Master Thesis

Double MSc Degree in Industrial Engineering and Energy Engineering

Dynamic Simulation and Stability Analysis of Power Systems: Development of RMS and EMT Tools within VeraGrid at eRoots

January 2026

Author: Maria Sans Esqué

Supervisors: Vinícius Albernaz Lacerda

Josep Fanals i Batllori

Escola Tècnica Superior
d'Enginyeria Industrial de Barcelona

Abstract

Resumen

Resum

Contents

List of Figures

List of Tables

Glossary

Symbols

$\boldsymbol{\theta}$	Voltage angle vector
λ	Eigenvalue
ζ	Damping ratio
$\boldsymbol{\omega}$	Angular frequency vector
$\boldsymbol{A}$	State matrix
$\boldsymbol{B}$	Input matrix
$\boldsymbol{C}$	Output matrix
$\boldsymbol{D}$	Feedthrough matrix
$\boldsymbol{x}$	State vector
$\boldsymbol{u}$	Input vector
$\boldsymbol{y}$	Output vector
τ	Time constant

Acronyms

AC	Alternating Current
CIG	Converter-interfaced Generation
DAE	Differential Algebraic Equation
DC	Direct Current
EEA	European Environment Agency
EMT	Electro Magnetic Transient
ETSEIB	Escola Tècnica Superior d'Enginyeria Industrial de Barcelona
GUI	Guided User Interface
HVAC	Heating Ventilation and Air Conditioning
IBR	Inverter-Based Resources
ICAEN	Institut Català de l'Energia
IGE	Induction Generator Effect
LCA	Life Cycle Assessment
ODE	Ordinary Differential Equation
PF	Participation Factor
PSCAD	Power Systems Computer Aided Design
RMS	Root Mean Square
SFA	Shifted Frequency Analysis
SPT	Singular Perturbation Theory
SSCI	Subsynchronous Control Interaction
SSR	Subsynchronous Resonance
TDS	Time Domain Simulation

Preface

1 Introduction

1.1 Motivation

...

1.2 Objectives

This thesis is part of the ongoing development of the dynamic simulation framework, with a focus on extending its capabilities to include small-signal stability analysis and foundational electromagnetic transient (EMT) modeling. The work is carried out within the Veragrid environment, where new modules and methodologies are being implemented. The main goal is to equip Veragrid with advanced tools for studying the dynamic behavior of power systems, integrating symbolic modeling, numerical routines, and graphical interfaces for analysis and visualization.

General Objectives

To develop and validate advanced methodologies for dynamic simulation and stability analysis of power systems by incorporating small-signal stability techniques and foundational EMT modeling, fully integrated into the Veragrid environment.

Specific Objectives

- To develop the small-signal analysis module for RMS models, including the formulation and linearization of differential-algebraic equations at the operating point, the computation of eigenvalues and participation factors from the jacobian matrix, and its integration into Veragrid's graphical interface.
- To implement the foundational components for EMT simulation, modeling transmission lines and system elements in the abc domain, applying discretization techniques such as the Dommel algorithm and alternatives like the 2S-DIRK method, and validating the EMT solver using benchmark systems compared against commercial tools such as PSCAD.
- To extend symbolic system formulation to support custom models and control schemes, improving numerical routines, and ensuring consistent initialization of dynamic studies.
- To validate the developed methodologies through case studies, continuously comparing results with commercial tools to ensure model reliability and correctness.

1.3 Scope

This thesis is part of the ongoing development of Veragrid, a leading software platform for power system planning and simulation. The work focuses on improving dynamic simulation tools for modern grids, particularly in the context of small-signal stability and electromagnetic transient (EMT) modeling. Over a nine-month period—from September 2025 to May 2026—the project aims to build essential components that support symbolic formulation, numerical validation, and integration with existing simulation environments.

The first major area of focus is the implementation of small-signal stability analysis using RMS-based state-space models. This includes the computation of eigenvalues and participation factors, symbolic reduction of system equations, and integration of these routines into the VeraGrid graphical interface. The goal is to provide researchers and engineers with intuitive and accurate tools for identifying dominant modes and assessing system stability under varying conditions.

The second area involves the development of a foundational EMT solver in the abc domain. This includes modeling transmission lines and components, implementing discretization techniques such as the Dommel algorithm and two-stage diagonally implicit Runge-Kutta (2S-DIRK) methods, and benchmarking solver performance against commercial tools like PSCAD. Although the EMT module is not intended to be exhaustive, it serves as a proof of concept for future expansion and integration.

All development is conducted in Python, with an emphasis on code quality, symbolic computation, and reproducibility. The thesis also includes continuous benchmarking and validation using real-world data, including industrial cases. Technical supervision is provided by the eRoots team, ensuring alignment with architectural standards and long-term project goals.

1.4 Structure of the document

This thesis considers all the steps involved in the development process of the RMS small-signal stability analysis and the whole EMT dynamic simulation, from the mathematical modelling to the software implementation and validation of results.

- Chapter 3 describes from the theoretical framework to the implementation and benchmarking of the small-signal stability analysis in VeraGrid.
- Chapter 4 describes ...

1.5 State of the art

things to say:

The most developed software solutions for EMT simulations focusonwaveformdynamicswithandwithout-

converterswitch ing and include models for very detailed component-level anal ysis. Examples of software environments are PSCAD, EMTP, Simulink, PLECS, among others.

1.6 Veragrid

VeraGrid is a comprehensive software platform for power system planning and simulation, developed to offer both technical accuracy and accessibility. It integrates a wide range of analytical and optimization tools, covering everything from traditional steady-state analyses to advanced planning functions that address the challenges of modern electrical grids. Its capabilities include conventional studies such as power flow, short-circuit, and contingency analyses, as well as linear and non-linear optimization modules used for operational decision-making and long-term investment assessment. Many of these functions are based on established industry standards, while others are the result of ongoing research and innovation, designed to push the boundaries of what is possible in open and high-performance grid modelling.

Figure 1.1: VeraGrid banner. *Source*: VeraGrid documentation [1].

The development of VeraGrid began in 2015 with a clear objective: to create a robust programming library supported by a user-friendly interface. This pragmatic vision led to a unique ecosystem where reliability and simplicity coexist with scientific rigour. Over the years, the platform has evolved through a combination of commercial projects, academic collaborations, and internal research initiatives, ensuring that its algorithms and methods remain both practical and forward-looking. Some of its innovations emerged from the need to address real-world industrial requirements, while others stemmed from curiosity and the exploration of new computational paradigms.

VeraGrid serves a wide audience. For professionals, it provides transparent, efficient, and reproducible tools that enable detailed grid analysis and operational planning. For researchers, it represents an open and validated environment capable of integrating experimental algorithms and comparing methodologies. For educators and students, it offers a pedagogical platform that connects theoretical concepts with practical, industry-grade implementations. This versatility allows VeraGrid to act as a bridge between academia, industry, and future generations of engineers.

Beyond conventional functionalities, VeraGrid includes an extensive set of features designed for

modern power systems. These include a multi-layered architecture for both usability and computational efficiency; an AC/DC generalized power flow engine that supports hybrid grids and converter-based systems; short-circuit and fault analysis modules that incorporate converter control logic; and a suite of optimal power flow, expansion planning, and investment analysis tools. The platform also integrates time-series simulation capabilities for renewable energy forecasting, storage operation, and market coupling, enabling comprehensive scenario-based studies.

Thanks to its open-core design, VeraGrid can be easily extended and interfaced with external tools, ensuring interoperability and adaptability to specific project needs. It stands not only as a software product but as a complete analytical framework that evolves alongside the energy transition, enabling engineers, researchers, and institutions to model, plan, and optimize electrical networks with transparency and scientific depth.

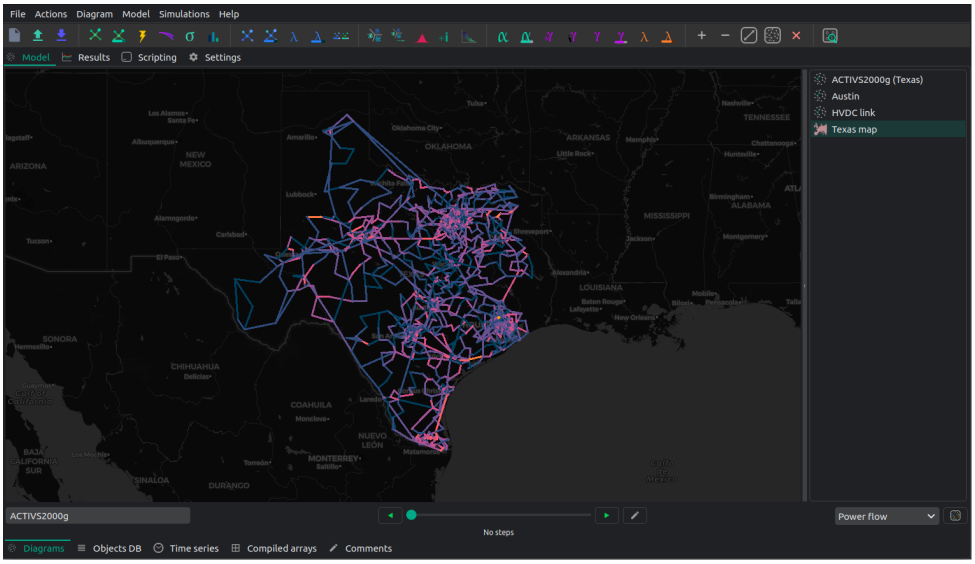

Figure 1.2: VeraGrid main page. *Source*: VeraGrid documentation [1].

2 Time-Domain Simulations

2.1 Dynamic framework

The evolution of power systems towards greater complexity and decentralization has introduced new challenges in modeling and analyzing system dynamics. Traditional approaches, which often relied on simplifying assumptions and linear models, are increasingly inadequate for capturing the intricate behaviors exhibited by modern grids. These grids, characterized by high penetration of Inverter-Based Resources (IBRs), such as photovoltaic and wind generation, needing more detailed and accurate simulation methods to ensure reliable operation and stability [2].

Time-domain simulations (TDS) have become a key point in the study of power system dynamics, providing a temporal perspective on system behavior following disturbances. Unlike steady-state analyses, TDS allows for the examination of transient phenomena, including voltage sags, frequency deviations, and oscillatory modes, which are critical for assessing system stability and performance. The ability to simulate the system's response over time enables engineers to design and test control strategies, protection schemes, and operational protocols in a virtual environment before implementation in the field.

A TDS requires the specification of two parts: the model of the system which must include the differential equations that describe the system and the integration algorithm to be used to solve those equations over time [2].

The system model is a set of dynamic equations that represent and interconnect the components of a system expressed with differential equations. Those equations are divided between the device equations that store the state of the system and the circuit dynamics of the network. Depending on the type of simulation to be performed, the model of the system can have many different forms, mainly influenced by simplifications and transformations of the system equations and variables. A general explanation of the main methods to simplify and transform systems based on [2] are listed below.

- *Averaging dynamics*: These simplifying methods replace fast dynamics of complicated time-varying variables with their average value. An example is setting the system's frequency to its averaged value or replacing discontinuous switching in a converter with its average behavior. Three main methods are distinguished:
  - *Steady-state phasors*: Commonly used in engineering using the wave's root-mean-square value instead of the instantaneous value. It requires stationary conditions and homogeneous frequency.

  - *Dynamic phasors*: Used to describe the behavior of time-varying periodic signals, particularly under conditions where the system frequency or amplitude may vary slowly compared to the fundamental period. These capture both the slowly-varying average and instantaneous phase of the signal. Three different approaches are considered (Fourier approach, time-varying phasors, and Shifted Frequency Analysis (SFA) based), but all of them must ensure that the integral of the signal over the studied period is well-defined and that the signal is band-limited for a given frequency.

  - *Three-Phase signals & models*: For three-phase systems, SFA is applied phase by phase to a three-phase *abc* representation to study unbalanced systems. Using the Fortescue transform [3], multi-phase quantities (voltages, currents, electromagnetic fluxes, and impedances) with a common system frequency w_s and offset θ are represented in a two-dimensional vector dividing the positive and negative-frequency. It should be noted that in balanced systems, the signal phasor is completely determined by the positive-sequence phasor.

- *Reference frame transformations*: Such as the Park Transform (or *dq0* transform), which allow the representation of time-variant signals as quasi-static ones, simplifying its analysis.

- *Singular Perturbation Theory* (*SPT*): Used to analyze dynamic systems with widely separated time scales by decomposing them into fast and slow subsystems. In power systems, SPT is applied to separate the fast transients of synchronous machines and converters (milliseconds) from the slow dynamics of controllers and system frequency (seconds to minutes), enabling more efficient RMS and small-signal analyses.

2.1.1 Types of TDS

In power systems, time-domain simulations can generally be divided into two main categories [2].

The first category, *Quasi-Static Phasor* (*QSP*) *simulations*, models the transmission network dynamics in an algebraic manner by considering discrete transitions between successive steady-state operating points, in other words, QSP focus on dynamics that do not deviate much from steady-state frequency. These simulations are primarily used to analyze low-frequency phenomena, such as inertial responses and frequency regulation. Many different options are exposed for QSP:

- *RMS balanced* or *Positive sequence*: The system is modeled using a single phase, the positive sequence. The models are formulated as DAEs of the general form. These types of simulators are primarily designed for the analysis of machine angles in balanced transmission systems.

- *RMS unbalanced*: These simulations are applied to networks with asymmetrical configurations. The three sequence networks are simplified according to the type of fault being analyzed. Such models are typically used to assess the system's transient response both during and after unbalanced fault events.

- *RMS Dynamic Phasors*: This approach models the network using Fourier dynamic phasors, representing each harmonic component with a subset of differential-algebraic equations. In most RMS applications, the derivative terms of the network are neglected, as they correspond to faster dynamics than those of the connected devices. The model's accuracy depends on the number of harmonics included and on whether the dynamics around each harmonic allow this simplification.

- *dq0-Model with Algebraic Network*: This simulation approach applies a dq0 transformation and assumes steady-state conditions by neglecting the derivative terms of the network. In balanced systems, it is equivalent to the positive-sequence model. For asymmetric networks or when including harmonics, the method can require larger integration steps, and its usefulness is limited due to the resulting time-varying behavior.

The second category, *Electromagnetic Transient (EMT) simulations*: These models incorporate detailed system dynamics to capture fast transient phenomena and deviations from steady-state conditions. They are commonly used to study high-frequency events such as transmission line dynamics, converter switching, machine flux behavior, overvoltages, harmonic propagation, sub-synchronous resonance, transient recovery voltages, and external disturbances like lightning surges. A variety of modeling and computational techniques are employed to balance accuracy with the computational cost of these simulations. Many different options are exposed for EMT:

- *Waveform* or *abc-simulations*: They represent the full voltage and current waveforms throughout the simulation, producing a fully time-varying system. Waveform simulations require careful initialization and selection of integration techniques, and typically employ detailed transmission line models, such as Bergeron or frequency-dependent representations, to capture fast electromagnetic phenomena.

- *dq0-model*: This approach models the network in a rotating reference frame using a transformed π-model. In balanced systems, the model produces a time-invariant formulation and shares the same initialization routine as the positive-sequence RMS model. Since the network is represented using only ODEs, the system becomes a set of stiff differential equations due to the multi-rate dynamics, often necessitating simultaneous-implicit solution methods. However, the advantages of this formulation diminish in unbalanced networks or when harmonics are included, as the model then produces time-varying signals similar to full waveform simulations.

- *Dynamic Phasors*: This method models power system transients using dynamic phasors, ideal for narrow-band signals around harmonic frequencies. It can produce time-invariant models with additional states or time-variant models for advanced analyses like small-signal sensitivity studies. The approach improves computational efficiency when reducing bandwidth outweighs the cost of extra states.

In this thesis, and within the context of the VeraGrid project, the simulations primarily rely on RMS balanced models, which are already implemented in VeraGrid. Additionally, EMT waveform or *abc* domain simulations will be developed as a foundational part of this work.

2.2 Differential-Algebraic Equations (DAEs)

At the core of time-domain simulations lie the mathematical models that describe the system's dynamics. These models are typically formulated as Differential-Algebraic Equations (DAEs), which combine differential equations that represent the system's dynamic behavior with algebraic equations that describe instantaneous relationships between system variables. In contrast, Ordinary Differential Equations (ODEs) are used to model systems where all variables change continuously over time without algebraic constraints. The choice between DAEs and ODEs depends on the specific characteristics of the system being modeled and the phenomena of interest.

In many dynamic systems, including power systems, their model is not only composed of purely differential equations but also algebraic constraints linking the state variables. Such systems are naturally represented by differential-algebraic equations (DAEs). Formally, a DAE is an equation of the form:

$$F\big(\dot{x}(t), x(t), y(t), t\big) = 0 \tag{2.1}$$

Where $x(t) \in \mathbb{R}^n$ are the differential variables, $y(t) \in \mathbb{R}^m$ are algebraic variables, and the function F encodes both the differential and algebraic relations. In contrast to an ordinary differential equation (ODE), not all derivatives $\dot{x}$ can be explicitly solved from the system because some variables appear only algebraically, that is, without derivative [4].

One common formulation is the so-called semi-explicit DAE:

$$\begin{cases} \dot{x} = f\big(x,\, y,\, t\big) \\ 0 = g\big(x,\, y,\, t\big) \end{cases} \tag{2.2}$$

Where $f\colon \mathbb{R}^{n+m+1} \to \mathbb{R}^n$ and $g\colon \mathbb{R}^{n+m+1} \to \mathbb{R}^m$. In this form, one treats some variables x as dynamic and others y as constrained by algebraic equations. The condition to solve the DAE is that the partial jacobian $\partial g/\partial y$ is nonsingular so that y can be locally solved as a function of

x and t [4]. If the partial jacobian is singular an iterative process must be done re-casting the problem until it is possible to solve.

An alternative is the fully implicit form:

$$f\big(\dot{x},\, x,\, y,\, t\big) = 0 \tag{2.3}$$

With no explicit separation of differential and algebraic parts. In that representation, one works directly with the combined vector $(\dot{x}, x, y)$ and the solver must treat the singularity in $\partial f/\partial \dot{x}$. Implicit DAEs are more general and often arise in multiphysics problems and constrained mechanical systems [4].

In power system dynamics, the network and algebraic constraints (Kirchhoff's current law, network admittance equations, etc.) naturally provide algebraic equations, while generator rotor dynamics, excitation systems, governor control, and other dynamic devices supply the differential equations. Thus a typical formulation is

$$\begin{cases} \dot{x} = f(x,\, y) \\ 0 = g(x,\, y) \end{cases} \tag{2.4}$$

Where x includes rotor angles, speed deviations, internal voltages, control states, and y includes bus voltages, phase angles, algebraic currents, and so on [5]. The coupling is strong: every time step, the algebraic network equations must be solved together with the differential updates, typically via Newton-Raphson or other Newton-based solvers applied to the full system jacobian.

This DAE-based representation ensures that the physical constraints of the network are never violated, and that stability and modal analysis reflect the full coupled system behavior, rather than an oversimplified reduced ODE form.

2.3 Symbolic framework

The development of modern power system simulation tools has progressively incorporated symbolic computation to enhance both the analytical transparency and numerical robustness of dynamic models. A symbolic framework refers to an environment in which the mathematical equations describing a physical system are manipulated symbolically before numerical evaluation. This approach enables the explicit representation of the differential and algebraic equations governing power system dynamics, allowing the automatic derivation of jacobians and linearized models directly from the symbolic expressions. Such symbolic processing bridges

the gap between manual analytical formulation and automated numerical computation, ensuring consistency and reproducibility across different analysis tasks [6].

In general, a symbolic modeling framework treats system equations as algebraic entities rather than arrays of numbers. Each state variable, parameter, and equation is represented symbolically and stored in structured form. Once the symbolic system is constructed, operations such as differentiation, substitution, and matrix assembly can be performed analytically. These expressions can then be compiled into efficient numerical routines to be executed by solvers. This two-step process, symbolic formulation followed by numerical evaluation, provides the flexibility to modify models, perform analytical checks and generate adapted code for various simulation modes without redefining the mathematical structure.

Power systems are naturally suited to this formulation, as their behavior is described by a set of coupled DAEs representing both dynamic components and network constraints. In this context, generators, exciters, governors, and control systems contribute differential equations, whereas network variables, bus voltages, currents, and algebraic constraints from Kirchhoff's laws; are represented algebraically. By expressing these components symbolically, they can be automatically assembled into a unified system model and linearized analytically to obtain small-signal representations. This methodology ensures that the global model preserves physical consistency and that linearization results, such as eigenvalues and participation factors, are computed with exact analytical derivatives rather than numerical approximations.

It should be noted that the DAEs representing each element and device in the network can be simplified or extended depending on the desired level of modelling accuracy. A trade-off must therefore be established between computational complexity and numerical accuracy. In practical applications, simplified formulations are often used for large-scale steady-state or RMS simulations, whereas more detailed dynamic representations are required for transient and small-signal stability studies. The flexibility of the symbolic framework allows the modeller to adapt the formulation of each component accordingly, preserving analytical consistency while optimizing computational performance.

2.4 RMS simulations in VeraGrid

In the context of VeraGrid, time-domain simulations have been implemented to model the dynamic behavior of power systems. The framework supports balanced Root Mean Square (RMS) simulations, which are suitable for analyzing low-frequency dynamics in systems dominated by synchronous machines. The implementation includes:

- *Symbolic framework*: A new set of classes is created in order to be able to operate in the symbolic framework

- *Models*: Detailed representations of system components, including generators, transformers, and transmission lines, are developed to capture their dynamic responses accurately.
- *Solver*: The implicit Euler integration method is employed to solve the system's DAEs, ensuring stable and accurate simulation results.
- *Initialization*: Procedures are established to determine the steady-state operating point of the system before introducing disturbances, providing a consistent starting point for dynamic simulations.

This subsection acts as a summary of the already done work on RMS simulations in Veragrid.

2.4.1 Symbolic framework

To implement the symbolic formulation in practice, VeraGrid defines a minimal hierarchy of classes that abstract mathematical entities such as variables, constants, and operators. The main classes are stated below.

- *Expr*: Expression is an abstract class that can be used as any expression for a node. Depending of it's nature it can work as many different objects.
- *Var*: Variables are stored as their name (string) and their value set as Number (integer, float or complex number) or *Expr*. This allows values stored in *Var* to be able to change. Some of the variables can be assigned as *Dynamic variable* which means those variables are part of an other device but are also used in the DAEs of the device created.
- *Const*: Constants are stored as their name (string) and their value set as Number (integer, float or complex number). Those are frozen parameters of the grid.
- *BinOp*: Binary operator stores each possibility of binary operation. The following arithmetic operations are defined: addition, subtraction, multiplication, division and exponentiation.
- *Func*: Functions store each individual function possible that can be part of an equation: sinus, cosinus, tangent, logarithm, real or imaginary parts among others.

All these elements are unified to create *Blocks*, a set of differential and algebraic equations and stated variables that form the symbolic representation of the DAEs of each element. Blocks also include a set of initializing variables and equations and fixed variables and equations that help initialize the DAEs and allow a fastest solution. All variables in a Block are stored as children of that Block in order to allow recursivity. This recursive structure means that each block can contain child elements (either variables or other blocks) allowing a hierarchical representation of the power system. Such a design enables complex devices and networks to be constructed

from smaller symbolic components while maintaining full analytical traceability. Moreover, parameters are also defined as the variables or constants affected by events that can be added to the grid.

2.4.2 RMS models

VeraGrid's version of the symbolic framework includes three different types of elements in the grid: buses, injections and branches. Injections include all types of elements that contribute actively to the system: generators, loads, batteries, shunts... Branches include the devices that connect buses: lines, transformers, converters...

In this subsection, the RMS models of the grid elements in VeraGrid are exposed.

Injections

- **Static load** The model for the static load is quite simple since P and Q are meant to be constant. Therefore, the algebraic equations corresponding to the load are only initializing the power to its value.

$$\begin{cases} P_l - P_{l,0} = 0 \\ Q_l - Q_{l,0} = 0 \end{cases} \tag{2.5}$$

  Where:

  - P_l: (Algebraic variable) Active power consumed by the load [MW].
  - Q_l: (Algebraic variable) Reactive power consumed by the load [Mvar].
  - $P_{l,0}$: (Algebraic variable) Active power consumed by the load at time 0 [MW].
  - $Q_{l,0}$: (Algebraic variable) Reactive power consumed by the load at time 0 [Mvar].

- **Shunt** For the shunt, the algebraic equations are also simple. They define the formula for the active and reactive power.

$$\begin{cases} P_{sh} - gV^2 = 0 \\ P_{sh} - bV^2 = 0 \end{cases} \tag{2.6}$$

  Where:

  - P_{sh}: (Algebraic variable) Active power injected or absorbed by the shunt [MW].
  - Q_{sh}: (Algebraic variable) Reactive power injected or absorbed by the shunt [Mvar].

- g: (Constant parameter) Shunt conductance [p.u.].
- b: (Constant parameter) Shunt susceptance [p.u.].
- V: (Constant parameter) Shunt voltage [p.u.].

- **Synchronous generator** The synchronous generator is defined by a set of differential(or state) and algebraic equations.

$$\begin{cases} \delta = 2\pi f\,(\omega - \omega_{ref}) \\ \omega = \dfrac{\Gamma_m - \Gamma_e - D(\omega - \omega_{ref})}{M} \end{cases} \tag{2.7}$$

$$\begin{cases} \psi_d - (R \cdot i_q + v_q) = 0 \\ \psi_q + (R \cdot i_d + v_d) = 0 \\ 0 - (\psi_d + X \cdot i_d - v_f) = 0 \\ 0 - (\psi_q + X \cdot i_q) = 0 \\ v_d - (V \cdot \sin(\delta - \theta)) = 0 \\ v_q - (V \cdot \cos(\delta - \theta)) = 0 \\ \Gamma_e - (\psi_d \cdot i_q - \psi_q \cdot i_d) = 0 \\ P_g - (v_d \cdot i_d + v_q \cdot i_q) = 0 \\ Q_g - (v_q \cdot i_d - v_d \cdot i_q) = 0 \\ \Gamma_m - (\Gamma_{m0} + K_p(\omega - \omega_{ref}) + K_i \cdot e_t) = 0 \\ 2\pi f \cdot e_t - \delta = 0 \end{cases} \tag{2.8}$$

Where:

- δ: (State variable) Rotor electrical angle with respect to the synchronous reference frame [rad].
- ω: (State variable) Rotor angular speed [rad/s].
- P_g: (Algebraic variable) Active electrical power delivered by the generator [MW].
- Q_g: (Algebraic variable) Reactive electrical power delivered by the generator [Mvar].
- v_d: (Algebraic variable) d-axis stator voltage component in Park reference frame [p.u.].
- v_q: (Algebraic variable) q-axis stator voltage component in Park reference frame

[p.u.].

- i_d: (Algebraic variable) d-axis stator current component in Park reference frame [p.u.].
- i_q: (Algebraic variable) q-axis stator current component in Park reference frame [p.u.].
- ψ_d: (Algebraic variable) d-axis stator flux linkage in Park reference frame [p.u.].
- ψ_q: (Algebraic variable) q-axis stator flux linkage in Park reference frame[p.u.].
- Γ_e: (Algebraic variable) Electromagnetic torque produced by the generator [p.u.].
- Γ_m: (Algebraic variable) Mechanical torque applied to the rotor [p.u.].
- e_t: (Algebraic variable) Integral term of the governor controller, representing the accumulated speed error [rad/s].
- V: (Algebraic variable from the bus) Magnitude of the terminal voltage phasor [p.u.].
- θ: (Algebraic variable from the bus) Angle of the terminal voltage phasor with respect to the reference frame [rad].
- $\Gamma_{m,0}$: (Fixed variable) Initial mechanical torque reference [p.u.].
- f: (Constant parameter) Nominal electrical frequency [Hz].
- D: (Constant parameter) Damping coefficient representing mechanical damping effects [p.u.].
- M: (Constant parameter) Inertia constant of the rotor [p.u.·s].
- R: (Constant parameter) Stator winding resistance [p.u.].
- X: (Constant parameter) Stator synchronous reactance [p.u.].
- v_f: (Constant parameter) Field excitation voltage [p.u.].
- K_p: (Constant parameter) Proportional gain of the speed governor control [-].
- K_i: (Constant parameter) Integral gain of the speed governor control [-].

Branches

- **Line** The line algebraic equations are the active and reactive power equations correspond-

ing to the π model of lines. In this case, the RMS model for the lines is also valid for the transformers.

$$\begin{cases} P_f &= V_f^2\, g - g\, V_f V_t \cos(\theta_f - \theta_t) + b\, V_f V_t \cos\left(\theta_f - \theta_t + \frac{\pi}{2}\right) \\ Q_f &= V_f^2 \left(-\frac{b_{sh}}{2} - b\right) - g\, V_f V_t \sin(\theta_f - \theta_t) + b\, V_f V_t \sin\left(\theta_f - \theta_t + \frac{\pi}{2}\right) \\ P_t &= V_t^2\, g - g\, V_t V_f \cos(\theta_t - \theta_f) + b\, V_t V_f \cos\left(\theta_t - \theta_f + \frac{\pi}{2}\right) \\ Q_t &= V_t^2 \left(-\frac{b_{sh}}{2} - b\right) - g\, V_t V_f \sin(\theta_t - \theta_f) + b\, V_t V_f \sin\left(\theta_t - \theta_f + \frac{\pi}{2}\right) \end{cases} \tag{2.9}$$

Where:

- P_f: (Algebraic variable) Active power at the beginning of the line (at the bus defined as beginning) [MW].
- Q_f: (Algebraic variable)Reactive power at the beginning of the line (at the bus defined as beginning) [Mvar].
- P_t: (Algebraic variable) Active power at the end of the line (at the bus defined as end) [MW].
- Q_t: (Algebraic variable)Reactive power at the end of the line (at the bus defined as end) [Mvar].
- V_f: (Algebraic variable) Voltage magnitude at the beginning of the line (at the bus defined as beginning) [p.u.].
- V_t: (Algebraic variable) Voltage magnitude at the end of the line (at the bus defined as end) [p.u.].
- θ_f: (Algebraic variable) Voltage angle at the beginning of the line (at the bus defined as beginning) [rad].
- θ_t: (Algebraic variable) Voltage angle at the end of the line (at the bus defined as end) [rad].
- g: (Constant parameter) Line conductance [p.u.].
- b: (Constant parameter) Line susceptance [p.u.].
- b_{sh}: (Constant parameter) Shunt susceptance [p.u.].

Bus The buses store the conservation equations itself. Those are algebraic equations in 2.10. For example in a bus where a generator is connected, the power that the generator delivers to the bus must be the same as the power in the line connected to the bus. However, the variables associated to the bus are the voltage magnitude V and angle θ. All the variables in the bus algebraic equations are defined as *dynamic variables* since they also appear on other devices (such as the generators or lines connected to the bus) so those must be able to be callable from all the device's blocks.

$$\begin{cases} \sum_i P_{inj} + \sum_i P_{branch} = 0 \\ \sum_i Q_{inj} + \sum_i Q_{branch} = 0 \end{cases} \tag{2.10}$$

Where:

- P_{inj}: (Algebraic variable) Active power from injections. If the injections are generators the power will be positive and if they are loads, it will be negative [MW].
- Q_{inj}: (Algebraic variable) Reactive power from injections. If the injections are generators the power will be positive and if they are loads, it will be negative [Mvar].
- P_{branch}: (Algebraic variable) Active power that is transported from or to the line connected to the bus [MW].
- Q_{branch}: (Algebraic variable) Reactive power that is transported from or to the line connected to the bus [Mvar].

2.4.3 Integration method: Implicit Euler

In the numerical integration of differential or differential-algebraic equations (DAEs), Euler methods represent one of the most fundamental approaches for approximating the time evolution of system variables. They discretize the continuous time domain into a series of steps of size h, and approximate the derivative using finite differences. In RMS power system simulations, which combine slow electromechanical dynamics with algebraic network equations, Euler methods are often used to understand the basic principles of time integration before moving to more advanced schemes [7].

The explicit, or forward, Euler method estimates the derivative at the current time step t_i and updates the state variable y according to:

$$y_{i+1} = y_i + h \left. \frac{\delta y}{\delta t} \right|_{t_i,\, y_i} \tag{2.11}$$

This simple expression is obtained by integrating the differential equation $\dot{y} = f(t, y)$ over the

interval $[t_i, t_{i+1}]$ and approximating the derivative by its value at t_i. The method is straightforward to implement and computationally inexpensive, as the right-hand side depends only on known quantities at the current time step. However, it is *conditionally stable*, which means that the time step h must remain below a certain limit for the numerical solution to remain stable [7].

For stiff systems, those in which some variables evolve much faster than others, this stability condition becomes extremely restrictive. In power system models, where fast control loops and electromagnetic effects coexist with slow mechanical dynamics, the forward Euler method becomes inefficient and may even diverge if the step size is not sufficiently small [2].

On the contrary, the implicit, or backward, Euler method overcomes these limitations by evaluating the derivative at the next time step t_{i+1}:

$$y_{i+1} = y_i + h \left. \frac{\delta y}{\delta t} \right|_{t_{i+1},\, y_{i+1}} \tag{2.12}$$

This can alternatively be expressed as:

$$y_i = y_{i+1} - h \left. \frac{\delta y}{\delta t} \right|_{t_{i+1},\, y_{i+1}} \tag{2.13}$$

Since the derivative depends on the unknown future value y_{i+1}, this approach is called implicit. Solving for y_{i+1} requires finding the root of a nonlinear equation, which is typically done using an iterative method such as Newton-Raphson [8]. Although this increases the computational cost per time step, the backward Euler method provides unconditional stability for linear systems and is therefore well-suited for stiff problems [7].

Intuitively, the method tends to damp fast transients, which makes it ideal for systems dominated by both fast and slow dynamics, as in power systems. Its numerical damping helps maintain stability even with relatively large time steps, although at the expense of reduced accuracy, since the method is only first order.

Applying the implicit Euler method to DAE systems:

$$\begin{cases} \dfrac{x_{i+1} - x_i}{h} = f(x_{i+1}, y_{i+1}) \\ 0 = g(x_{i+1}, y_{i+1}) \end{cases} \tag{2.14}$$

which forms a set of nonlinear algebraic equations in the unknowns (x_{i+1}, y_{i+1}). This system is solved iteratively at each time step using Newton-Raphson, requiring the evaluation of the Jacobian matrix that couples the differential and algebraic components [8].

Despite its simplicity and low order of accuracy, the backward Euler method remains one of the

most robust integration schemes for stiff power system DAEs. Its unconditional stability makes it particularly suitable for RMS simulations involving large time constants or strong coupling between network and control equations [2].

2.4.4 Initialization

coses de pablo i marina

3 Small-signal stability analysis

3.1 Theoretical background

3.1.1 Power system stability

Power system stability is defined as that property of a power system that enables it to remain in a state of operating equilibrium under normal conditions and to regain an acceptable state of equilibrium after being subjected to a disturbance [9]. Other definitions state not only the state of equilibrium must be acceptable but also most system variables must be bounded so that practically the entire system remains intact[10].

Although the primary concern is the behavior of the interconnected system as a whole, the stability of individual components such as generators, motor loads, or regional subsystems; can be equally significant, particularly when localized instability does not propagate to the broader network. The system's dynamic behavior is governed by nonlinear interactions among its elements, and its response to perturbations is influenced by both the prevailing operating conditions and the specific nature of the disturbance. Stability is understood around an equilibrium point and is subject to change under small or large disturbances.

Power system stability is commonly classified depending on the physical nature of the instability, the size of the disturbance considered and the devices, processes and time span that must be considered to assess stability [10]. The combination of these factors influence the methodologies, tools and considerations used in the analysis. The main categories of power system stability are described in the following enumeration.

- *Rotor angle stability*: The ability the ability of the interconnected synchronous machines in a power system to remain in synchronism under normal operating conditions and to regain synchronism after being subjected to a small or large disturbance [9]. A synchronous machine stays in synchronism when the electromagnetic torque exactly balances the mechanical torque from the prime mover, producing zero net accelerating torque. Stability therefore depends on the machine and its controls restoring that torque balance after a disturbance; failure to do so causes rotor acceleration or deceleration and loss of synchronism.

- *Voltage stability*: The capacity of the network to maintain acceptable voltage magnitudes at all buses during normal operation and following disturbances, such that voltages do not decrease sustainedly. Loss of this capacity manifests as progressive voltage drops and, ultimately, voltage collapse, which may force extensive load disconnection or generalized service interruption.

- *Frequency stability*: The capability of the system to preserve a near-nominal frequency fol-

lowing a major imbalance between generation and demand, through inertial response and secondary/tertiary control actions. Inadequate frequency stability results in sustained under-frequency or over-frequency transients that can damage equipment, trigger protective disconnections, and precipitate broader system failure.

Due to the increasing penetration of power electronics into the grid, two new categories of stability have been considered [11].

- *Converter-driven stability*: Refers to oscillatory behavior caused by control interactions in converter-interfaced generation (CIG). Fast-interaction instabilities arise from high- frequency dynamics (hundreds of Hz to kHz) involving inner control loops and grid components. Slow-interaction instabilities occur at low frequencies (<10 Hz), driven by PLL and outer-loop controls, especially in weak grids. Synchronization issues and power transfer limits further compromise stability. These phenomena differ from classical generator dynamics and require tailored mitigation strategies

- *Resonance stability*: Refers to the system's ability to withstand oscillatory energy exchange without magnifying voltage, current, or torque beyond safe limits. It includes subsynchronous resonance (SSR), which arises from interactions between series compensation and either mechanical shaft modes or electrical generator characteristics. The mechanical form leads to torsional resonance, while the electrical form—9 (Induction Generator Effect (IGE)) can cause self-excitation. Converter controls in DFIGs can exacerbate these effects, leading to subsynchronous control interaction (SSCI). These phenomena pose risks to both mechanical integrity and electrical equipment.

Rotor angle stability is inherent of classical power systems with synchronous machines. Modern power grids, with the increasing penetration of power electronics, have introduced new dynamics and interactions that can affect rotor angle stability. Converters decrease the inertia of the system and have an effect on the electromechanical modes. However, the fundamental principles of rotor angle stability remain intact and still takes a crucial role on stability analysis of power systems.

Therefore, understanding and analyzing rotor-angle stability remains essential for ensuring the overall reliability of power systems. Insufficient or negative synchronizing torque produces aperiodic, non-oscillatory transient instability that drives large rotor-angle deviations and is typically studied with time-domain numerical integration. In contrast, the absence of adequate damping torque gives rise to small-disturbance oscillatory instability.

In the context of this thesis, the focus is on rotor angle stability, particularly small-signal stability, its eigenvalue-based characterization, modal properties, and analysis methods. The following sections describe the theoretical background and methodologies used for small-signal stability

analysis in power systems.

3.1.2 Stability of a dynamic system

A dynamic system is considered stable if, when subjected to a disturbance, it returns to its original state or to a new equilibrium state without exhibiting unbounded behavior. The equilibrium points are those states where all the derivatives $\dot{x}$ are zero, meaning the system is at rest or in a steady state.

Linearity affects on the stability of a system. The stability of a liner system is independent of the input and the initial conditions. However, for a non-linear system, stability depends on the magnitude of the input and initial conditions. Depending on the region of the state-space, stability is classified into the following categories:

- *Local stability*: the system is locally stable around an equilibrium point if when a small perturbation is applied, it remains around the equilibrium point. If as time increases it returns to the equilibrium point, it is locally asymptotically stable[9].
- *Finite stability*: the system is finitely stable if when a perturbation of finite size is applied, it remains bounded and does not diverge to infinity.
- *Global stability*: the system is globally stable if it returns to an equilibrium point for any initial condition in the whole state-space.

Therefore, linearizing a non-linear system around an equilibrium point allows to study its local stability as if it was a linear system.

3.1.3 Small-signal stability

Small-signal stability refers to the ability of a power system to maintain synchronism when subjected to small disturbances [9], such as minor load changes or small faults. These small disturbances (typically within 1%)occur frequently in power systems and allow the linearization of non-linear system equations around a specific operating point in order to perform analysis. The resulting linear representation enables the use of standard control engineering tools to assess system stability and dynamic performance [12].

The resulting instability due to those small perturbations can have two forms: Non-oscillatory unstability defined as an increase in rotor angle due to insufficient synchronizing torque and oscillatory unstability, oscillations of increasing magnitude due to insufficient damping torque [9]. In practice, most of the instabilities come from insufficient damping torque. The following list summarizes the main oscillatory modes to consider:

- *Local modes*: Oscillations involving individual generators or small groups of units swing-

ing against the rest of the system, typically localized near a generating station.

- *Inter area modes*: Low-frequency oscillations between large groups of generators in different regions, often linked by weak transmission corridors.
- *Control modes*: Oscillations arising from interactions between poorly tuned control systems—such as exciters, speed governors, HVDC converters, or static var compensators.
- *Torsional modes*: Oscillations associated with the mechanical shaft system of turbine-generators, which may become unstable due to interactions with control systems or series-compensated transmission lines.

Although small-signal analysis only applies to small variations around a fixed operating point, it remains a practical and widely used method for studying power system dynamics. By linearizing the system, it allows to apply control theory tools like eigenvalue analysis and state-space modeling. This helps identify poorly damped modes and assess how the system responds to small disturbances. Despite its limitations, it is a reliable approach for early detection of potential instabilities and for designing stabilizing controls.

3.1.4 State-space representation

Small-signal stability assessment methods are generally categorized into two main groups: state-space techniques and frequency-domain techniques. State-space techniques allow one to represent the system using a set of first order differential equations written in the following form:

$$\dot{x} = f(x, u, t) \tag{3.1}$$

Where x is the state column vector that stores the state variables, u is the input column vector that stores external signals that influence the system performance and t is time. The system can also not depend on time, that system is called time-invariant. It is also important to note that state variables are the minimum amount of variables needed to represent the system and be able to compute its future behavior.

Often the purpose of the state-space representation is to look at a set of the system variables, called outputs y. Then, a new expression is added to the state-space representation:

$$y = g(x, u, t) \tag{3.2}$$

Where y is the output column vector that stores the output variables.

A dynamic system can be described in many different ways depending on which variables are chosen as states, inputs, and outputs. These choices shape how the system behaves mathemati-

cally and how easily can it be analyzed. For example, using electrical quantities like voltage and current might be more practical for converter models, while mechanical variables such as rotor angle and speed are better suited for synchronous machines. The flexibility in selecting these variables allows to adapt the model to the specific goals of the study while still ensuring that the essential dynamics of the system are captured.

State-space models are commonly represented in their matrix formulation as follows:

$$\dot{x} = Ax + Bu \tag{3.3}$$

$$y = Cx + Du \tag{3.4}$$

Where:

- x : state variables vector
- u : system inputs vector
- y : outputs vector
- A : state matrix
- B : input matrix
- C : output matrix
- D : direct transmission matrix

Linearization of state-space models In order to linearize a non-linear state-space model, a small perturbation is applied around the operation point (equilibrium point).

$$\mathcal{X} \triangleq x - x^* \tag{3.5}$$

$$\mathcal{Y} \triangleq y - y^* \tag{3.6}$$

$$\mathcal{U} \triangleq u - u^* \tag{3.7}$$

Where x^*, y^* and u^* are the state, output and input vectors at the equilibrium point respectively. The new linearized state-space model is given by:

$$\dot{\mathcal{X}} = A\mathcal{X} + B\mathcal{U} \tag{3.8}$$

$$\mathcal{Y} = C\mathcal{X} + D\mathcal{U} \tag{3.9}$$

Where the new matrices are computed as the Jacobian matrices of the non linear system evaluated at the equilibrium point:

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \dots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \dots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \tag{3.10}$$

$$B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \dots & \frac{\partial f_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial u_1} & \dots & \frac{\partial f_n}{\partial u_n} \end{bmatrix} \tag{3.11}$$

$$C = \begin{bmatrix} \frac{\partial g_1}{\partial x_1} & \dots & \frac{\partial g_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial x_1} & \dots & \frac{\partial g_n}{\partial x_n} \end{bmatrix} \tag{3.12}$$

$$D = \begin{bmatrix} \frac{\partial g_1}{\partial u_1} & \dots & \frac{\partial g_1}{\partial u_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial g_n}{\partial u_1} & \dots & \frac{\partial g_n}{\partial u_n} \end{bmatrix} \tag{3.13}$$

For simplicity, the perturbation notation is often omitted, and the linearized state-space model is expressed as:

$$\Delta\dot{x} = A\Delta x + B\Delta u \tag{3.14}$$

$$\Delta y = C\Delta x + D\Delta u \tag{3.15}$$

3.1.5 DAE to state-space representation

In power systems, the dynamic behavior is mathematically represented by a set of Differential-Algebraic Equations (DAEs) that capture the interaction between dynamic components and network constraints. This formulation naturally arises because power systems combine elements with both dynamic and instantaneous responses.

- *Differential equations*: describe the time-dependent evolution of state variables associated with components that possess energy storage or control dynamics. These include synchronous generators (rotor angle and speed dynamics), excitation systems, governors, power electronic converters, and various control loops such as voltage and frequency regulators.

- *Algebraic equations*: represent the instantaneous electrical relationships and constraints imposed by the network. They stem primarily from Kirchhoff's laws, ensuring power balance and voltage-current consistency at each bus, as well as from static components like loads, transmission lines, and transformers, which are assumed to reach steady-state conditions instantaneously.

The explicit formulation of the DAE system is given by:

$$T\dot{x} = f(x, y) \tag{3.16}$$

$$0 = g(x, y) \tag{3.17}$$

Which is linearized around an equilibrium point as follows:

$$T\Delta\dot{x} = \frac{\delta f}{\delta x}\Delta x + \frac{\delta f}{\delta y}\Delta y \tag{3.18}$$

$$0 = \frac{\delta g}{\delta x}\Delta x + \frac{\delta g}{\delta y}\Delta y \tag{3.19}$$

From the second equation, Δy can be expressed in terms of Δx:

$$g_y\Delta y = -g_x\Delta x \rightarrow \Delta y = -g_y^{-1}g_x\Delta x \tag{3.20}$$

And then substituted into the first equation:

$$T\Delta\dot{x} = f_x\Delta x + f_y(-g_y^{-1}g_x\Delta x) \tag{3.21}$$

Rearranging the equation gives the linearized state-space representation and the expression for the state matrix A:

$$\Delta\dot{x} = T^{-1}(f_x - f_y g_y^{-1}g_x)\Delta x \rightarrow A = T^{-1}(f_x - f_y g_y^{-1}g_x) \tag{3.22}$$

The A matrix encapsulates the dynamic interactions between the system's state variables, accounting for both the intrinsic dynamics of the components and the constraints imposed by the network. From the state matrix the stability assessment can be performed as explained in the next section.

3.1.6 Stability assessment: Liapunov's first method

The stability of a system can be studied in large-signal and small-signal therms. Stability *in the large* needs to study the whole non-linear system. This method is complex and requires a high computational effort. On the other hand, stability *in the small* studies the system behavior around an equilibrium point. This method is simpler and less computationally intensive, but it only provides information about the local stability of the system [9]. Computing the eigenvalues of the state matrix A allows to determine the small-signal stability of the system.

Eigenvalue analysis and participation factors (PF) are key tools for identifying dominant modes and evaluating system stability. These methods are well established in conventional power systems and are increasingly being applied to power-electronics-based systems, where dynamic behavior is often more complex and sensitive to operating conditions.

The eigenvalues λ of the state matrix A, commonly referred to as the system's modes, characterize its small-signal stability according to the following criteria:

- All modes satisfy $Re(\lambda) < 0$: the system is asymptotically stable
- All modes satisfy $Re(\lambda) \leq 0$: the system is marginally stable
- At least one mode satisfies $Re(\lambda) > 0$: the system is unstable

In this way, for the stable poles, the real part defines the time constant as $\tau = -1/Re(\lambda)$ [s] which represents the time that passes until the 63,2% of the steady state value is achieved, said in other words, the time that passes for the oscillations to decay by a 37%.

When a linearized system has complex conjugate modes, they represent oscillatory modes in the dynamic response:

- The real part determines damping:
  - $Re(\lambda) < 0$: exponential decay
  - $Re(\lambda) = 0$: oscillations persist indefinitely
  - $Re(\lambda) > 0$: exponential growth
- The imaginary part determines the oscillation frequency defined as: $f = \frac{Im(\lambda)}{2\pi}$

An other way to look at the damping of a mode is through the damping ratio ζ. The damping ratio is a dimensionless measure that describes how oscillations in a system decay after a disturbance. It is defined as the ratio of actual damping to critical damping. The critical damping is the minimum amount of damping that prevents oscillations. The damping ratio is given by:

$$\zeta = -\frac{Re(\lambda)}{\sqrt{Re(\lambda)^2 + Im(\lambda)^2}} \tag{3.23}$$

Where $Re(\lambda)$ is the real part of the eigenvalue and $Im(\lambda)$ is the imaginary part of the eigenvalue. The interpretation of the damping ratio is described below.

- $\zeta < 0$: *Unstable oscillations.* The system exhibits modes that grow exponentially with time,

caused by eigenvalues located in the right half of the complex plane.

- $\zeta = 0$: *Marginal stability.* The system produces undamped, sustained oscillations since the eigenvalues lie exactly on the imaginary axis.
- $0 < \zeta < 1$: *Stable oscillatory response.* The system returns to its equilibrium point through oscillations that gradually decay over time. In practical terms, a damping ratio of about $\zeta = 0.05$ is generally considered sufficient for well-damped behaviour.
- $\zeta = 1$: *Critical damping*. The system returns to equilibrium without oscillations, reaching the steady state in the shortest possible time without overshoot.

Finally, participation factors quantify the relative influence of each state variable on the different dynamic modes of the system. In essence, they indicate how much a given state contributes to a specific mode and, conversely, how strongly that mode affects the state. This dual interpretation makes participation factors a valuable tool for understanding the internal structure of system dynamics [13].

In the context of power systems, participation factors play a key role in identifying the physical origin of oscillations and instabilities. By analyzing these factors, it is possible to determine which components—such as generators, controllers, or converter units—are most involved in poorly damped or unstable modes. This information supports targeted actions for control tuning, model validation, and stability improvement. Participation factors are calculated as follows:

$$PF_{i,k} = W_{i,k} \cdot V_{i,k} \tag{3.24}$$

Where:

- $PF_{i,k}$ is the participation factor of the k-th state variable to the i-th mode.
- $W_{i,k}$ is the left eigenvector of the k-th state variable to the i-th mode of matrix A. It satisfies $w^T A = \lambda w^T$.
- $V_{i,k}$ is the right eigenvector of the k-th state variable to the i-th mode of matrix A. It satisfies $Av = \lambda v$.

The graphical representation of the eigenvalues in the complex plane, provides a visual tool for assessing system stability. Then, it is possible to identify the stability of the system and the oscillation of the modes at a glance. An example of an eigenvalue plot is shown in Figure 3.1.

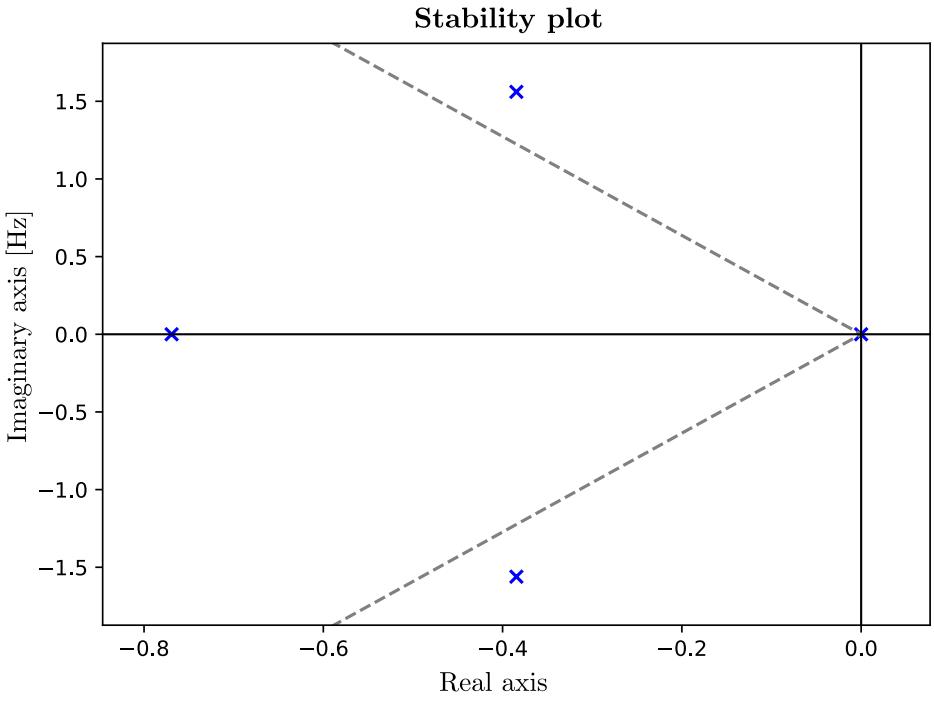

Figure 3.1: Eigenvalue plot example.

The imaginary axis divide the stable part (negative real part) from the unstable part (positive real part). Therefore, all the eigenvalues represented in the Figure are stable except for one which is in the origin, therefore the system is marginally stable. Moreover, modes outside the real axis represent oscillatory modes. In this case, one can see that there are two complex conjugate eigenvalues are represented, which means that the system has an oscillatory mode. the mode in the real axis is a non-oscillatory mode. Since it is the one with the highest real part, it is the dominant mode of the system.

3.2 Implementation

The implementation of the small-signal stability analysis is divided mainly into two parts: the code development and the GUI implementation.

3.2.1 Code development

The small-signal stability analysis is implemented in Python, following the general structure of the dynamic simulation framework in Veragrid. The general simulation structure for any simulation in VeraGrid is shown in Figure 3.2.

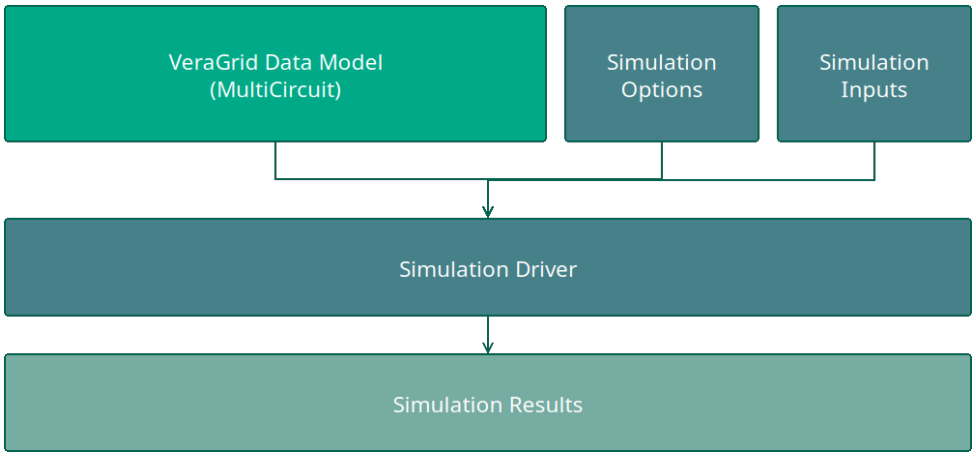

Figure 3.2: General simulation structure. *Source*: VeraGrid documentation [1].

Therefore, each simulation has three main classes explained below:

- **Driver:** This class is responsible for running the simulation. It contains the simulation analysis function itself that computes the results and the run function that is called from the GUI, and it is responsible for executing the simulation.

  The block diagram of the small-signal stability analysis simulation function is shown in Figure 3.4. It consists on the A matrix computation from the jacobian matrix, the eigenvalues computation and then 2 main postprocesses: the participation factors computation and the damping ratios and oscillation frequencies computation.

  The full simulation block diagram is shown in Figure 3.3 where one can see the main steps of the simulation from importing the system data to the final results. The approach given is to give the user the option to run the small-signal stability analysis whenever he/she wants in the dynamic simulation. The user can initialize the system, check the stability in the first operation point, add an event and then check the stability again in the new operation point just choosing the assessment time.

- **Results:** This class is responsible for storing the results of the simulation. It contains the data structure that holds the results and the way they are accessed from the GUI. In this case, the results class show three main results: the eigenvalues with their corresponding damping ratios and oscillation frequencies, the participation factors matrix with the corresponding eigenvalues and states for each column and row respectively, and the complex plane plot of the eigenvalues in bot rad/s and Hz for the imaginary axis.

- **Options:** This class is responsible for defining the options of the simulation. It contains the parameters that can be set by the user from the GUI. In this case, the options class

allows the user to set the assessment time, and in case the dynamic simulation is needed, the options for the simulation: integration method, time step and tolerance.

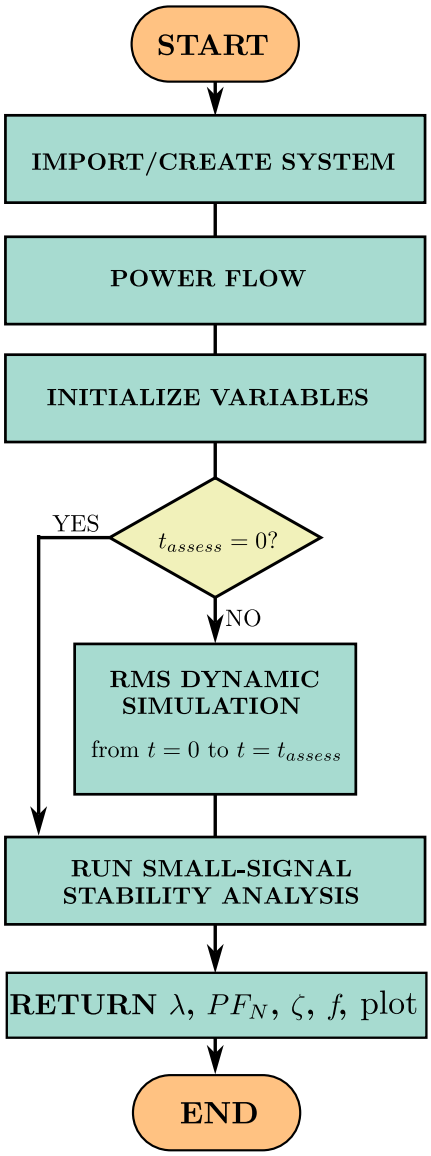

Figure 3.3: Small-signal stability analysis full simulation block diagram.

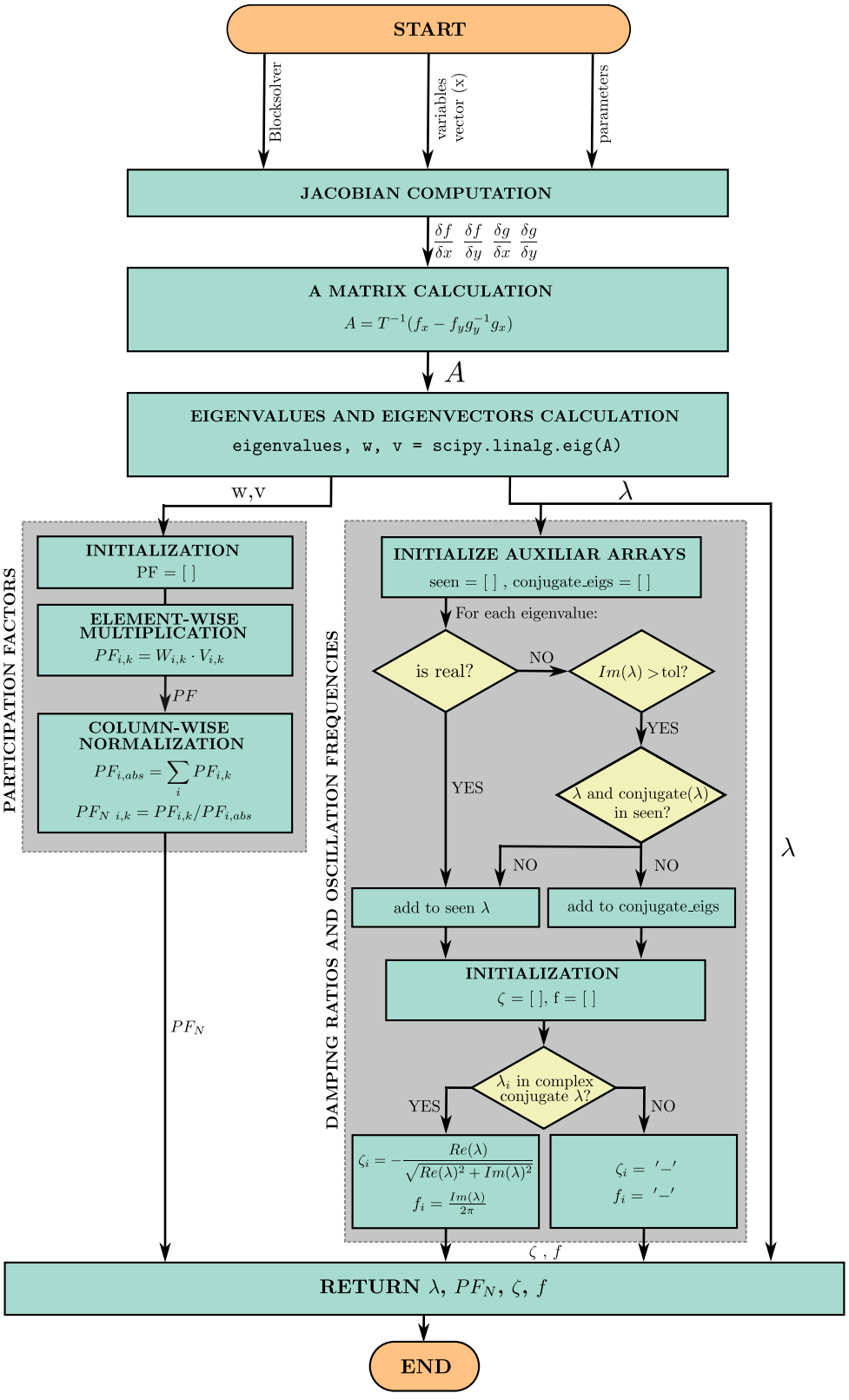

Figure 3.4: Small-signal stability analysis computation block diagram.

3.2.2 GUI implementation

The GUI is creased using the open source program Qt Designer [14], which allows to create the GUI using a drag and drop interface. The GUI implementation consists on creating a new settings page for the small-signal stability analysis, adding the option to run the simulation in the main tools bar, and creating a new results page to show the results of the simulation. The GUI implementation is not only adding the new pages, but also connecting the GUI with the code developed in Python.

However, the first initial step is to choose an icon for the small-signal stability analysis that will be shown in the tools bar and will help users identify the option. The icon chosen is shown in Figure 3.5. The icon represents a magnifying glass looking at a wave. The magnifying glass represents the small-signal analysis, which works around a small perturbation and a small interval around the operating point, so the user needs the magnifying glass to see those small perturbations. The wave represents the system variables represented in the time domain, which are the ones that will be analyzed in the small-signal stability analysis.

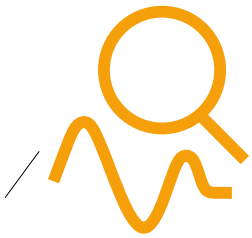

Figure 3.5: Icon for the small-signal stability analysis.

The settings page shown in Figure 3.6 allows the user to set the parameters needed for the small-signal stability analysis. It is noted how the small-signal settings are added into the dynamic simulation settings. The settings shown are explained in the list below.

1. *Integration method*: The integration method to use if the RMS dynamic simulation is performed. There are two options: trapezoidal or implicit euler.

2. *Tolerance*: per-unit error tolerance to use in the integration method. Only needed if the Rms dynamic simulation is performed.

3. *Assessment time* (s): The time instant in seconds where the stability assessment is performed.

4. *Time step* (s): Step size in seconds between each numerical evaluation in the integration method. Smaller intervals increase accuracy but require more computation. Only needed

if the RMS dynamic simulation is performed.

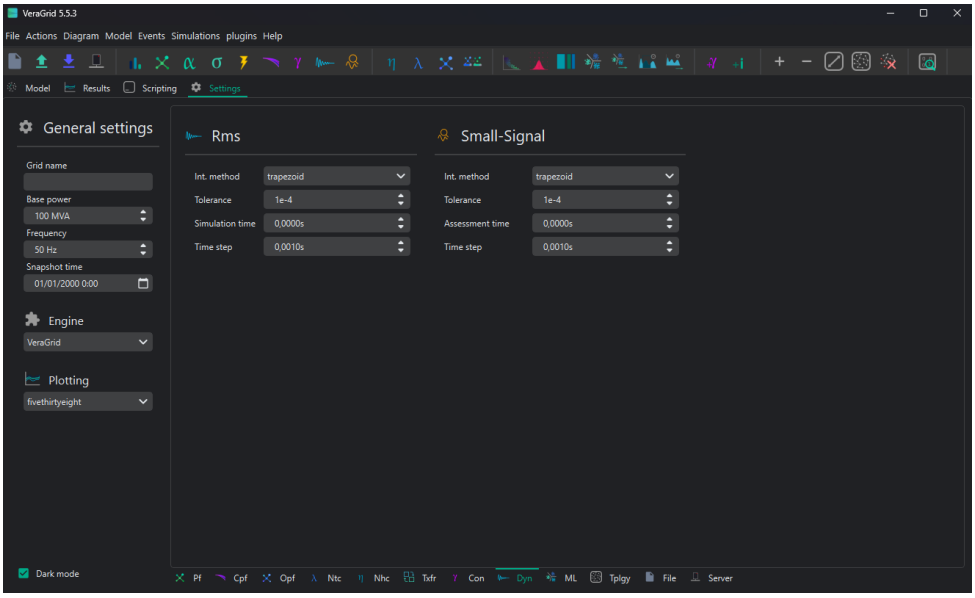

Figure 3.6: Small-signal stability analysis settings page.

The small-signal stability analysis can only be performed if the power flow has been computed, as it is needed to obtain the operating point of the system.

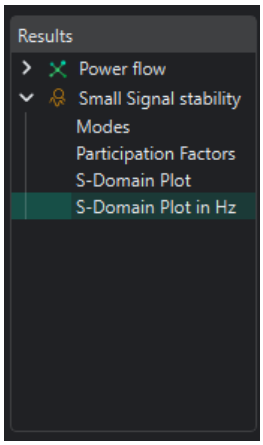

Figure 3.7: Results dropdown.

The results page shows the results of all the simulations performed in VeraGrid. The user can choose the results to show using the dropdown menu shown in Figure 3.7. In this case, the power flow and small-signal stability analysis results are shown. As one can see, the small-signal results are divided into 4 options: the modes table, the participation factors table, the complex domain plot and the complex domain plot in Hz. These results are shown in Figures 3.8, 3.9, 3.10 and 3.11 respectively.

	Real	Imaginary	Damping ratio	Oscillation frequency
0: Mode 0	-1.9999999999999991	43.134764875209456	0.046316553039368706	6.865111048996247
1: Mode 1	-1.9999999999999991	-43.134764875209456	-	-
2: Mode 2	-2.000000000000002	22.315295098173767	0.08926682293398204	3.551589521428697
3: Mode 3	-2.000000000000002	-22.315295098173767	-	-
4: Mode 4	1.1675638183136432e-14	0.0	-	-
5: Mode 5	-2.0000000000000013	16.807701792552095	0.11815947755298381	2.675028822299176
6: Mode 6	-2.0000000000000013	-16.807701792552095	-	-
7: Mode 7	-1.9999999999999998	12.979801242012693	0.15228832723936697	2.0657995280166426
8: Mode 8	-1.9999999999999998	-12.979801242012693	-	-
9: Mode 9	-4.000000000000011	0.0	-	-

Figure 3.8: Table result with the modes, damping ratio and oscillation frequencies.

	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6	Mode 7	Mode 8	Mode 9
0: deltagen 0	0.187576	0.187576	0.161546	0.161546	0.184218	0.014712	0.014712	0.044056	0.044056	0.000000
1: omegagen 0	0.187576	0.187576	0.161546	0.161546	0.000000	0.014712	0.014712	0.044056	0.044056	0.184218
2: deltagen 1	0.302303	0.302303	0.047806	0.047806	0.195215	0.013639	0.013639	0.038644	0.038644	0.000000
3: omegagen 1	0.302303	0.302303	0.047806	0.047806	0.000000	0.013639	0.013639	0.038644	0.038644	0.195215
4: deltagen 2	0.009750	0.009750	0.287054	0.287054	0.210886	0.044659	0.044659	0.053094	0.053094	0.000000
5: omegagen 2	0.009750	0.009750	0.287054	0.287054	0.000000	0.044659	0.044659	0.053094	0.053094	0.210886
6: deltagen 3	0.000295	0.000295	0.003527	0.003527	0.203850	0.393858	0.393858	0.000395	0.000395	0.000000
7: omegagen 3	0.000295	0.000295	0.003527	0.003527	0.000000	0.393858	0.393858	0.000395	0.000395	0.203850
8: deltagen 4	0.000075	0.000075	0.000067	0.000067	0.205831	0.033132	0.033132	0.363811	0.363811	0.000000
9: omegagen 4	0.000075	0.000075	0.000067	0.000067	0.000000	0.033132	0.033132	0.363811	0.363811	0.205831

Figure 3.9: Table result with the participation factors.

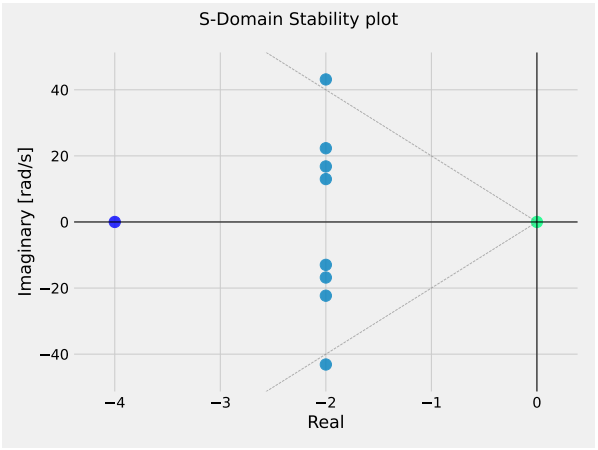

Figure 3.10: Complex domain plot.

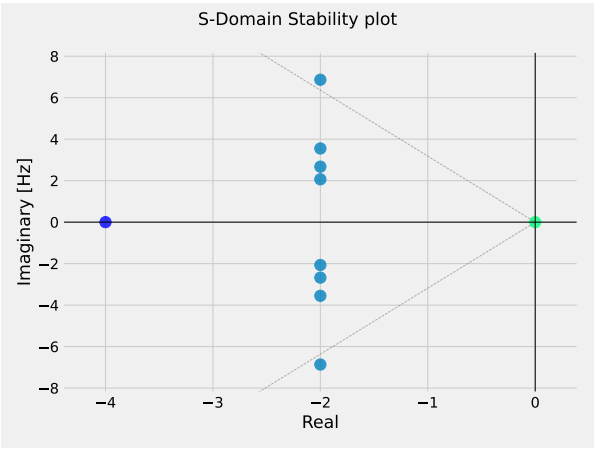

Figure 3.11: Complex domain plot in Hz.

In the eigenvalue plots, the user can take a look at the eigenvalues exact values by hovering

the mouse over the points. Moreover, the 5% damping ratio line is shown to help the user identify the unstable modes. It should also be noted that the color of the modes is determined by their real part. Therefore, the mode with the largest real part (the dominant mode) will be the darkest, and the mode with the smallest real part will be the lightest.

3.3 Benchmark and validation

The validation of the developed implementation is performed by comparing its results with those obtained using *Advanced Network Dynamics and Equilibrium Simulator* (ANDES). The comparison is carried out on the classical Kundur two-area test system, a standard benchmark for dynamic systems, enabling a direct verification the small-signal stability analysis between both tools.

3.3.1 ANDES

Thanks to its combination of symbolic precision and robust numerical performance, *Advanced Network Dynamics and Equilibrium Simulator* (ANDES) provides an excellent reference platform for the validation of dynamic and stability analysis tools in modern power system research. Developed as an open-source framework for modelling differential-algebraic systems, ANDES enables a fully symbolic formulation of network equations and control models, which ensures reproducibility and analytical transparency across a wide range of test systems. For these reasons, ANDES is widely recognised as a reliable benchmark for many types of simulations such as small-signal stability studies and has been used here as the reference software against which the performance of VeraGrid has been assessed.

Figure 3.12: ANDES logo. *Source*: ANDES documentation [15].

VeraGrid imports ANDES models directly from standardized `.json` files, allowing the seamless translation of system data, dynamic models, and numerical parameters. Once imported, VeraGrid performs the ANDES small-signal stability computations internally, based on ANDES numerical engine. The results obtained from VeraGrid have been compared with those from ANDES. VeraGrid successfully reproduces the full set of eigenvalues, damping ratios, and oscillation frequencies computed by ANDES, confirming the equivalence of its linearisation procedures and numerical solvers.

Beyond direct eigenvalue matching, the validation process also includes the comparison of mode shapes, damping factors, and participation factors matrices to ensure consistency in both

the algebraic formulation and numerical sensitivity of the two tools. These extended checks are essential to confirm that the dynamic responses, not only the eigenvalue, coincide under identical modelling assumptions. In practice, this guarantees that VeraGrid preserves the physical interpretation of oscillatory modes and stability margins originally provided by ANDES, while operating within its own software framework.

In the following subsection the Kundur two-area system is used as a benchmark test case to illustrate the accuracy of VeraGrid in reproducing ANDES results.

3.3.2 Test case: Kundur two-area system

The Kundur two-area system is a standard benchmark network widely used for small-signal and transient stability studies. It was introduced in the P. Kundur power system stability literature [9] as a compact, yet representative, test case that exposes inter-area oscillatory modes and control interactions without excessive model complexity.

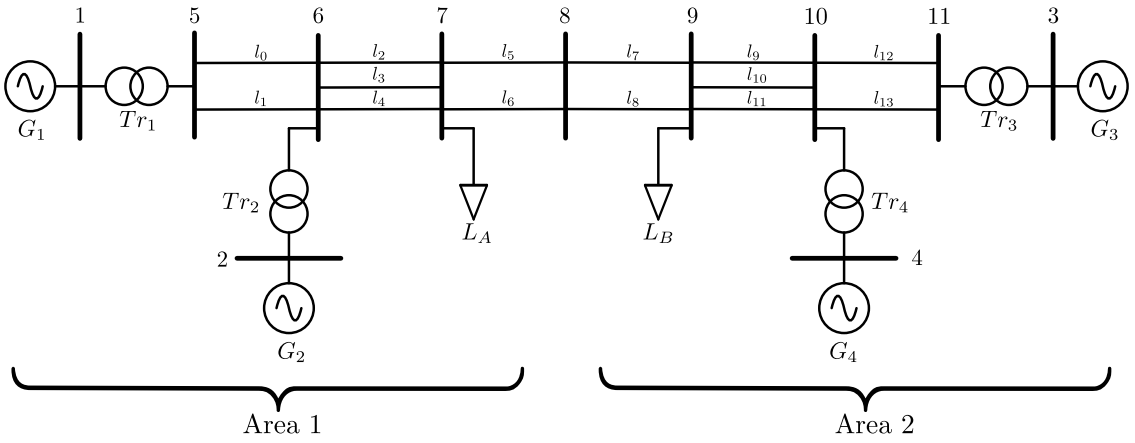

Figure 3.13: Kundur two-area system without shunt.

Figure 3.13 shows the one-line diagram of the Kundur two-area system without shunt. The main characteristics of the system are the following:

- There are two areas connected by a pair of parallel lines. In each area, 2 synchronous generators are placed so that each area can swing against each other and produce inter-area oscillations.
- All the synchronous generators are connected to the network through a transformer.
- In this version of the Kundur two-area system no shunts are connected to buses 7 and 9.
- While the system seems exactly symmetrical, the loads and generators are not exactly

equal. This provoques the need of power exchange between areas.

Table 3.1 summarizes the parameters of the system.

Table 3.1: Main parameters of the Kundur two-area system without shunts.

Element	Parameter	Symbol / Value	Units / Notes
Buses			
Bus1 to 4	Nominal voltage	20	kV
Bus5 to 11	Nominal voltage	230	kV
Generators			
G1, G2	Nominal Power	900	MVA
	Nominal voltage	20	kV
	Inertia constant	$M = 13$	s
	Damping coefficient	$D = 10$	–
	Impedances	$r_a = 0,0$, $x_d = 0,3$	p.u.
G3, G4	Nominal Power	900	MVA
	Nominal voltage	20	kV
	Inertia constant	$M = 12,35$	s
	Damping coefficient	$D = 10$	–
	Impedances	$r_a = 0,0$, $x_d = 0,3$	p.u.
Transformers			
Tr 1, 2, 3, 4	Impedance	$R = 0,0$, $X = 0,15$, $B = 0,0$	p.u.
	Rate	900	MVA
Transmission lines			
Line 0, 1, 12, 13	Impedance	$R = 0,005$, $X = 0,05$, $B = 0,02187$	p.u.
	Rate	750	MVA
Line 2, 3, 4, 9, 10, 11	Impedance	$R = 0,003$, $X = 0,03$, $B = 0,00583$	p.u.
	Rate	700	MVA
Line 5, 6, 7, 8 (tie-line)	Impedance	$R = 0,011$, $X = 0,11$, $B = 0,19250$	p.u.
	Rate	400	MVA
Loads			
Load A	Power	$P_L = 967$, $Q_L = 100$	MW, Mvar
Load B	Power	$P_L = 1767$, $Q_L = 100$	MW, Mvar

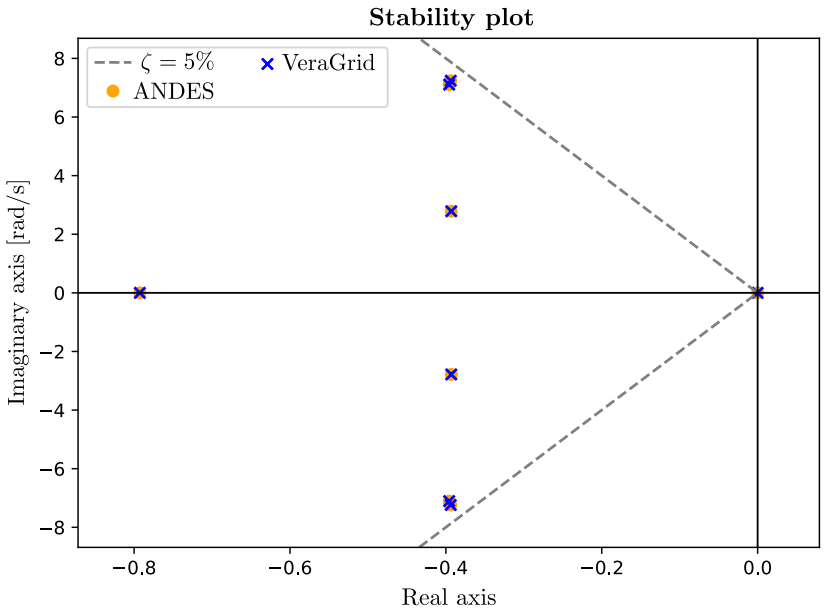

Figure 3.14: Andes vs VeraGrid complex domain plot comparison.

Figure 3.14 shows the comparison in the complex plane of the eigenvalues obtained with VeraGrid and Andes for the Kundur two-area system without shunts, while Tables 3.2 and 3.3 show the modes, damping ratios, oscillation frequencies and participation factors of the system. The results between Andes and VeraGrid are practically identical, with less than 0,0001% absolute error, this one due to numerical computation differences.

Table 3.2: Modes, damping ratios and oscillation frequencies of the Kundur Two-Area System.

Mode	Real part	Imaginary part	Damping ratio	Oscillation Frequency [Hz]
Mode 0	-0,3938	7,2377	0,0543	1,1519
Mode 1	-0,3938	-7,2377	–	–
Mode 2	-0,3958	7,1061	0,0556	1,1310
Mode 3	-0,3958	-7,1061	–	–
Mode 4	-0,3931	2,7838	0,1398	0,4431
Mode 5	-0,3931	-2,7838	–	–
Mode 6	0,0000	0,0000	–	–
Mode 7	-0,7926	0,0000	–	–

Table 3.3: Participation factors of the Kundur two-area system.

State/Mode	Mode 0	Mode 1	Mode 2	Mode 3	Mode 4	Mode 5	Mode 6	Mode 7
δ_1	0,123	0,123	0,102	0,102	0,174	0,174	0,197	0,006
ω_1	0,123	0,123	0,102	0,102	0,174	0,174	0,000	0,197
δ_2	0,151	0,151	0,122	0,122	0,117	0,117	0,214	0,006
ω_2	0,151	0,151	0,122	0,122	0,117	0,117	0,000	0,214
δ_3	0,103	0,103	0,157	0,157	0,102	0,102	0,283	0,006
ω_3	0,103	0,103	0,156	0,156	0,102	0,102	0,000	0,272
δ_4	0,123	0,123	0,119	0,119	0,108	0,108	0,306	0,006
ω_4	0,123	0,123	0,119	0,119	0,107	0,107	0,000	0,293

The small-signal stability analysis of the Kundur two-area system shows that the system is asymptotically stable, despite the presence of one zero eigenvalue associated with the rigid-body motion. Three oscillatory mode pairs are identified: two local modes around 1,14 Hz with damping ratios of about 5,5%, and one inter-area mode at 0,44 Hz with a damping ratio of 13,98%. In addition, two real modes appear, the dominant pole -0,7962 corresponding to a fast mechanical decay and another representing the neutral reference frame.

The local modes are associated with oscillations of the generators within the same area against one another. In this case, two pairs of complex conjugate eigenvalues located at $-0,3938 \pm j7,2377$ and $-0,3958 \pm j7,1061$ correspond to frequencies of $1,1519$ and $1,1310$ Hz respectively with damping ratios near $5,5\%$. These modes involve mainly the rotor angles and speeds of generators within each area, indicating local electromechanical interactions between closely coupled machines. The higher frequencies reflect the local nature of these oscillations, which are typically faster than inter-area modes due to the stronger electrical coupling and lower inertia involved. The relatively low damping (just above the threshold 5%) means that oscillations at these frequencies would decay slowly after a disturbance, and therefore these modes are usually targeted for improvement through power system stabilizers designed to increase the effective damping while maintaining the natural frequency range around one hertz.

The inter-area mode, represented by the eigenvalue pair $-0,3931 \pm j2,7838$, exhibits a frequency of $0,4431$ Hz and a damping ratio of $13,98\%$. The participation factors show that the mode involves the collective motion of generators in one area oscillating against those in the other, with all machines in each group moving coherently. This behavior reflects the exchange of power through the tie-line connecting the two subsystems, which gives rise to slow, large-scale oscillations of the inter-area type. The relatively higher damping compared with the local modes suggests that the tie-line contribute sufficient stabilizing torque under the present operating point. Nevertheless, variations in power transfer or network strength could reduce this damp-

ing and make the inter-area oscillation more critical, as observed in many large interconnected grids.

Among the real modes, one negative mode at approximately $-0,7926$ represents a non-oscillatory but fast-decaying mode, mainly related to the mechanical dynamics of the generating units. Its time constant ($\tau = -1/Re(\lambda)$), around $1,26$ s, indicates a rapid stabilizing effect that brings the system back to equilibrium after active-power mismatches or primary control actions. This mode is considered dominant in the sense that it governs the short-term return to balance but does not limit the overall stability margins of the system.

Finally, the eigenvalue located at the origin, $\lambda = 0$, corresponds to the uniform or rigid-body mode. Mathematically, it indicates marginal stability, as perturbations projected onto this mode neither grow nor decay. Physically, however, it simply reflects the absence of an absolute angular reference: if all generator rotor angles are shifted by the same constant, the electrical state of the system remains unchanged. Only differences in rotor angles carry physical meaning, so this mode has no effect on the stability of the relative motion among machines. In practical studies, this degree of freedom is not considered for the stability analysis. Consequently, although the rigid-body mode is formally marginally stable, it is not regarded as a sign of instability in the power system.

These results are consistent with the classical interpretation of the two-area system and confirm its suitability as a benchmark for dynamic studies.

4 EMT Dynamic framework

5 Environmental impact

An environmental contribution of this work lies in its support for energy efficiency and the ongoing energy transition. The developed framework contributes to the optimal operation and planning of power systems with high shares of renewable generation. Stability analysis allows network operators to safely integrate variable renewable sources such as wind and solar while maintaining system reliability. This, in turn, facilitates higher utilization of clean energy resources and reduces curtailment, leading to a more efficient use of installed capacity and a measurable reduction in carbon emissions. In a broader context, the development of robust and open simulation tools is essential to achieving global decarbonisation objectives and ensuring a stable, sustainable transition towards a low-carbon energy system [16].

Although the present work involves primarily computational and simulation tasks rather than physical infrastructure, it nonetheless carries meaningful environmental implications. Advanced RMS and EMT tools aid in the integration of renewable energy sources by improving the predictability and security of grid behavior, thereby helping reduce reliance on fossil fuel backup plants and associated greenhouse gas emissions[17]. The trade-off between simulation fidelity and computational cost is also well documented; for example, Yan et al. examine the impact of different numerical integration schemes on simulation accuracy and efficiency [18]. Minimizing computational energy consumption through algorithmic optimization is thus a relevant concern.

Moreover, the promotion of open-source platforms in power system research enhances resource efficiency by reducing duplicated development efforts and enabling community-driven innovation. The use of openly shared modeling libraries accelerates progress and lowers barriers to entry[19]. By providing validated simulation tools, this work contributes to more sustainable and cost-effective grid planning, potentially reducing the environmental footprint of energy infrastructure expansion.

On the other side, the actual footprint of the project development can be computed as kg of CO_2 equivalent emitted to the atmosphere. This study estimates the direct environmental footprint accounting for the electricity consumed by the computer and monitor, the corresponding share of HVAC energy use and the amortized embodied emissions of the electronic equipment. The calculation considers a five-month period (September to January), with a working schedule of eight hours per day, Monday to Friday. This corresponds to approximately $22 \text{ weeks} \cdot 5 \text{ days/week} \cdot 8 \text{ h/day} \approx 880$ hours.

The working environment corresponds to a small office space in the center of Barcelona, shared by approximately 20 occupants. In order to simplify the study, the following assumptions are done:

- Only the energy used by the laptop and the monitor are considered.
- The kg of CO_2 equivalent emitted by the manufacturing and transportation of the laptop and monitor are considered. The office equipment is not considered.
- Only the energy consumption due to HVAC in the office is considered due to its high contribution to energy consumption in the office.

For the Laptop embodied emissions Circular Computing [20] proposes 331 kg CO_2eq/device accounting for manufacturing, transport and 4 first years of use. According to their calculations, manufacturing and transport accounts by approximately 90% of those emissions. For the monitor, a standard monitor has been selected. HP sustainability report [21] states that 210 kg CO_2eq/device are emitted during manufacturing, transport, use and end of life. Manufacturing, transport and end of life account for a 47% of those emissions. According to l'Institut Català de l'Energia (ICAEN) [22], the recommended average HVAC energy consumption in an office building is 140 kWh/m^2 year. Scaling this consumption to an average of 5m^2 per person in a office and 260 working days a year the resulting energy consumption is 0,299 kWh/hour person. For the allocation factor of the laptop and monitor, 5 years of lifespan are assumed.

All parameters used are summarized in Table 5.1.

Table 5.1: Emission factors and embodied carbon data used in the estimation.

Source / Component	**Value**	**Unit**	**Reference**
Electricity emission factor (Spain, 2023)	0,26	kg CO_2eq/kWh	[23]
Laptop embodied emissions (manufacturing)	297,9	kg CO_2eq/device	[20]
Monitor embodied emissions (manufacturing)	94,47	kg CO_2eq/device	[21]
Average laptop power draw	45	W	[24]
Average monitor power draw	37	W	[25]
HVAC energy use per occupant (shared office)	0,299	kWh/h pers	[22]
Allocation factor laptop and monitor	0.084	–	assumed

The total electricity consumption during the five-month period can be expressed as:

$$E_{total} = E_{PC+monitor} + E_{HVAC}\ [kWh] \tag{5.1}$$

where:

$$E_{PC+monitor} = (P_L\ W + P_M\ W) \cdot \frac{880\ h}{1000\ W/kW} = 72,16 kWh \tag{5.2}$$

$$E_{HVAC} = 0,299\ kWh/h \cdot 880\ h = 263,25\ kWh \tag{5.3}$$

Corresponding to a total of 87,206 kg of CO_2eq due to energy consumption.

The allocation contribution of the LCA for the laptop and monitor correspond to a total of $(297,9 + 94,47)$ kg CO_2eq $\cdot 0,084 = 32,959$ kg CO_2eq.

The total emissions are 120,165 kg of CO_2eq. Table 5.2 summarizes the contribution of each element to the total

Table 5.2: Estimated carbon footprint of the thesis work including HVAC, lighting, and office equipment.

Component	**kg CO_2eq emissions**	**Contribution**
Laptop + monitor operation	$18,761$	$15,613\ \%$
HVAC energy consumption	$68,444$	$56,958\ \%$
Allocated embodied emissions (laptop + monitor)	$32,959$	$27,428\ \%$
Total estimated footprint	**120,165**	

Results conclude that the main contribution to the project development global emissions are due to the office HVAC being the 56,95% of the total equivalent CO_2 emissions. Regarding the laptop and monitor it is seen how the manufacturing and transporting contribution is much higher than the operation itself, highlighting the importance of Life Cycle Assessments on the environmental assessments.

To contextualize the estimated footprint of this work, it is useful to compare it with reference studies on greenhouse gas emissions in office environments. According to the European Environment Agency, the annual emissions associated with a standard office workplace in Europe (including HVAC, lighting, equipment, and commuting) are between 1500 and 2000 kg CO_2eq per employee [26]. The corresponding contribution of 5 months is between 625 and 833 kg CO_2eq. As noted, in this study the EEA considers also the commuting to work. Circular Ecology [27] states that, in Europe. only 53% of the emissions of going to work in an office account for the office itself meaning 47% of the emissions correspond to commuting. Applying this percentage the average office emissions(including HVAC, lighting, equipment) in Europe are between 331,250 and 441,677 kg CO_2eq.

In comparison, the total estimated footprint of approximately 120 kg CO_2eq for this thesis project is lower than the studies stated in the previous paragraph. It is to be considered that the EEA study considers both lightning and equipment while this study does not. Moreover, the average energy consumption in Europe is higher than in Spain so this result may be affected. Therefore, the result is considered correct and highlights the importance of energy-efficient habits.

6 Gender and social impact

TO CHECK!!

Beyond technical contributions, this project supports broader social goals through the promotion of inclusive and accessible tools. Open-source frameworks for power system analysis democratize advanced engineering capabilities, enabling participation from researchers and engineers in diverse regions and institutions. For instance, the development of open and transparent electricity network models demonstrates how access to modeling tools can empower under-resourced communities [28]. Open-source frameworks support the idea that knowledgment must be a social right and accessible for everybody.

From a gender perspective, power systems engineering remains dominated by men in many contexts. Disseminating open methodologies and encouraging diverse collaboration may help widen participation of underrepresented groups in STEM fields.

Finally, improving the reliability, stability, and efficiency of power systems has direct social benefits: fewer outages, better access to electricity, and more resilient grids that support sustainable development goals.

7 Budget

TO CHECK!!!

The complete costs are detailed in Table 7.1.

- Les hores emprades per la realització del treball (les corresponents als crèdits del TFE). Millor si estan separades per les tasques executades segons la planificació. En una primera aproximació podeu considerar que una de les vostres hores es paga a 15 €/h. - Les despeses operatives: electricitat, calefacció, aigua o telefonia. Sumeu la part proporcional dels termes fixos. També tingueu en compte els viatges, les despeses d'oficina i d'altres. - Les despeses experimentals: materials, aigua, electricitat, combustibles, llicències, llibres. - Si és el cas, amortització dels equips. Considereu que un PC s'amortitza linealment en 5 anys i un telèfon mòbil en 3 anys. - Serveis abonats per obtenir algunes dades (per exemple d'un servei de microscòpia electrònica, encara que no ho hagi pagat l'estudiant). - I totes aquelles que haurien de constar si el treball fos realista i s'hagués de facturar.

Table 7.1: Total Costs.

Concept	**Unit cost (€/h)**	**Quantity (h)**	**Total (€)**
Development engineer	25.00	1,040	26,000.00
Supervisor	30.00	260	7,800.00
Total			33,800.00

8 Time Planning

Figure 8.1 shows the temporal evolution of the various tasks that have constituted the project.

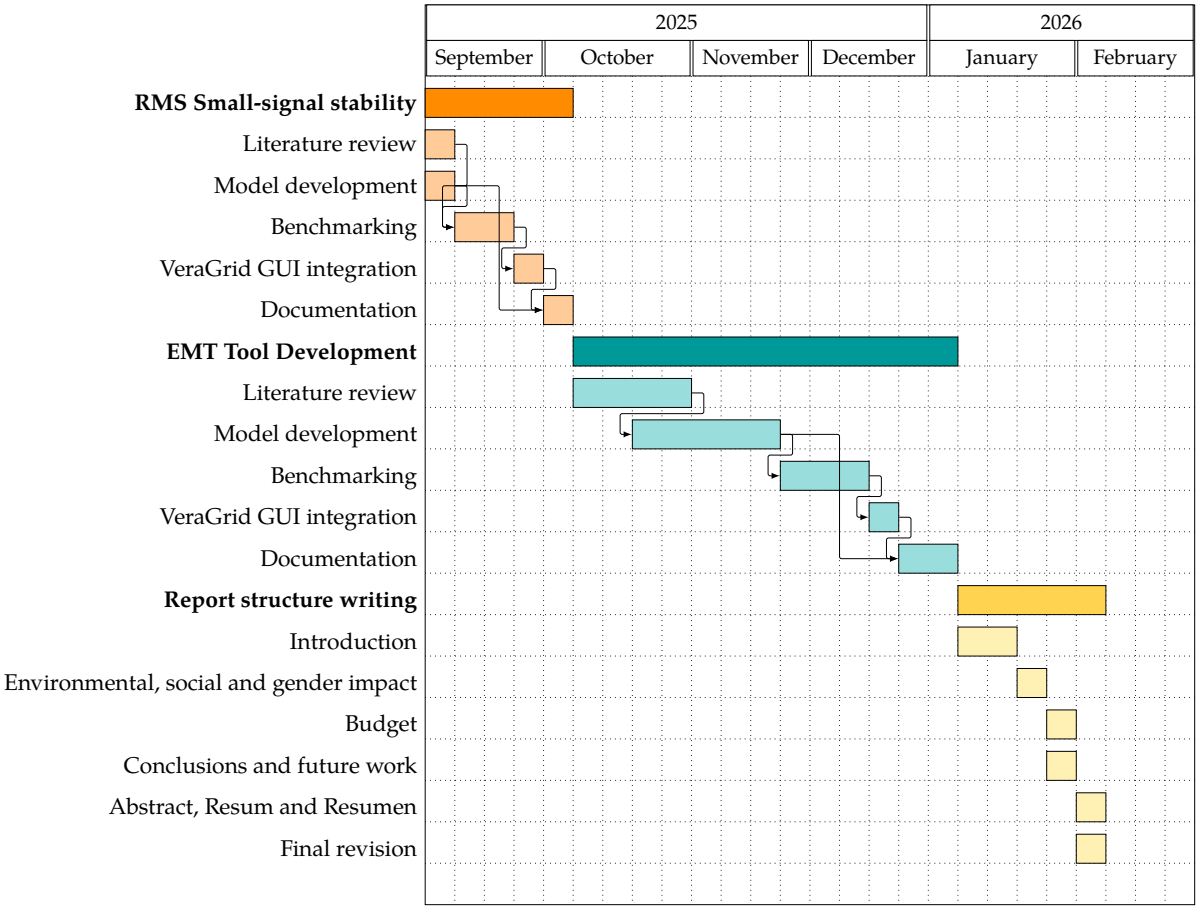

Figure 8.1: Gantt Chart of the project.

9 Conclusion

9.1 Further Work

- -

Acknowledgments

References

[1] eRoots Analytics S.L., *Veragrid: Open-source code to power model handling, powerful simulations, and seamless visualization.* Version v1.0, Combines AC/DC power flow, optimisation, and time-series simulation in an open and extensible architecture., Barcelona, Spain. [Online]. Available: https://www.eroots.io/veragrid

[2] J. D. Lara, R. Henriquez-Auba, D. Ramasubramanian, S. Dhople, D. S. Callaway, and S. Sanders, "Revisiting power systems time-domain simulation methods and models," *IEEE Transactions on Power Systems*, vol. 39, no. 2, pp. 2421–2437, 2024. DOI: 10.1109/TPWRS.2023.3303291

[3] C. L. Fortescue, "Method of symmetrical co-ordinates applied to the solution of polyphase networks," *Transactions of the American Institute of Electrical Engineers*, vol. XXXVII, no. 2, pp. 1027–1140, 1918.

[4] F. Casella, *Properties of differential-algebraic equation systems (dae)*, Lecture slides, Dipartimento di Elettronica e Informazione, Politecnico di Milano, Available at https://staff.polito.it/roberto.zanino/sub1/teach_files/modelica_minicourse/02%20-%20DAE%20Systems.pdf., 2010.

[5] P. W. Sauer and M. A. Pai, *Power System Dynamics and Stability*. Prentice Hall, 1997.

[6] H. Cui, F. Li, and K. Tomsovic, "Hybrid symbolic-numeric framework for power system modeling and analysis," *IEEE Transactions on Power Systems*, vol. 36, no. 2, pp. 1373–1384, 2021. DOI: 10.1109/TPWRS.2020.3017019

[7] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations*. Chichester, UK: John Wiley & Sons, 2008.

[8] A. C. Hindmarsh, "Algebraic equations, part i: Euler methods and error behaviour," *Computing in Science & Engineering*, vol. 9, no. 1, pp. 34–44, 1995.

[9] P. S. Kundur and O. P. Malik, *Power System Stability and Control*, English, 2nd ed. New York: McGraw Hill, 2022, ISBN: 9781260473544.

[10] P. Kundur et al., "Definition and classification of power system stability," *IEEE Transactions on Power Systems*, vol. 19, no. 2, pp. 1387–1401, May 2004. DOI: 10.1109/TPWRS.2004.825981

[11] N. Hatziargyriou et al., "Definition and classification of power system stability – revisited & extended," *IEEE Transactions on Power Systems*, vol. 36, no. 4, pp. 3271–3281, 2021. DOI: 10.1109/TPWRS.2020.3041774

[12] M. Cheah-Mane, A. Egea-Alvarez, E. Prieto-Araujo, H. Mehrjerdi, O. Gomis-Bellmunt, and L. Xu, "Modeling and analysis approaches for small-signal stability assessment of power-electronic-dominated systems," English, *Wiley Interdisciplinary Reviews: Energy and Environment*, vol. 12, no. 1, 2023, ISSN: 2041-8396. DOI: 10.1002/wene.453

[13] V. Konoval and R. Prytula, "Participation factor in modal analysis of power systems stability," *Poznan University of Technology Academic Journals. Electrical Engineering*, 2016.

[14] Qt Group, *Qt designer: Cross-platform gui design tool*, Part of the Qt framework for developing graphical user interfaces and cross-platform applications., Espoo, Finland. [Online]. Available: https://www.qt.io/product/development-tools

[15] H. Z. et al., *Andes: Advanced network dynamics and equilibrium simulator*, version v1.8.0, Available at https://docs.andes.app. [Online]. Available: https://docs.andes.app/en/stable/getting_started/index.html

[16] International Energy Agency, "World energy outlook 2024: The energy transition," International Energy Agency (IEA), 2024. [Online]. Available: https://www.iea.org/reports/world-energy-outlook-2024

[17] M. Xiong et al., "An open-source parallel emt simulation framework," *Elsevier / Electric Power Systems Research*, 2024.

[18] X. Yan et al., "Computational efficiency-accuracy trade-offs in emt simulation," *Energies*, 2025. DOI: 10.3390/en18095173

[19] L. Tesfatsion, *Open source software for electric power industry*, Iowa State University site, n.d.

[20] Circular Computing, *The carbon footprint of a laptop: Understanding the true impact*, 2022. [Online]. Available: https://circularcomputing.com/news/carbon-footprint-laptop/

[21] HP Inc., "Hp monitor life cycle assessment summary report," HP Sustainability Report, 2021. [Online]. Available: https://h20195.www2.hp.com/v2/GetDocument.aspx?docname=c08699464

[22] Institut Català d'Energia (ICAEN), *Guia per a la millora de l'eficiència energètica en les instal·lacions industrials*, Online resource, 2023. [Online]. Available: https://icaen.gencat.cat/web/.content/migracio_automatica/documents/sala_de_premsa/arxius/guia.pdf

[23] Generalitat de Catalunya, Departament d'Acció Climàtica, Alimentació i Agenda Rural, *Emission factors associated to the energy mix*, Online resource, 2023. [Online]. Available: https://canviclimatic.gencat.cat/en/actua/factors_demissio_associats_a_lenergia/index.html

[24] ASUS, *Technical specifications asus vivabook 15*, 2025.

[25] Display Specifications, *Power consumption specifications aoc w-led*, 2025.

[26] European Environment Agency, "Occupant-related energy use and emissions in office buildings in europe," EEA Technical Report No 07/2021, 2021. [Online]. Available: https://www.eea.europa.eu/publications

[27] Circular Ecology, *The carbon emissions of homeworking and office working*, 2023. [Online]. Available: https://circularecology.com/news/the-carbon-emissions-of-homeworking-and-office-working

[28] D. Kirli et al., "Pypsa meets africa: Developing an open source electricity network model of the african continent," in *arXiv preprint arXiv:2110.10628*, 2021.