.

[238] R. Sauter and J. Watson, "Strategies for the deployment of micro-generation: Implications for social acceptance," *Energy Policy*, vol. 35, pp. 2770–2779, May 2007.

[239] W. Jager, "Stimulating the diffusion of photovoltaic systems: A behavioural perspective," *Energy Policy*, vol. 34, pp. 1935–1943, Sept. 2006.

[240] A. M. Adil and Y. Ko, "Socio-technical evolution of Decentralized Energy Systems: A critical review and implications for urban planning and policy," *Renewable and Sustainable Energy Reviews*, vol. 57, pp. 1025–1037, May 2016.

[241] M. O. Müller, A. Stämpfli, U. Dold, and T. Hammer, "Energy autarky: A conceptual framework for sustainable regional development," *Energy Policy*, vol. 39, pp. 5800–5810, Oct. 2011.

[242] K. Chmutina, B. Wiersma, C. I. Goodier, and P. Devine-Wright, "Concern or compliance? Drivers of urban decentralised energy initiatives," *Sustainable Cities and Society*, vol. 10, pp. 122–129, Feb. 2014.

[243] R. P. J. M. Raven, R. M. Mourik, C. F. J. Feenstra, and E. Heiskanen, "Modulating societal acceptance in new energy projects: Towards a toolkit methodology for project managers," *Energy*, vol. 34, pp. 564–574, May 2009.

[244] A. Niez, "Comparative study on rural electrification policies in emerging economies: Keys to successful policies," 2010.

[245] S. K. Sahoo, A. K. Sinha, and N. K. Kishore, "Control Techniques in AC, DC, and Hybrid AC–DC Microgrid: A Review," *IEEE Journal of Emerging and Selected Topics in Power Electronics*, vol. 6, pp. 738–759, June 2018.

[246] E. Espina, J. Llanos, C. Burgos-Mellado, R. Cárdenas-Dobson, M. Martínez-Gómez, and D. Sáez, "Distributed Control Strategies for Microgrids: An Overview," *IEEE Access*, vol. 8, pp. 193412–193448, 2020.

[247] H. Louie, P. Dauenhauer, M. Wilson, A. Zomers, and J. Mutale, "Eternal Light: Ingredients for Sustainable Off-Grid Energy Development," *IEEE Power and Energy Magazine*, vol. 12, pp. 70–78, July 2014.

[248] M. S. S. Danish, H. Matayoshi, H. R. Howlader, S. Chakraborty, P. Mandal, and T. Senjyu, "Microgrid Planning and Design: Resilience to Sustainability," in *2019 IEEE PES GTD Grand International Conference and Exposition Asia (GTD Asia)*, pp. 253–258, Mar. 2019.

[249] M. F. Zia, M. Benbouzid, E. Elbouchikhi, S. M. Muyeen, K. Techato, and J. M. Guerrero, "Microgrid Transactive Energy: Review, Architectures, Distributed Ledger Technologies, and Market Analysis," *IEEE Access*, vol. 8, pp. 19410–19432, 2020.

[250] A. Cagnano, E. De Tuglie, and P. Mancarella, "Microgrids: Overview and guidelines for practical implementations and operation," *Applied Energy*, vol. 258, p. 114039, Jan. 2020.

[251] A. Boche, C. Foucher, and L. F. L. Villa, "A systemic and multi-disciplinary diagnosis model for microgrids sustainability studies," in *International Conference on Innovations in Energy Engineering & Cleaner Production (IEE CP 2022)*, 2022.

[252] F. Norouzi, T. Hoppe, L. R. Elizondo, and P. Bauer, "A review of socio-technical barriers to Smart Microgrid development," *Renewable and Sustainable Energy Reviews*, vol. 167, p. 112674, Oct. 2022.

[253] T. Brown, J. Hörsch, and D. Schlachtberger, "Pypsa: Python for power system analysis," *arXiv preprint arXiv:1707.09913*, 2017.

[254] Y. Van Tendeloo and H. Vangheluwe, "An overview of pythonpdevs," *JDF*, vol. 2016, pp. 59–66, 2016.

[255] Y. Van Tendeloo and H. Vangheluwe, "Activity in pythonpdevs," in *ITM Web of Conferences.-Place of publication unknown*, vol. 3, p. 01002, 2014.

[256] Y. Van Tendeloo and H. Vangheluwe, "Pythonpdevs: a distributed parallel devs simulator," in *Proceedings of the Symposium on Theory of Modeling & Simulation: DEVS Integrative M&S Symposium*, pp. 91–98, 2015.

Microgrid State of Health Systemic Survey

Energy Flows

How well is the energy production and consumption balanced in terms of sizing and capacity?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How reliable and resilient is the microgrid in terms of components quality and grid stability?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How stable and reliable is the energy quality and grid performance in terms of protection against power outages and failures?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Financial Flows

How well is the microgrid's business model designed and optimized for sustainable revenue generation and cost management?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively is the microgrid's financial performance being managed and monitored to ensure long-term financial sustainability?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How well is the microgrid's long-term planning and investment strategy aligned with its financial goals and objectives?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How is the situation in term of exterior partnership with the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Information Flows

How efficient and reliable is the control and management of the microgrid's information and data systems?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively is the microgrid's data being collected, processed and utilized to improve performance and decision-making?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively is the microgrid's data being collected, processed and utilized to improve performance and decision-making?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Social Flows

How well is the community's knowledge and understanding of the microgrid technology and its benefits?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively are the rules and regulations governing the microgrid designed and implemented to meet the needs and expectations of the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively are the rules and regulations governing the microgrid designed and implemented to meet the needs and expectations of the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

What the energy tier evolution of the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Energy → Finance

How efficient is the energy generation capacity in relation to the microgrid's capital expenditure?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How well is the energy production aligned with the microgrid's revenue generation strategy?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively does the microgrid's energy generation contribute to the overall financial sustainability of the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Energy → Information

How accurate and reliable are the data measurement and monitoring systems within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

What is the extent of data coverage collected within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively the measures in place ensure the security and privacy of collected data within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Energy → Social

How well does the microgrid's energy generation align with the community's energy needs and preferences?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How well does the microgrid's energy tier system accurately reflect and address the diverse energy needs and consumption patterns of the community it serves?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively does the microgrid's energy generation contribute to the overall social development of the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Finance → Energy

How does the microgrid's financial performance impact its ability to invest in and maintain energy infrastructure?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what extent is the microgrid's financial management system able to adapt to changes in energy demand and supply?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How is the speed of maintenance response?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Finance → Information

How does the microgrid's financial performance impact its ability to invest in and maintain information and communication technology infrastructure?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what extent does the microgrid’s financial structure effectively balance the implementation and utilization of advanced information technologies for improved microgrid management and optimization?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How fast is the response for addressing and resolving maintenance issues within the microgrid’s information infrastructure?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Finance → Social

How does the microgrid's financial structure support the frequency of engagement and communication with the local community in regards to microgrid usage and maintenance?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How robust is the microgrid's payment collection infrastructure in terms of accommodating and collecting payments from the local community in a timely and efficient manner?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How does the microgrid's financial plan ensure the participation and empowerment of the local community in decision-making processes related to the management and operation of the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Information → Energy

How accurate and stable is the power control of the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How well does the information field support the prediction and management of power fluctuations within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively are any issues or failures within the microgrid control system detected and addressed in a timely manner?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Information → Finance

To what degree does the availability and quality of data facilitate accurate and efficient financial planning and operations within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How efficiently is payment collection facilitated by the data and control objectives established for the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what degree does the microgrid's data management system aid in the detection and prevention of financial losses or fraud?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Information → Social

How transparent is the microgrid in terms of communicating data usage policies and practices to the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How well is the community engagement and communication supported by the information system?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How effectively does the information system support the dissemination of knowledge about the microgrid to the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Social → Energy

To what extent does the community usage of the microgrid energy align with the microgrid's sustainability objectives?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How prevalent is illegal overconsumption within the community utilizing the microgrid energy?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

How active is the community in terms of participating in the maintenance and upkeep of the microgrid infrastructure?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Social → Finance

How effectively are community members able to access and accept billing practice associated with their microgrid usage?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what degree are illegal energy connections present within the community?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what extent do community members participate in the decision-making process related to the allocation of funds within the microgrid?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

Social → Information

To what extent does the community provide accurate and timely data on energy consumption patterns within the microgrid system?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what degree does the community actively participate in the maintenance and upkeep of the microgrid's information systems?

☐ Excellent ☐ Good ☐ Poor ☐ Bad

To what extent does the community actively engage in the identification and reporting of potential issues or concerns related to the microgrid system?

☐ Excellent ☐ Good ☐ Poor ☐ Bad