NESO
National Energy
System Operator

Delivered in partnership with

tnei

WP1 Report

SSO Identification Tool

Document Reference: 15167-001-D1

NESO
National Energy
System Operator

Quality Assurance

TNEI Services Ltd and TNEI Africa (Pty) Ltd. ("TNEI") operates an Integrated Management System and is registered with Ocean Certification Limited as being compliant with ISO 9001(Quality), ISO 14001 (Environmental) and ISO 45001 (Health and Safety).

Disclaimer

National Grid Electricity System Operator ("NGESO") has endeavoured to prepare the published report ("Report") in respect of Probabilistic Stability ("Project") in a manner which is, as far as possible, objective, using information collected and compiled by NGESO and its Project partners ("Publishers"). Any intellectual property rights developed in the course of the Project and used in the Report shall be owned by the Publishers (as agreed between NGESO and the Project partners). The Report provided is for information only and viewers of the Report should not place any reliance on any of the contents of this Report including (without limitation) any data, recommendations or conclusions and should take all appropriate steps to verify this information before acting upon it and rely on their own information. None of the Publishers nor its affiliated companies make any representations nor give any warranties or undertakings in relation to the content of the Report in relation to the quality, accuracy, completeness or fitness for purpose of such content. To the fullest extent permitted by law, the Publishers shall not be liable howsoever arising (including negligence) in respect of or in relation to any reliance on information contained in the "Report".

Document Control

Revi	Status	Prepared by	Checked	Approved	Date
D0	DRAFT	Diptargha Chakravorty			21/12/2022
D1	FIRST ISSUE	Diptargha Chakravorty			10/02/2023

TNEI Services Ltd
Company Registration Number: 03891836 VAT Registration Number: 239 0146 20

Registered Address
Bainbridge House
86-90 London Road
Manchester
M1 2PW
Tel: +44 (0)161 233 4800

7th Floor West One
Forth Banks
Newcastle upon Tyne
NE1 3PA
Tel: +44 (0)191 211 1400

7th Floor
80 St. Vincent Street
Glasgow
G2 5UB
Tel: +44 (0)141 428 3180

TNEI Ireland Ltd
Registered Address: 104 Lower Baggot Street, Dublin 2, DO2 Y940
Company Registration Number: 03891836 VAT Registration Number: 239 0146 20

For enquires and general
Correspondence - use
Manchester
office details
Tel: +353 (0)190 36445

TNEI Africa (Pty) Ltd
Registered: Mazars House, Rialto Rd, Grand Moorings Precinct,7441 Century City, South Africa

Company Number: 2016/088929/07

1st Floor Willowbridge Centre
39 Carl Cronje Drive
Cape Town
South Africa, 7530
Tel: +27 (0)21 974 6181

Contents

APPENDICES

TABLES

FIGURES

1 Introduction

The UK has been at the forefront of connecting large offshore windfarms to the transmission system. Over the past few years, the size of the windfarms has increased significantly from a few hundred megawatts to more than 2 gigawatts like Hornsea, off the east coast of England. Even larger projects are in the pipeline such as Dogger Bank A and B. As the size of individual turbines have increased in the last decade, it is now fairly common to have offshore windfarms of more than 1 gigawatts capacity.

However, this transition from synchronous generation (SG) dominated conventional grid, to a composite grid where inverter-based resources (IBRs) might have a higher proportion in the future, has led to some new phenomena in the system. These are collectively termed as sub-synchronous oscillations (SSO) and an in-depth discussion on this topic is presented in Section 2.

The fundamental reason for SSO in any system is the existence of poorly damped oscillatory modes which are induced due to interaction between different technology types. The challenges in addressing these interactions are –

- The interactions are complex and difficult to identify beforehand without exhaustive studies.
- It is difficult to identify the components or assets participating in any interaction.
- New developments in the network such as connection of new sites or reinforcements can introduce poorly damped modes in a system.

A state-space approach is an effective method for identifying the oscillatory modes in a power system and determining the underlying cause of these modes [1]. This can be done by analyzing the states with high participation in the oscillatory modes and controlling the corresponding power system component to effectively dampen the modes. The analysis of system instability, design of damping controllers, and similar tasks can be performed using eigen decomposition, participation factor analysis, eigenvalue sensitivity analysis, pole-placement, and other linear stability analysis tools. However, a complete understanding of the system is necessary for state-space modeling. This was feasible in traditional systems with synchronous generators and associated controllers such as exciters and power system stabilizers. In current systems, the presence of converter-interfaced sources means that the control algorithms are vendor-specific and considered confidential commercial assets, rarely disclosed as white-box models. Instead, manufacturers provide black-box models with input-output relationships without internal details, which are shared with others as binary files (e.g. .lib and .dll) that can be used with a numerical solver to perform time-domain simulations [2]. These simulations, particularly for large practical power systems, can become very complex and exhaustive, such as in the case of electromagnetic transient studies, and may not provide a clear understanding of the root cause.

The premise of this project is to develop a framework for system operators to scan through a large number of scenarios in acceptable time to identify potential SSO phenomena. The framework will utilize a combination of automation techniques, frequency domain analysis methods (such as frequency scan for non-linear elements, passivity tests, grey box approach) and machine learning to automate the end-to-end process with minimal supervision of engineers.

1.1 Hyperlinks to navigate through the report

This report covers a wide range of topics around SSO analysis. For ease of navigation, selected few topics are listed below along with hyperlinks to the corresponding sections.

Topic	**Section**
Reported events of subsynchronous oscillations	Section 2.2
Common frequency domain analysis techniques	Section 3.1
Common time domain analysis techniques	Section 3.2
Frequency scan method for non-linear components	Section 4.2
Grey box approach	Section 5.3
Implemented automated identification technique	Section 7.3
Validation of the implemented approach	Section 7.4
Pipeline of implemented classifiers	Section 8.1
Results from initial studies	Section 8.5

2 Subsynchronous oscillations

The interaction of the electrical network with energy producing sources in a frequency range lower than the fundamental frequency has been observed for a long time and the first reported event was in 1970 in Mohave, US. The transmission connected sources were predominantly synchronous machines with different technology types such as coal fired, gas fired, nuclear and hydro. These interactions were termed Subsynchronous Resonance (SSR) as the phenomenon originated due to a resonance in the electrical network at a frequency close to the mechanical frequency of oscillation of the turbines.

Since the first reported event, the network has evolved significantly and so as the interaction phenomena. Share of traditional generation technology has steadily declined and volume of Inverter Based Resources (IBR) have increased. It has been observed that the new technologies (both control and converter design) could introduce Subsynchronous Oscillations (SSO) in the network (under certain condition) which do not fall under the traditional definition of SSR.

The following Section 2.1 discusses the current accepted classification of SSO and the corresponding technology types involved in the interaction.

2.1 Subsynchronous oscillation classification

Based on the phenomenon reported so far in the literature, SSO can be divided into three categories, as shown in Figure 1. The origin of these interactions are due to different network components and the oscillations are triggered under certain network configuration and operating conditions. More details on each category is provided in the following subsections.

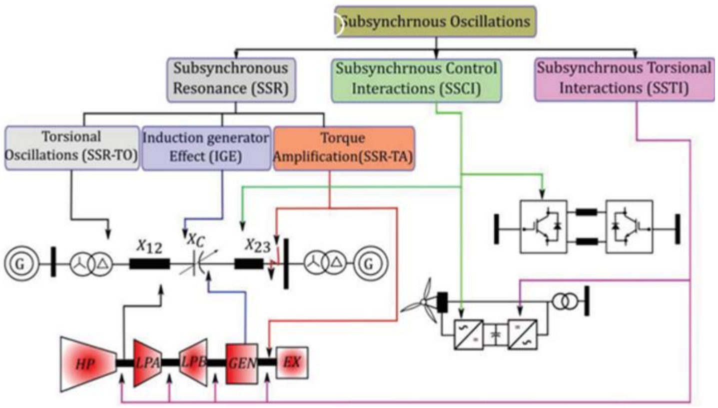


Figure 1 Classification of Subsynchronous oscillation phenomena

2.1.1 Subsynchronous Resonance (SSR)

This phenomenon results in poorly damped mechanical oscillations on a turbine generator shaft due to exchange of energy with the electrical network. This is usually triggered after a large network disturbance such as a fault when the generator is connected radially with a series compensated line. The generator shaft is comprised of a multi-mass system, each mass with its own natural frequency of oscillation, also referred to as its modal frequency.

A mechanical oscillation of f_m on the turbine side will translate into $f_0 - f_m$ and $f_0 + f_m$ oscillations on the electrical network side, where f_0 denotes the nominal electrical frequency. The modal frequencies of the masses in a turbine-generator shaft system fall within a typical range such that a subsynchronous series resonant point in the electrical system can excite electromechanical oscillations between the power system and the turbine when the electrical resonant frequency excites a modal frequency of the turbine. This condition can occur when $f_0 - f_{mr} \approx f_{er}$, where f_{mr} and f_{er} denote the shaft modal and network electrical resonant frequencies, respectively. When $f_0 - f_{mr} = f_{er}$ the machine sees the system as purely resistive. Otherwise, it sees a capacitive nature when $f_0 - f_{mr} < f_{er}$ and an inductive nature when $f_0 - f_{mr} > f_{er}$.

The SSR phenomenon can be further subcategorised into three types of oscillations depending on the network condition initiating the event.

2.1.1.1 Torsional Oscillations (SSR-TO)

SSR-TO (also called Torsional Interaction in literatures and books such as [3]) is considered to be more dangerous and complex than Induction Generator Effect. Both phenomena are a type of small signal stability problem and can be triggered in the system without any large disturbances.

In SSR-TO the generator rotor oscillates sinusoidally about a constant speed of ω_0 (synchronous speed). To develop the equations for the induced voltages and the corresponding torques, it is assumed that the amplitude of rotor oscillation (A) is very small and the induced voltage in the stator consists of three sinusoidal components, a fundamental frequency (f_0) one, two other components of frequencies $f_0 \pm f_m$(Figure 2). These induced voltages lead to the flow of subsynchronous and supersynchronous currents in the machine. The currents produce corresponding flux and results in electrical torques, which can be represented by the following equations.

$$T_e^{sub} = -\frac{AE^2}{2\omega_m Z_{sub}}(\omega_0 - \omega_m)\sin\left(\omega_m t + \varphi_{sub}\right) \tag{1}$$

$$T_e^{sup} = \frac{AE^2}{2\omega_m Z_{sup}}(\omega_0 + \omega_m)\sin\left(\omega_m t - \varphi_{sup}\right) \tag{2}$$

In (1) and (2), ω_m stands for the frequency of oscillation of the generator rotor, $Z_{sub}\angle\varphi_{sub}$ & $Z_{sup}\angle\varphi_{sup}$ are the network impedances (magnitude and angle) at subsynchronous and supersynchronous frequencies, viewed from the generator internal bus, and E is the peak value of the machine internal emf. The detailed maths to calculate these equations can be found in [3].

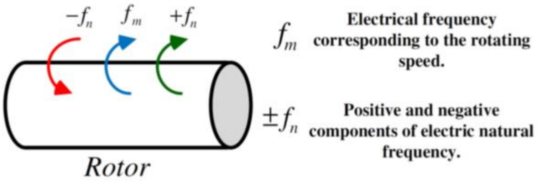


Figure 2 Synchronous and other frequency components in the rotor shaft

The presence of subsynchronous torques in the generator rotor could be dangerous, as the turbine-generator system has its own natural modes of oscillation that are typically below the synchronous frequency. If the induced subsynchronous torques coincide with one of the shafts natural modes of oscillation, the shaft begins to oscillate at this natural frequency with high amplitude. This phenomenon is SSR-TO (or SSR-TI) and can cause shaft fatigue, damage and even failure. The frequencies of oscillation that are of most interest lie between 4 and 47 Hz.

Figure 3 shows the phasor diagram giving the position of the torque components in relation to the rotor velocity. It is interesting to note that the supersynchronous frequency currents in the network give rise to a small magnitude of positive damping torque while the subsynchronous component causes negative damping. The smaller the oscillation frequency, higher is the negative damping. Thus the first torsional mode (smallest natural frequency of the turbine) can cause the most severe problem if the network impedance is minimum at that mode.

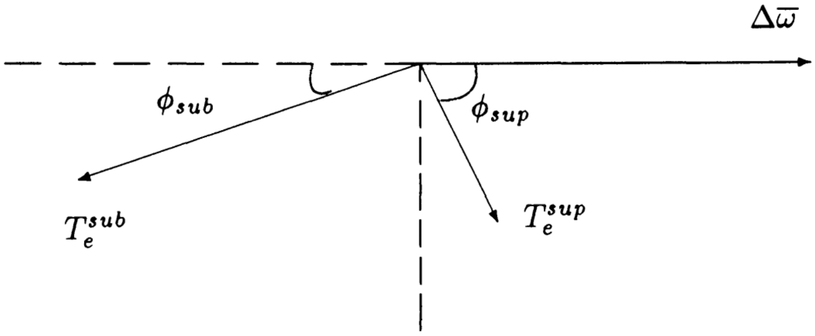


Figure 3 Phasor diagram of subsynchronous and supersynchronous electrical torques in a machine

2.1.1.2 Induction Generator Effect (IGE)

This phenomenon occurs due to self-excitation of the generator when currents at subsynchronous frequencies appear in generator windings and a negative resistance (i.e., negative damping) is encountered at these frequencies due to the induction generator effect. As the rotating Magneto Motive Force (MMF) produced by the subsynchronous frequency armature currents moves slower than the speed of the rotor, the resistance of the rotor (at subsynchronous frequency as viewed from the armature terminals) is negative as the slip of an induction generator is negative (see Figure 4 for the equivalent circuit of an induction generator). In contrast, the network presents a positive resistance to these currents. However, if the negative resistance of the generator is greater in magnitude than the positive resistance of the network, at a resonant frequency, there is a sustained subsynchronous current (self-excitation). This condition is the IGE [3], [4].

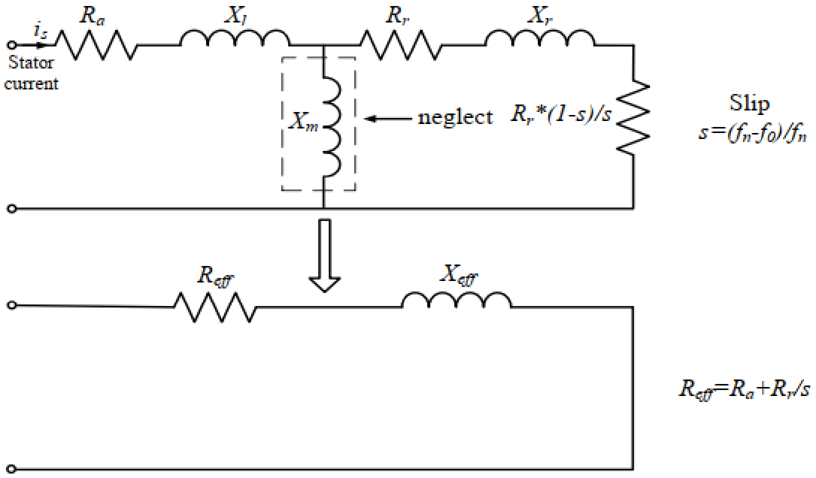


Figure 4 Equivalent circuit diagram of an induction generator

Although this phenomenon is named as the induction generator effect, it can occur in both synchronous and induction machines.

2.1.1.3 Torque Amplification (SSR-TA)

Transient torques can result from large system disturbances that cause sudden changes in current that oscillate at system natural frequencies. In transmission systems without series capacitors, system disturbances produce DC transients that eventually decay with time constants that are dictated by the ratio of inductance to resistance. However, for systems with series capacitors these transient currents are sinusoidal and will contain one or more oscillatory frequencies that are dependent on network capacitance and inductance, given by the following equation

$$i(t) = Asin(\omega_0 t + \varphi_0) + Be^{-\epsilon\omega_1 t}\sin(\omega_1 t + \varphi_1) \qquad (3)$$

where, $i(t)$ is the current, t is time, A is the amplitude of the fundamental frequency component, ω_0 is the fundamental frequency, φ_0 is phase of the fundamental frequency component, B is the amplitude of the oscillatory mode component, ϵ is the damping ratio of the oscillatory mode component, ω_1 is frequency of the oscillatory mode component, φ_1 is the phase of the oscillatory mode component.

A network with multiple series capacitor installations can have multiple subsynchronous frequencies, as opposed to the case shown above where only ω_1 is present. If the complement of any of these subsynchronous network frequencies coincide with one of the natural modes of a turbine-generator shaft, then large torque peaks will be induced. Transient electrical torque can have many components, such as unidirectional, exponentially decaying, and oscillatory torques with frequencies ranging from subsynchronous to multiples of the synchronous frequency. This form of SSR can induce large amplitude torques at subsynchronous frequency immediately following the disturbance [5]. Each occurrence of these high amplitude transient torques can result in loss of shaft life or even shaft failure.

2.1.2 SSTI

The phenomenon can be triggered due to an interaction between a multi-mass generator shaft and line commutated converter (LCC) based high voltage direct current (HVDC) controls. The source of the sub-synchronous oscillations are control interactions with HVDC converters, without any participation of network modes. This is the fundamental difference compared to SSR, and SSCI, where a network series resonance participates in the event. SSTI can also occur between a generator shaft and a static var compensator (SVC), and between a wind turbine generator (WTG) shaft and an associated power electronic converter. A power system stabiliser (PSS) can also participate in this event [6].

2.1.3 SSCI

This is purely an electrical phenomenon as opposed to classical SSR which occurs only at frequencies corresponding to the mechanical modes of the turbine generator shaft. SSCI events reported so far do not depend on the mechanical modes. Although this can occur with any type of converter interfaced generation, current research and publications have mainly observed this phenomenon with Type 3 wind farms, which make use of doubly fed induction generators (DFIGs). The DFIG controls act to amplify subsynchronous currents (i.e., provide negative damping at these frequencies) that might appear in the network after a large disturbance due to exchange of energy between a series compensated transmission line and the controller modes.

SSCI is an interaction between a converter-based generation and the network to which it is connected. The network has to exhibit a series resonance point or a reactance dip with specific characteristics at a subsynchronous frequency. A series resonance occurs when the effective series inductive reactance equals the series capacitive reactance of the network, thus resulting in a purely resistive circuit at the resonant frequency. This frequency characteristic is typical of a series compensated line. A network disturbance in the presence of a series resonance point can excite subsynchronous oscillations from the network side. It has been shown that by increasing the degree of series compensation on a line, the series resonance point is moved closer to the system's nominal frequency.

In other parts of the world, for example in China and the USA, SSCI has been observed in Type 3 wind farms that are connected to series compensated transmission lines. In 2009, an unplanned line outage in south Texas resulted in the direct interconnection of two Type 3 wind farms via a 50% series compensated 345 kV transmission line. On clearing the fault, a 20 Hz growing oscillation was measured in the system voltage and current which eventually forced the wind farms to enter a crowbar state [7]. Damage to the wind farm and network components were reported.

SSCI studies in literature are usually carried out with a lumped representation of the DFIG and its controller model with an equivalent impedance for the whole wind farm collector network in series with the step-up transformer [1]. Theoretical studies have shown that in case of SSCI the negative damping or resistance is mainly contributed by the rotor-side converter (RSC) of the DFIG. If the negative resistance generated by a wind farm overcomes the positive resistance of the grid, then unstable subsynchronous controller interactions will be observed, and the frequency of the oscillation will be dictated by the total inductance and capacitance of the network. The impedance model of a wind farm containing n identical DFIGs can be expressed by the following equation [8] –

$$Z_{DFIG} = \frac{R_{DFIG} + jX_{DFIG}}{n} = \{\left[\frac{K_p + R_r}{s} + jw_{SSR}L_r\right] // (jw_{SSR}L_m) + R_s + \, jw_{SSR}L_s\}/n \qquad (4)$$

where R_{DFIG} is the equivalent resistance and X_{DFIG} is the equivalent reactance of the whole wind farm, K_p is the proportional gain of the RSC control, R_r and L_r are the rotor side resistance and inductance and R_s and L_s are the stator side resistance and inductance, respectively.

As the slip s is generally negative at the SSCI frequency, the rotor side resistance will be a negative quantity and it will be significantly amplified by the proportional gain of the RSC control, thus contributing to the unstable oscillations.

Research and publications point out that, in series compensated systems, DFIGs are more vulnerable to SSCI as opposed to fixed speed WTGs and WTGs connected by back-to-back converters (Type 4 wind farms). The probability of SSCI is lower with fully rated converters such as those used in Type 4 wind farms, or with converters that couple a BESS with the grid, but it can still occur under special operating conditions. Recent research shows that for a standard Type 4 turbine control system, the closed-loop bandwidth of the reactive power controller and the phase locked loop (PLL) can have significant impact on the stability of the system [9]. An increase in the active power output of the turbine further compounds this problem.

2.2 Reported events

Several events have been reported in literature since the first recorded case of SSR at the Mohave generating station in 1970. We have tried to consolidate the reported events in the following table along with the references which mentions it. Please refer to the respective papers to learn more about the events.

A couple of points to note here –

- Some of the events have recorded more than one oscillation frequency and more than one phenomenon as the cause of the oscillation.
- From 2009 onwards, SSCI has been the predominant cause for all reported events and there is a clear increase in the frequency of these events as well.

Table 2-1 List of reported SSO events

Year	Location	Cause	Oscillation Frequency (Hz)	Reference
1970	Mohave, US	SSR	26.7, 30.1	[10]
1977	Square Butte, US	SSTI	11.5 (Simulation)	[11]
1990	Chester, US	SSTI	~7 (Simulation)	[12]
2007	Minnesota, US	SSR	9.44	[13]
2009	ERCOT, US	SSCI	20	[11], [14]
2011	Shangdu, China	SSR	26 (Simulation)	[15]
2012	Guyuan, China	SSR, SSCI	6 - 9	[8], [16]
2013	Guyuan, China	SSR, SSCI	8.1	[8]
2015	Hami, Xinjiang, China	SSCI	27 - 33	[17], [18]
2017	ERCOT, US	SSR, SSCI	20 - 30	[19]
2019	GB grid, UK	SSCI	8-9 (TBC)	[20]

3 Analysis techniques

Several analysis techniques are available for dynamic systems and the use of a particular technique depends on the purpose of the study. In some cases, a single technique may not be sufficient to capture the overall characteristic and more than one approach may be necessary.

In this chapter we discuss the different methods applied to solve power system problems broadly divided into frequency domain and time domain methods.

3.1 Frequency domains methods

As the name suggests, the methods in this category use frequency as an independent variable. No concept of time is considered here, and the dynamics of the system is captured as a function of the frequency. Frequency domain methods are suitable for small signal stability analysis i.e., to study the behaviour of a system surrounding its nominal operating point.

3.1.1 Transfer function based

The transfer function (TF) of a linear system is defined as the ratio of the Laplace Transform of the output variable to the Laplace Transform of the input variable with all initial conditions assumed to be zero. A TF is therefore an input-output description of the behaviour of a system, and it does not include any information concerning the internal structure of the system.

TF is extensively used in control theory and is used in power system for developing controllers like AVR, governor, inverter control etc. It provides a convenient way to represent the input-output relationship of different parts of the system such as the plant model, the associated controllers, feedback systems etc. Routh- Hurwitz stability criterion can be used to analyse the stability of a closed loop system as it provides a necessary and sufficient condition for stability identification. However, the limitation of this method or TF analysis in general is to identify the root cause of a problem.

For SSO analysis, TF can be used to represent linear systems and for non-linear systems (such as converters) the equations would need to be linearised, but it would not provide any information about the source of any interaction.

3.1.2 Passivity based

The Passivity Based Control (PBC) theory originates from the physical law of energy conservation[21]. A system is said to be passive if there exists a nonnegative storage function $H(x)$ and a nonnegative dissipation function $d(t)$ [22] such that the energy supplied to the system (the left-hand term in equation (5)) is equal to the sum of energy stored and energy dissipated. This can be given by the following equation –

$$\int_0^t u^T(\tau)y(\tau)d\tau = H\big(x(t)\big) - H\big(x(0)\big) + d(t) \tag{5}$$

Where x is the state variable of the system, and u and y are the input and output vectors with the same dimension.

A passive system is intrinsically stable because of the dissipation function. The dissipation of energy drives the state of the system to an equilibrium point. An important property of the passivity concept is that any number of passive systems combined through parallel or feedback connections will result in an overall passive system. The interconnection of the passive sub-systems has no bearing on the overall system response, so the passivity property is independent of the network topology.

The PCB concept is explained below with the help of a simple DC circuit. A circuit like the one in Figure 5 can be considered passive only if the overall admittance or impedance has a positive real part at all frequencies. This definition is valid for both linear and non-linear circuits as long as these can be represented linearly around an operating point.

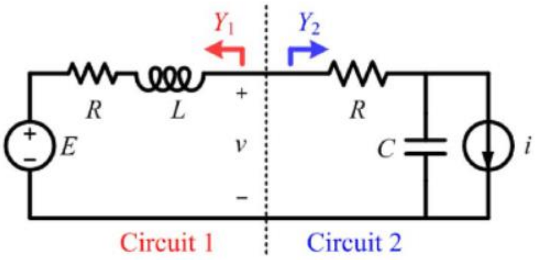


Figure 5 Simple RLC circuit

The above circuit is a typical second order system and the transfer function from the current i to the voltage v can be given by

$$\frac{v(s)}{i(s)} = \frac{R + Ls}{LCs^2 + 2RCs + 1} \tag{6}$$

The damping ratio can be calculated from the transfer function as $R\sqrt{\frac{C}{L}}$. If the resistance of the circuit is small, the damping ratio will tend to zero and may result in an under damped oscillation due to a transient change in current. This insufficient damping will be interpreted in the frequency domain as a lack of phase margin. To analyse this, the circuit in Figure 5 can be split into two admittance parts, Y_1 and Y_2 and plotted in the frequency domain. This is shown in Figure 6 as bode plots of the magnitude and phase of the admittances.

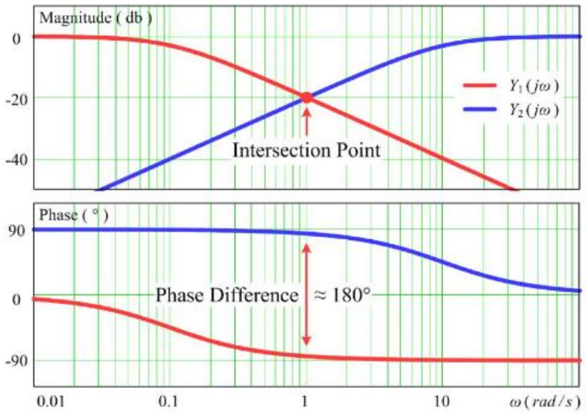


Figure 6 Bode plot of admittances Y_1 and Y_2

The intersection of the bode magnitude plot of Y_1 and Y_2 shows the resonance point where the phase difference is almost 180°. This means that the phase margin is almost zero leading to damping issues. This problem can be solved by modifying the passivity criteria of Re $(Y_1) > 0$ and Re $(Y_2) > 0$ to introduce a limit on the phase of the admittances. This will ensure that sufficient margin is available at the resonance frequency and if individual sub-systems satisfy the criteria, then the overall system will be passive as well.

However, the PCB is reliant on the concept that every individual sub- system should be passive. Since these sub-systems are provided by different OEMs there is no guarantee that every controller in the network is passive. So, the application of the concept in our case would be to identify if the overall system is passive. But this will not help us to identify the root cause of the problem.

3.1.3 Impedance based

Impedance based approach has been extensively used in the past to assess SSR problems. The traditional impedance scan method works well when only linear elements are present in the network. With the advent of non-linear devices such as converters, the traditional method does not work any longer and certain modifications are necessary. The modified impedance scan method is discussed in detail in Section 4.2.

The modified analysis method can be used to evaluate the performance of controllers in regulating the power output of the converter interfaced sources such as wind farms during and after a small disturbance. The impedance scans can also be used to determine the transfer function of the controller and to identify any oscillatory behaviours in the system. To accurately evaluate the frequency domain behaviour, impedance scans may need to be performed under different operating conditions and for different levels of power output from the wind farm.

In addition, impedance scan analysis can also be used to identify interactions between the wind farm controller and other components of the power system, such as other wind farm controllers, power system stabilizers, and voltage regulators. This information can be used to identify potential interactions between components that may affect the stability of the power system.

3.1.4 Eigenvalue based

An eigenvalue analysis is a mathematical technique that is used to determine the stability of a system by examining its eigenvalues. Eigenvalues represent the dynamic characteristics of a system and provide information about its stability, damping, and oscillatory behaviour. The stability of the system can be examined by observing the eigenvalues of the system matrix, which represents the power system state equations.

Each eigenvalue of a system will have a set of eigenvectors. These eigenvectors can be mathematically represented as left and right eigenvectors and the product of these are called state participation factor (SPF). It is a measure of the contribution of each state variable (e.g., voltage, current, angle) to the dynamic behaviour of the system. The SPF is calculated for each eigenvalue and provides information about which state variables are most important for the stability of the system. This information can be used to identify the most critical components of the power system and serve as a means of root cause analysis.

3.2 Time domain methods

3.2.1 EMT simulations

Electromagnetic transient (EMT) simulation refers to the simulation of changes in electric and magnetic fields over time in electrical power systems. It is a computational technique to accurately represent the behaviour of electrical power systems in response to rapidly changing electrical conditions, such as faults, switch events, energisation of assets, lightning events etc.

EMT simulations allows to capture the non-linearities in controllers accurately over a wide range of frequencies. This is suitable for controller interaction studies where the oscillation frequency is not limited to a typical range and therefore requires assessment of the system at all possible frequencies.

The limitation of EMT simulations is the lack of convenience to perform root cause analysis and the associated computational burden due to significantly lower integration step size compared to root mean square (RMS) simulations (more than an order of 100).

3.2.2 Co-simulations

Co-simulations is an approach to reduce the computational burden of EMT simulation by modelling certain parts of the network (such as a wind farm sites) in EMT domain and representing the remaining network in RMS domain. This reduces the simulation time significantly if a large network is studied. However, for connection studies it is only necessary to model the wind farm site in detail and the external grid is usually represented by an equivalent beyond one or two substations from the point of connection. In this case, co-simulation is not going to provide any significant advantage in terms of speeding up the simulations. Additionally, the limitation of root cause analysis in EMT simulations applies equally to co-simulation as well.

3.2.3 Real-time simulations

Real-time simulation such as Hardware-in-the-Loop (HIL) simulation refers to a type of simulation where real hardware components are integrated with a simulated environment to form a closed-loop system. In HIL simulation, the physical components of the power system, such as generators, transformers, power electronics devices, and control systems, are connected to a simulation environment, allowing for real-time interaction between the physical hardware and the simulated environment.

In HIL simulation, the physical components are connected to the simulation environment through a real-time interface, such as a programmable logic controller (PLC), a digital signal processor (DSP), or a field-programmable gate array (FPGA). The simulated environment is designed to mimic the behaviour of the power system and its components under various operating conditions.

This is widely used by equipment manufacturers for testing, validation, and verification of control systems and power electronics devices. However, HIL simulation requires special setup which has significant cost implications and requires specialist skills to develop, maintain and use. Also, studying a wide range of scenarios would be a time intensive process.

4 Industry experience

SSO phenomena can be triggered when there is a large disturbance such as a fault, in the presence of the frequency behaviour detailed in Table 4-1, on both the generator and the network sides for a 50Hz system. SSR and SSTI is included in this discussion since it aids a holistic understanding of SSO phenomena.

Table 4-1 Frequency Domain Behaviour of Interest in SSO Analysis for 50 Hz System

SSO Type	Generator Side Behaviour	Network Side Behaviour
SSR	Shaft modal frequency (ies) located at the 50 Hz complement of the network electrical resonant frequency	Network reactance minimum in proximity of 50 Hz complement of shaft modal frequency (ies)
SSTI (with power electronic controllers)	Shaft modal frequency (ies) located at the 50 Hz complement of the controller modal frequency[1]	Controller modal frequencies located at 50 Hz complement of shaft modal frequency
SSCI	Controller negative damping frequencies[2] located in same region as network electrical resonance frequency	Network reactance minimum in proximity of controller negative damping frequencies

The *reactance minimum* (on the network side) referred to in Table 4-1 for SSR and SSCI need not be a perfect series resonance point (where $X_L + X_C = 0$) but can also be where there is a large dip in reactance value within the sub-synchronous range.

The severity of the problem depends on both the percentage dip at a reactance minimum and the proximity of the dip to the 50 Hz complement of the shaft modal frequency (for SSR) or the negative damping frequencies of the controller (for SSCI).

The % dip can be calculated as

---

[1] This refers to an electrical frequency (ies) at which the controller displays SSTI behaviour with the mechanical shaft system. This is a different concept to that of a series resonance point in the network. Since this is not the focus of this analysis, further discussion thereof is omitted.

[2] Explained in upcoming paragraphs.

$$\%\, dip = 100 \times \frac{X_{f1} - X_{f2}}{X_{f1}}$$

Past investigations have shown that a 5% network reactance dip within ±3 Hz of the 50 Hz complement of the shaft modal frequency could be considered as a potential cause for concern. A generic example depicted in Figure 7 below could thus also present a scenario of interest for further analysis. Even though there is no perfect series resonance point, a 31 % dip in reactance is observed between 28 and 32 Hz.

For SSR analysis, any shaft modal frequencies between 15 Hz to 21 Hz (±3 Hz from 50 – 32 = 18 Hz) could potentially be exposed to SSR risk.

Conversely, when considering SSCI, one would be concerned with the controller's damping characteristics around the same frequency as that of the network dip/resonance, since there is a 1:1 relationship between the frequencies in the controller and the frequencies in the network - both are electrical frequencies and there's no electromechanical frequency ratio as is the case with SSR and SSTI.

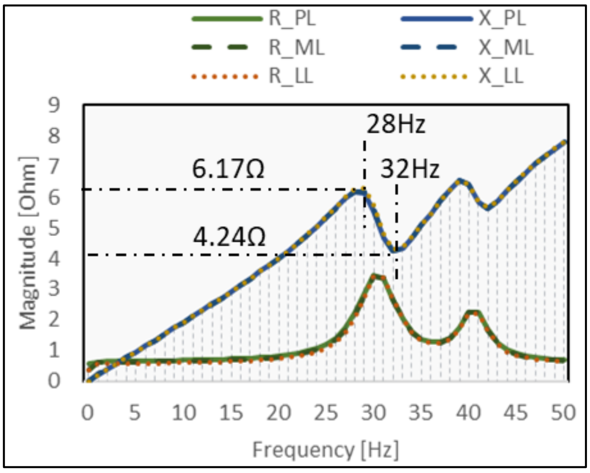


Figure 7 Generic Example of network impedance dip behaviour

Once the conditions for SSO exist, and it is triggered by a network disturbance, the resulting voltage and/or current oscillations will exhibit behaviours that is determined by the damping in the system.

In the case of SSR and SSTI, damping is electromechanical in nature – there's both mechanical damping offered by the turbine shaft masses, and electrical damping offered by the network. The mechanical and electrical systems act together to provide damping to the oscillations. Based on the magnitude of the electromechanical damping offered, the oscillations can either be well-damped, poorly damped, or undamped, the latter being the most severe case since this will subject shaft systems to growing torque stresses that could reduce shaft life and lead to shaft failures.

Electrical damping is just another way to refer to electrical resistance - the physical part of the network that dissipates energy from the system. Electrical damping is offered by two components – the resistance of network components such as transmission lines, cables, and transformers, as well as by loads.

For SSCI, damping is purely electrical in nature – there's the damping of the network, and the damping of the converter. The net resistance of the converter (virtual in nature) and the network determines the resulting behaviour. It is the unwanted negative damping offered by converter control system that causes the potential SSCI issue. Converter control systems (especially that of type 3 WTGs) can exhibit a negative resistance over a sub-synchronous frequency range.

There is in reality no such physical thing as *negative resistance*. Due to the converter action, sub-synchronous currents that enter the converter could experience amplification, which appears to be a negative resistance. The negative resistance behaviour can therefore be thought of as an *apparent resistance*. An example of such behaviour, for a type 3 WTG, is depicted in Figure 8 below.

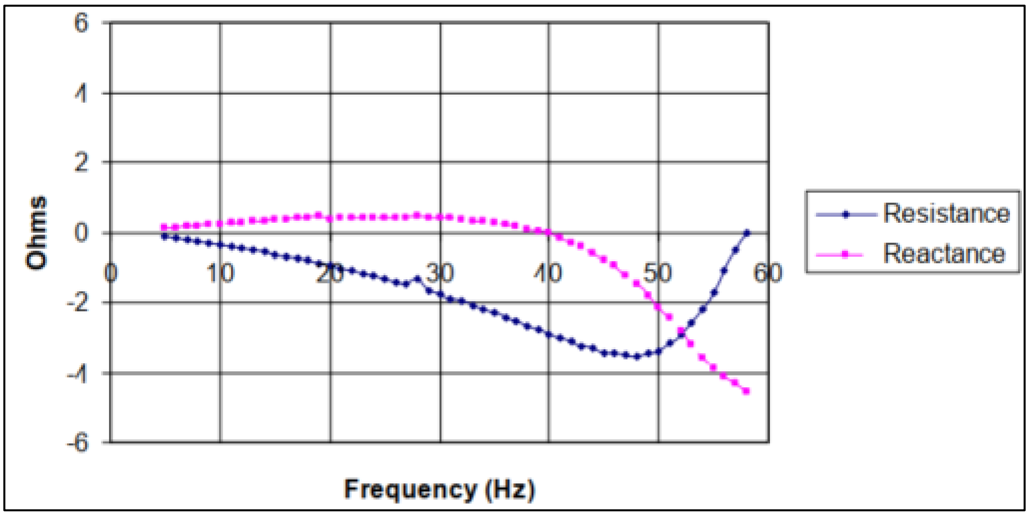


Figure 8 Example of converter negative subsynchronous damping [2]

Finally, it can be noted that several practical solutions exist to address SSO-related issues, such as static blocking filters, static excitation damping control, dynamic stabilisers and SSCI mitigation algorithms implemented in converter controls.

4.1 Frequency scan procedure for passive elements

Evaluation of system impedance versus frequency is a widely used measure to describe the linearized characteristics of an electrical system; the term Frequency Scan (FS) is used to describe this method. At any given frequency, the transfer function between current input and voltage output can be expressed as a complex number. The real part of the impedance is resistance, and the imaginary part is reactance.

A frequency scan analysis for passive elements is performed by looking from the POC of the Plant into the grid, to identify the presence of a reactance zero (or close to zero) when the net resistance of the system is negative, or any large dip in reactance value within the sub-synchronous range. This acts as a screening technique to quickly identify if the network presents conditions which could introduce risk of control interactions.

This is a well-known frequency domain analysis technique and has been used in several studies in the past to investigate both traditional SSR problems [23] and SSCI events [24]. Techniques such as the Ford-Fulkerson theorem can be applied to identify the relevant contingency sets which could lead to radial connection to series compensated lines in scenarios where it is not immediately apparent. A tool based on this concept is used by ERCOT screening the topology of the system and identify any (N-x) scenario which can lead to a radial connection of generator and series compensation devices. Figure 9 shows a flowchart of the approach to identify critical contingency scenarios.

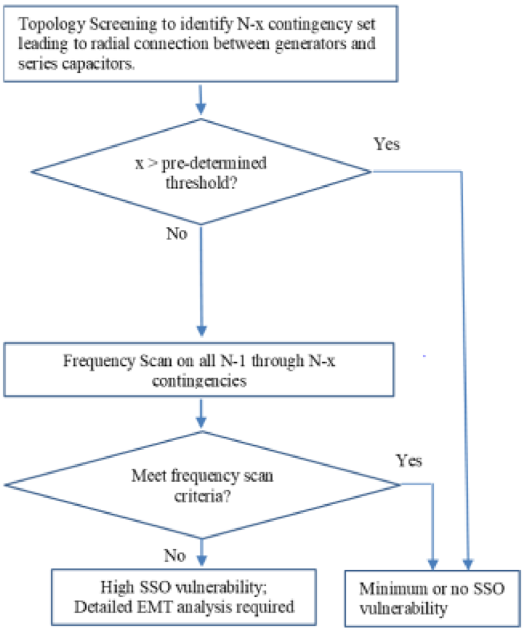


Figure 9 Topology screening tool used by ERCOT [24]

This frequency scan, also called a frequency sweep, is deemed to be passive, since it applies passive network reduction techniques in order to represent the impedance-vs-frequency characteristic for the network. Only passive RLC quantities are considered in such a scan, and no active or non-linear behaviour of the network is considered.

It is important, when conducting an SSCI assessment, that the network is not only considered under normal, contingency-free operation, but also under some realistic network running arrangements that could be expected under credible contingency scenarios.

4.2 Frequency scan procedure for active elements

The passive frequency scanning method introduced in the previous section for the network side is not appropriate for this purpose. Power electronic converters exhibit active and highly non-linear behaviour which cannot be accurately captured by traditional frequency sweep tools. To obtain an accurate approximation for the frequency domain behaviour of a converter, dynamic frequency scan techniques need to be applied.

To conduct a frequency scan to get the impedance of an electrical system (e.g., a WTG), a conventional procedure is as the following based on the figure below [25].

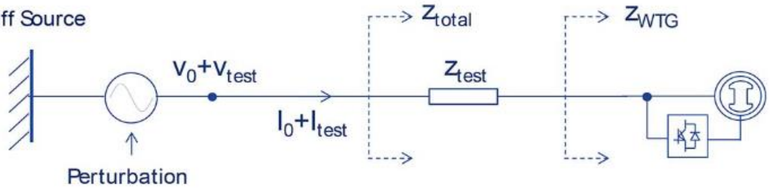


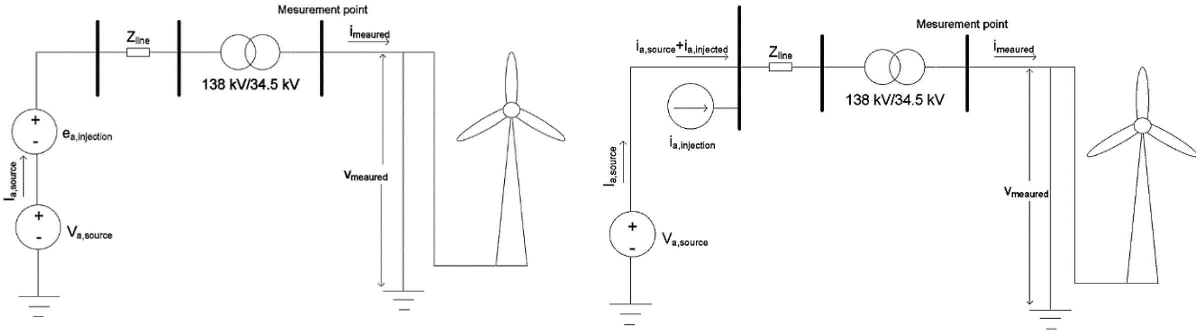
Figure 10 Schematic view of the frequency scan procedure

1. Establish a steady state operating condition of the system under test (e.g., power and reactive power out of the WTG, terminal voltage, frequency). Record the steady state quantities V_0, and I_0.
2. Inject a current (or voltage in this example) perturbation signal, V_{test}, superposed on the operating condition. The frequency of the perturbation signal is that at which the system impedance is to be derived. The magnitude of the perturbation signal has to be both small enough to retain the system's linearity and large enough to assure a reasonable signal-to-noise ratio. Record the resulting electrical quantities $V_0 + V_{test}$, and $I_0 + I_{test}$.
3. Subtract the perturbed system voltage and current by their steady-state quantities to acquire the small-signal delta change V_{test}, and I_{test}.

4. Perform FFT and calculate the phasor value at the testing frequency and calculate the impedance as $Z_{total} = \frac{V_{test}}{I_{test}}$. If a non-zero but known impedance, Z_{test}, is inserted between the measurement point and where the system impedance is to be derived, subtract Z_{test} to have $Z_{WTG} = Z_{total} - Z_{test}$
5. Repeat step 2 to 4 by sweeping the perturbation frequency over the range of interest.

4.2.1 Choice of injection signal

In practice, current injection has been used more often in grid side frequency scan while voltage perturbation used more often in WTG side frequency scan [25]. The application of current and voltage injection techniques for determining the harmonic impedance of the WTG are schematically shown below in Figure 11.



(a) (b)

Figure 11 Impedance measurement based on (a) voltage injection, and (b) current injection. [26]

In general, either a current signal or a voltage signal can be used as the excitation signal. For the DFIG WTG a voltage injection method is chosen due to a lower degree of sensitivity to variation of the injected signal, and hence minimizing the need for a trial-and-error approach; and the need for a smaller excitation signal, and hence creating a smaller disturbance on the operating point of the system. The advantages of the voltage injection method have been confirmed by performing several simulation case studies.

Using either method the WTG model should be disconnected from the interconnected network and connected to an ideal voltage source. This is because in the case of an unbalance in the network or presence of another power electronic components non-characteristic harmonics and inter-harmonics can be generated which may not be readily filtered out. Additionally, the focus of the investigation is the turbine side impedance scans whose characteristics are solely dependent on the response of the WTG and its controls. The remainder of the system can therefore be represented as an ideal voltage source.

4.2.2 Choice of injection and measurement location

Having selected a voltage signal as perturbation, the selection of injection and measurement points used in [26] is shown in Figure 12. This is because injecting the voltage at the WTG terminals would give rise to either steady-state instability of the turbine, or creates a too small signal, which cannot generate the desired harmonic impedance. To resolve these problems the location of the injection can be defined at a far busbar while keeping the same magnitude of injection and keeping the harmonic measurement directly after the turbine.

In test system described below, a known impedance $Z_{line} + Z_{tr}$, is between the measurement point and the WTG terminal. This impedance is beneficial to establishing the steady state, to limit the small signal magnitude if the converter exhibits singularity at certain frequencies, and to facilitate accurate frequency scan results. Note that while there are options in terms of the location of the injected signal, the measurement has to be performed at the turbine terminals.

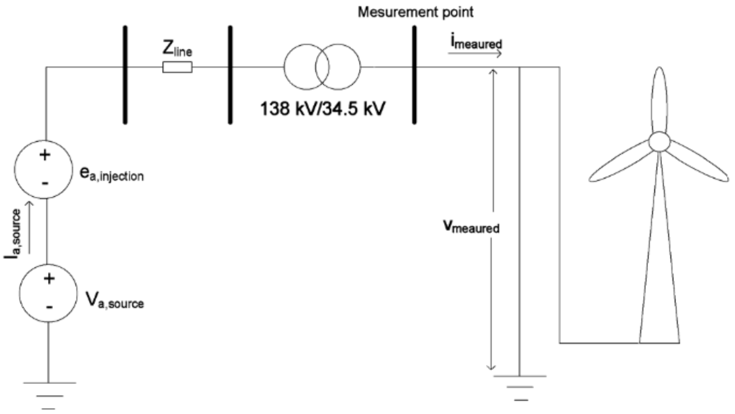


Figure 12 Point of injection and point of measurement for calculating harmonic impedance

4.2.3 Nature of injected signal

The main limitation of the conventional frequency scanning technique is that determining the harmonic impedance for each particular frequency requires one simulation run, and the time required to perform the frequency scanning for the entire sub-synchronous frequency range can be excessive. This is especially true when considering a very detailed model of wind turbine with fast acting controllers running at several kHz. Another disadvantage of this method is that it does not take account of system nonlinearities, and it is not therefore suitable for determining the harmonic impedance of WTG with highly non-linear behaviour. Lastly, with this method the power electronic devices such as a wind turbine are treated to be in the off-state conditions.

When considering a highly nonlinear and active device such as a wind turbine, the frequency scanning method utilized must be able to take account of these nonlinearities. The injected signal should be a wide-band signal such as an impulse or white noise to avoid the need for multiple simulation runs for each sub-synchronous frequency. An initial challenge dealing with the wind turbine or other similar nonlinear elements is that due to nonlinearities the principle of superposition does not hold true. The system, however, can still be considered linear for small changes around an operating point. The harmonic impedance of a non-linear system, linearized about an operating point can be determined with an FFT.

Another point to consider when determining the harmonic impedance of a WTG is the presence of sub-synchronous inter-harmonics prior to injection of the testing signal. In practice a WTG can generate inter-harmonics due to a) voltage unbalance in the supply, b) over-modulation of the PWM converters. If any appreciable level of inter-harmonics is generated by the WTG prior to the injection this has to be deducted from the voltage and current response of the turbine before deriving the harmonic impedance.

To test the background inter-harmonics of a WTG, harmonics are injected and later removed as the injected signals are removed it is expected that the measured current and voltage would drop to smaller values. The new steady-state values would be non-zero if any background inter-harmonics exist in the system. A case study, shown in the figure below, was conducted where a 0.1kV voltage is injected at the 138 kV voltage level and later removed at t=6 seconds. the study shows that there is no inter-harmonics generated by the tested WTG.

4.3 Frequency coupling effect

Both the grid and the WTGs are three phase AC systems, so the frequency scan can be conducted using phase, sequence, or *dq* domain. Modern power electronics control in many wind turbine generators features the use of *dq* rotating coordinates and the regulation of electrical quantities in these two axes. If the control is not perfectly symmetric, a coupling frequency effect may result in the sub-synchronous frequency range where SSCI stability is of concern. In fact, most real system control will be asymmetric as the two axes handle different control objectives, so this is a practical consideration. For more details on frequency coupling effect please refer to [25].

5 Analysis of black-box models

A state-space approach is best suited to identify the oscillatory modes of the system and to identify the root cause behind those modes [1] (for example, through identifying which states have high participation in the oscillatory modes, so that the corresponding power system component can be controlled to adequately damp the modes). The root-cause analysis of system instability, designing of damping controllers, etc. can be performed through eigen decomposition, participation factor analysis, eigenvalue sensitivity analysis, pole-placement, and other linear tools of stability analysis. However, the feasible applicability of state space analysis requires full knowledge of the model. This is termed as a 'white-box' model meaning that every state variable in the model is a known quantity. Some examples are synchronous generators and their associated controllers, like exciters and power system stabilizers, which makes state-space modelling tractable for conventional grids.

In contrast, the flexible options for designing the internal control system of an IBR, which dictates its dynamic characteristic, makes standardization difficult. Moreover, these control algorithms are vendor-specific and considered as commercial assets and, hence, are rarely disclosed as white-box models. Instead, the manufacturers provide 'black-box' models with input-output relationship without internal details. Most of these available black-box models are binary files (like .lib and .dll) which when incorporated with a numerical solver provide time-domain simulation results [27]. These simulations, especially for large practical power systems, can be time intensive (such as electromagnetic transient (EMT) studies) and they are insufficient to provide the underlying root-cause interpretations.

An alternative approach is to use impedance scan analysis. The impedance, or admittance, of a model represent the frequency-dependent relationship between the voltage and current. It is a linearised representation of the model capturing the small-signal dynamics through transfer functions or frequency spectra of input-output relationships.

The subsequent sections outline the existing methods of impedance modelling and how these methods can be extended to reveal more information from a black-box model to locate and identify the root cause of any oscillatory instability.

5.1 Impedance modelling in DQ domain and sequence domain

Several techniques are available in the existing literature to represent the impedance (or admittance) model of IBRs [28]. Some of these techniques include *dq* impedance model, stationary-frame (αβ)-based impedance model, sequence domain and modified sequence domain impedance model, and phasor-based impedance model. Among these, the most widely used are the impedance model based on *dq* domain and sequence domain. Both these domains have several characteristic merits and demerits, and impedances of one domain can be derived from the other [29], [30]. The transfer function-based impedance, and admittance model of a generic three-phase IBR in the *dq* frame [31] is first discussed here.

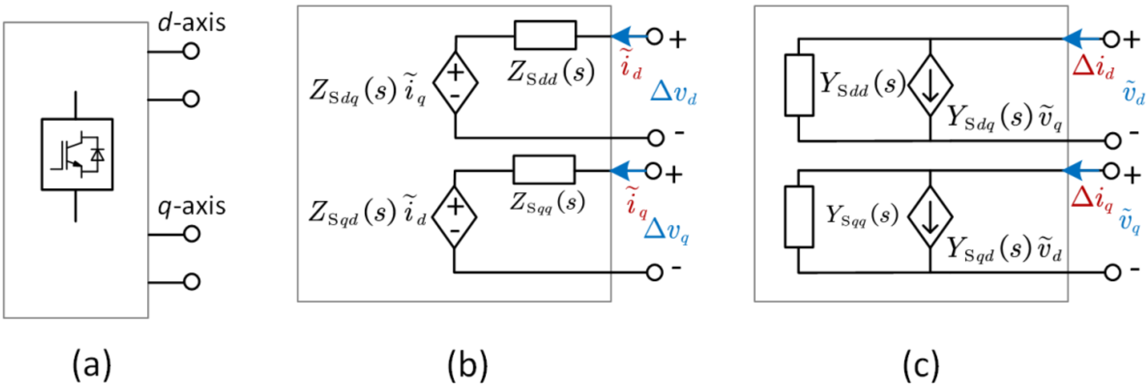


Figure 13 Three-phase IBR in dq frame (a) two-port depiction, (b) impedance model, (c) admittance model [31]

As shown in Fig. 1(a), a three-phase IBR is observed as a two-port system in the *dq* frame. Fig. 1.4(b) and Fig. 1.4(c), show the two Thevenin equivalent circuits and two Norton equivalent circuits to demonstrate the IBR impedance and admittance models, respectively. The coupling between the q-axis and d-axis is captured by the current-controlled voltage source, and the voltage-controlled current source, respectively. The relationship between the small-signal output voltage and output current is given as

$$\begin{bmatrix}\Delta v_d\\ \Delta v_q\end{bmatrix} = \begin{bmatrix}Z_{Sdd}(s) & Z_{Sdq}(s)\\ Z_{Sqd}(s) & Z_{Sqq}(s)\end{bmatrix}\begin{bmatrix}\tilde{i}_d\\ \tilde{i}_q\end{bmatrix} \tag{1}$$

and

$$\begin{bmatrix}\Delta i_d \\ \Delta i_q\end{bmatrix} = \begin{bmatrix}Y_{Sdd}(s) & Y_{Sdq}(s) \\ Y_{Sqd}(s) & Y_{Sqq}(s)\end{bmatrix}\begin{bmatrix}\tilde{v}_d \\ \tilde{v}_q\end{bmatrix} \tag{2}$$

from which the impedance model of the IBR, $Z_S(s)$ is given as

$$Z_S(s) = \begin{bmatrix}Z_{Sdd}(s) & Z_{Sdq}(s) \\ Z_{Sqd}(s) & Z_{Sqq}(s)\end{bmatrix} \tag{3}$$

and the admittance model, $Y_S(s)$ is given as

$$Y_S(s) = \begin{bmatrix}Y_{Sdd}(s) & Y_{Sdq}(s) \\ Y_{Sqd}(s) & Y_{Sqq}(s)\end{bmatrix} \tag{4}$$

As can be seen in Fig. 1(b) for the impedance model, in addition to the self-inductances Z_{Sqq} or Z_{Sdd}, each circuit contains a small-signal current-controlled voltage source (a trans-impedance) Z_{Sdq} or Z_{Sqd} that represents the cross-coupling terms between *q*-axis and *d*-axis.

Impedance modelling can also be obtained in the sequence domain through harmonic linearization and symmetrical component method. By neglecting the zero-sequence component and using the Fortescue transform, and Park transform matrices, a relationship between the *dq* and sequence impedance can be obtained [30]. The voltage phasors (and similarly current phasors) in the *dq* and sequence domains are related as

$$\begin{bmatrix}V_d \\ V_q\end{bmatrix} = \frac{\sqrt{3}}{2}\begin{bmatrix}1 & 1 \\ -j & j\end{bmatrix}\begin{bmatrix}V_p \\ V_n\end{bmatrix};\ \omega_{dq} = \omega_p - \omega_1 = \omega_n + \omega_1 \tag{5}$$

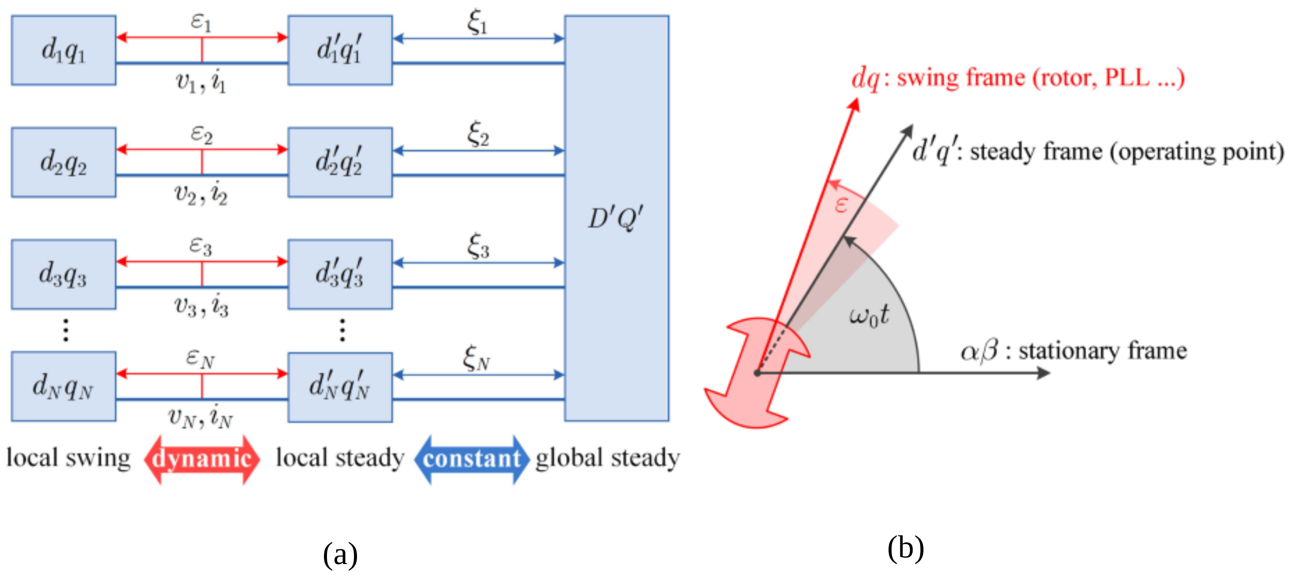
i.e. a general *dq* voltage phasor at a frequency ω_{dq} is equivalent to two sequence-domain voltage phasors at different frequencies, shifted by the fundamental (ω_1) (i) positive sequence voltage at $\omega_p = \omega_{dq} + \omega_1$; (ii) negative sequence voltage at $\omega_n = \omega_{dq} - \omega_1$. The *dq* and sequence transfer impedance matrix are related as

$$Z_{pn} = A_z . Z_{dq} . {A_z}^{-1} \tag{6}$$

$$Z_{dq} = {A_z}^{-1} . Z_{pn} . A_z \tag{7}$$

where $A_z = 1/\sqrt{2}\begin{bmatrix}1 & j \\ 1 & -j\end{bmatrix}$, and ${A_z}^{-1} = {A_z}^{*} = 1/\sqrt{2}\begin{bmatrix}1 & 1 \\ -j & j\end{bmatrix}$. The off-diagonal elements in Z_{pn} can be viewed as frequency coupling terms, since they represent cross-coupling between the positive sequence frequency ω_p and the negative sequence frequency ω_n. This frequency coupling was defined as Mirror Frequency Coupling (MFC) in [30], as the two coupled frequencies are separated by an equal distance from the fundamental frequency ω_1 (mirrored). Thus, through (6) and (7), the relationship between the impedance model in these two domains is established. Impedances in the sequence domain-based method can be measured through simplified experimental setups, whereas in the *dq* domain, the d-axis and q-axis components cannot be measured through experiments directly [32]. But once the impedances are derived in the *dq* domain, they are more compatible and mathematically less rigorous. Also, the *dq* domain is compatible with the overall system model. Hence, we will proceed with the impedance model in the *dq* domain.

The impedances (or admittances) models in *dq* domain are evaluated locally in the individual reference frame of the IBRs. For a composite grid with multiple and varying generation portfolio, it becomes critically important to represent the local impedances in a global reference frame before proceeding to develop the overall system impedance model using standard circuit analysis, and then to perform impedance-based stability studies using General Nyquist Criteria (GNC) [33]. Unlike the state-space model where this local-to-global frame alignment is relatively simple due to the availability of the local angle (rotor or PLL angle) as a system state, the impedance model does not explicitly include the angle information. Very few literatures in impedance modeling have tackled this frame alignment problem. One approach has been to model every machine directly in the global reference frame where the frame dynamics appear as explicit paths in the impedance diagram [34]. Such an approach reduces scalability and is difficult to implement. A promising approach has been to apply a rotation matrix for the reference transformation using load flow information [35],[36]. The alignment technique effectively provides the necessary frame transformation with the assumption that the local phase angles are constant. Such assumptions become invalid for low-inertia systems where the local frame dynamics of PLLs cannot be neglected [31].



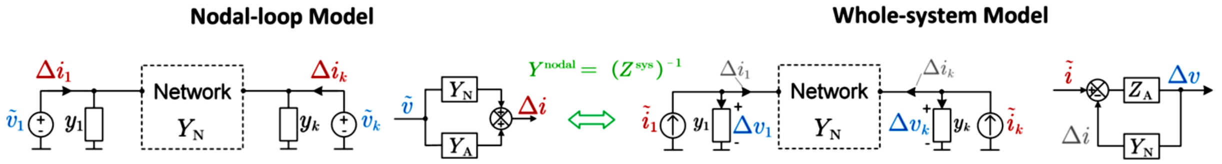
(a) (b)

Figure 14 (a) 2-step frame alignment, (b) local swing frame dq which swings around a local steady frame d'q' [37] [33]

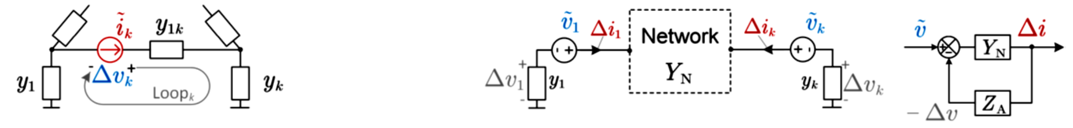
A two-step local-to-global frame transformation has been proposed in [37]. Here, as shown in Fig. 2(a), the impedance transformation is carried out from (a) local swing frame to local steady frame (dynamic to constant), and (b) local steady frame to global steady frame (constant to constant). Unlike the previous methods [34]–[36], which only performed the second step (local steady to global steady) through a fixed angle rotation (ξ) determined by the load flow solution, this method is more effective in capturing the local frame dynamics which are prevalent in PLL and rotors. The local impedance models are developed in the local swing *dq* frame, aligned to the instantaneous PLL angle or rotor angle, and can swing around the local steady frame by an angle (ε), as shown in Fig. 2(b). The local steady $d'q'$ frame is aligned to the local steady-state operating point and rotates at a constant speed and has no dynamics.

5.2 Impedance modelling of the overall system

Once all the machine impedance models are transformed to the global frame, the overall system model is developed next. Two types of system model formulation methods are found in literature (i) Nodal-loop Model [38], [39] in which circuit equations from either nodal or loop analysis are assembled in a matrix format, and (ii) Whole-system Model [35], [37] in which network branch admittances and impedances of apparatus at nodes are combined in a feedback loop and all elements contain dynamics of the whole-system.



(a) Nodal Admittance Model $\Delta i = Y^{nodal} \cdot \tilde{v}$ (c) Whole-System Impedance Model $\Delta v = Z^{sys} \cdot \tilde{i}$



(b) Loop Impedance Model $\Delta v = Z^{loop} \cdot \tilde{i}$ (d) Whole-System Admittance Model $\Delta i = Y^{sys} \cdot \tilde{v}$

Figure 15 The four types of networked impedance models and their characteristics (a) Nodal Impedance Model, (b) Loop Impedance Model, (c) Whole-system Impedance Model, (d) Whole-system Admittance Model [40]

5.2.1 Nodal loop model

In the first case, a set of circuit equations are established from the nodal or loop circuit analysis of a network and assembled into a transfer-function matrix to form a nodal-loop model. There are two formats: a nodal admittance model Y^{nodal} and a loop impedance model Z^{loop}. The formulation of Z^{loop} is non-unique as it can be formulated in a variety of ways for a given grid because there are multiple legitimate ways to define a set of loops covering the network. Hence it is better to proceed with Y^{nodal} formulation. The formulation of nodal admittance model of the system proceeds by connecting the admittance of apparatus Y_A with the admittance of the network Y_N, through a process which can be understood from considering a virtual injection, as shown in Fig. 3(a). Y^{nodal} expresses how a vector of voltage perturbations $\tilde{v}$ applied at the nodes of the grid creates a vector of corresponding changes in nodal currents Δi. Such formation is a simple open-loop relationship as

$$Y^{nodal} = Y_A + Y_N \tag{8}$$

Similarly, Z^{loop} expresses how a vector of current perturbations applied to each loop current of the grid creates a vector of corresponding additional loop voltage drops around the loop, as shown in Fig. 3(b).

As each element of the Y^{nodal} matrix refers to a physical element in the system, this modelling approach can be used for a more systematic and informative root-cause analysis when the structure and parameters of the network are known. It should be noted that Y^{nodal} cannot be measured online.

5.2.2 Whole system model

The whole-system impedance model Z^{sys} and the whole-system admittance model Y^{sys} are formed as illustrated in Fig. 3(c) and Fig. 3(d). Taking Z^{sys} as an example, the system is first separated into a diagonal impedance matrix Z_A, where the diagonal-entries are the impedances of apparatus at each node, and the network nodal admittance matrix Y_N containing the admittances of branches. Then, the model is formed with a virtual nodal injection of current $\tilde{i}$ whereas for the nodal admittance model the injection is in nodal voltage. The current $\tilde{i}$ causes a change in current through the apparatus and therefore a change in the apparatus voltage Δ*v*, which in turn creates a change of current flowing into the network Δ*i*, which changes the current flow in the apparatus in a feedback fashion (closed loop), as shown in Fig. 3(c). The response Δ*v* to perturbation $\tilde{i}$ is -

$$\Delta v = Z_A(I + Y_N Z_A)^{-1}.\tilde{i} \qquad (9)$$

From (9) we get Z^{sys} as

$$Z^{sys} = Z_A(I + Y_N Z_A)^{-1} \qquad (10)$$

Similarly, the formulation of whole-system admittance model is shown in Fig. 3(d) and

$$Y^{sys} = (I + Y_N Z_A)^{-1} Y_N \qquad (11)$$

defined as

5.2.3 Relation between nodal loop and whole system model

From Fig. 3, it can be seen that by changing the voltage injections into current injections, Y^{nodal} will become Z^{sys}, hence we have -

$$Y^{nodal} = \left(Z_{sys}\right)^{-1} \qquad (12)$$

Also,

$$\left(Y_{sys}\right)^{-1} = Y_N{}^{-1} + Y_A{}^{-1} \qquad (13)$$

5.2.4 Challenges with Y^{nodal}

One of the primary challenges lie in measuring Y^{nodal}. In practice, nodes will have stiff voltage sources, which makes perturbing the node voltage in parallel, which is required for online measurement of Y^{nodal} not feasible. Also, analysing the modal behaviour and participation factors using Y^{nodal} would require its complete information as the modes are calculated from its determinant [31]. Hence, any missing measurement or unknown elements of Y^{nodal} will consequently not allow to perform such studies. Moreover, if impedance models are also not available from apparatus vendors, or do not account for operating point variation, then practical application of eigenvalue sensitivity in the nodal admittance model may not be possible.

5.2.5 Proposed approach

The elements of Z^{sys} do not directly relate to a particular physical element in the network (unlike Y^{nodal}), instead, each element holds the information of the complete system dynamics. Also, unlike Y^{nodal}, elements of Z^{sys} can be measured online (as explained above). Hence, the whole-system modelling approach would be better for this project.

5.3 Root cause analysis – 'grey box' approach

Stability assessment of large power systems using the generalized Nyquist criterion, or several Bode plots becomes impractical. As such these methods do not provide insights or the root cause of possible unstable operating conditions to power engineers and transmission system operators (TSOs), so that corrective actions can be taken [38]. When considering large power systems through impedance models, using either nodal-loop model or whole-system model format, it is desirable to replace generalized Nyquist criteria and phase or gain margins with eigenvalue analysis and pole-zero assessment supplemented with modal analysis, such as sensitivity and participation analysis [41]–[43]. Creating such analysis for impedance models to build relationships between impedance models and state-space models will allow to look inside the black-box models and achieve almost the same level of transparency as white-box state-space models. This has been described as a grey-box approach in [44] for seamlessly combining the white box and black box models, so that stability analysis and root cause tracing can be facilitated without requiring the disclosure of internal details of the black box part of the combined system model.

In the next sections the key concepts that are fundamental to the grey-box model approach are provided.

5.3.1 Relationship between network impedance and state space

The state-space representation of a power system of *m* order and with *n* nodes can be represented as

$$\dot{x} = Ax + Bu \tag{14}$$

$$y = Cx \tag{15}$$

where A, B and C are the state matrix, input matrix and output matrix. x, u and y are the state vector, input vector, and output vector. The state equation is usually of very high order and the general approach is to break it down into a series of first order equivalents through coordinate transformation such that the state matrix in the new coordinate is diagonalized. Through eigen-decomposition and coordinate transformation, we consider $z = \Psi x$, and $x = \Phi z$, where Ψ, Φ correspond to the left eigenvector matrix and right eigenvector matrix, respectively, and $\Psi = \Phi^{-1}$. For whole-system impedance model, coordinate transformed system can be obtained for the choice of injection as input and observations taken as output. For the Z^{sys} case, the input would be the vector of current injections $(u = i)$ and output is the vector of apparatus voltages measured $(y = \Delta v)$.

$$\dot{z} = \Lambda z + \Psi B i \tag{16}$$

$$\Delta v = C\Phi z \tag{17}$$

where $\Lambda = \Psi A \Phi = diag(\lambda_1, \lambda_2, \dots \lambda_N)$. For this state-space model, Z^{sys} is essentially the transfer function, which can be expressed as

$$Z^{sys} = C(sI - A)^{-1}B = C\Phi(sI - \Lambda)^{-1}\Psi B \tag{18}$$

Hence the *k*th element of Z^{sys} is given as

$$Z_{kk}^{sys}(\mathrm{s}) = \sum_{i=1}^{m} \frac{c_k \phi_i \psi_i b_k}{s - \lambda_i} = \sum_{i=1}^{m} \frac{R_{ki}}{s - \lambda_i} \quad (19)$$

where R_{ki} is the residue of $Z_{kk}^{sys}(\mathrm{s})$ corresponding to the pole λ_i and is given as

$$R_{ki} = c_k \phi_i \psi_i b_k \quad (20)$$

One of the key observations made from (19) is that all the elements of Z^{sys} (for any k) share the same poles (λ), which are the eigenvalues of the system.

5.3.2 Participation analysis in impedance models

The information obtained from the previous section regarding residue will be instrumental in defining the participation factors in impedance models. From (20) a special scenario can be inferred when matrix B and C are considered identical, i.e., when the kth element in c_k and b_k are equal to 1 and all other elements are equal to 0. In such a hypothetical case

$$R_{ki} = \psi_{ik} \phi_{ki} = p_{ki} \quad (21)$$

where p_{ki} is the participation factor in the state space model. Hence in this special case, the residue of Z_{kk}^{sys} becomes identical to the state participation factor. Although the scenario is non-practical regarding the assumptions on B and C matrices, however it provides an important hint that residues play an important role in defining participation analysis in impedance model.

Lemma 1 defined in [31] establishes such a relationship with analytical rationale. Without going in-depth into the mathematics, the lemma is stated here as follows -

$$\Delta\lambda = \langle -Res_{\lambda}^{*} G_\rho, \Delta H_\rho \rangle_F \quad (22)$$

Here $\langle\cdot,\cdot\rangle_F$ is the Frobenius inner product of two matrices, G_ρ is a square transfer matrix depending on parameters ρ and with non-repeated poles λ. $H_\rho = {G_\rho}^{-1}$ is the inverse transfer function of G_ρ. (22) is formulated when the parameters are perturbed infinitesimally by $\Delta\rho$, and λ and H_ρ are perturbed by $\Delta\lambda$ and ΔH_ρ correspondingly. Lemma 1 in [44] proves that in a system where a parameter is perturbed, $\Delta\lambda$ can be calculated via the corresponding residues of the-whole system models. Taking $G_\rho = Y^{sys}$, we get corresponding H_ρ as

$$H_\rho = Z_{Ak} + Z_{Gk} \tag{23}$$

where Z_{Ak} is the impedance of the apparatus connected at the *k*th node and Z_{Gk} as the impedance of the rest of the grid as seen from the *k*th node. When subjected to perturbation, from (23) we get $\Delta H_\rho = \Delta Z_{Ak}$, as $\Delta Z_{Gk} = 0$. We can now use Lemma 1 and say -

$$\Delta\lambda = \langle -Res_\lambda^* Y_{kk}^{sys}, \Delta Z_{Ak}(\lambda)\rangle_F \tag{24}$$

As the poles of the transfer function Y_{kk}^{sys} are exactly equal to the eigenvalues of the system, we can conclude from (24) that the sensitivity of an eigenvalue to an apparatus impedance is determined by the whole system admittance as seen by that apparatus, and the *impedance participation factor* is defined as

$$p_{\lambda,Z_{Ak}} = -Res_\lambda^* Y_{kk}^{sys} \tag{25}$$

and (24) becomes -

$$\Delta\lambda = \langle p_{\lambda,Z_{Ak}}, \Delta Z_{Ak}(\lambda)\rangle_F \tag{26}$$

Using the impedance-admittance duality, the *admittance participation factor* can also be calculated as

$$p_{\lambda,Y_{Ak}} = -Res_\lambda^* Z_{kk}^{sys} \tag{27}$$

such that

$$\Delta\lambda = \langle p_{\lambda,Y_{Ak}}, \Delta Y_{Ak}(\lambda)\rangle_F \tag{28}$$

Now, if the sensitivity of the apparatus impedance to its parameters ρ, is known, i.e.

$$\Delta Z_{Ak}(\lambda) = \frac{\partial Z_{Ak}(\lambda)}{\partial \rho}\Delta\rho \tag{29}$$

then the *parameter participation factor* is defined as

$$p_{\lambda,\rho} = \langle p_{\lambda,Z_{Ak}}, \frac{\partial Z_{Ak}(\lambda)}{\partial \rho} \rangle_F \tag{30}$$

such that

$$\Delta\lambda = p_{\lambda,\rho}\Delta\rho \tag{31}$$

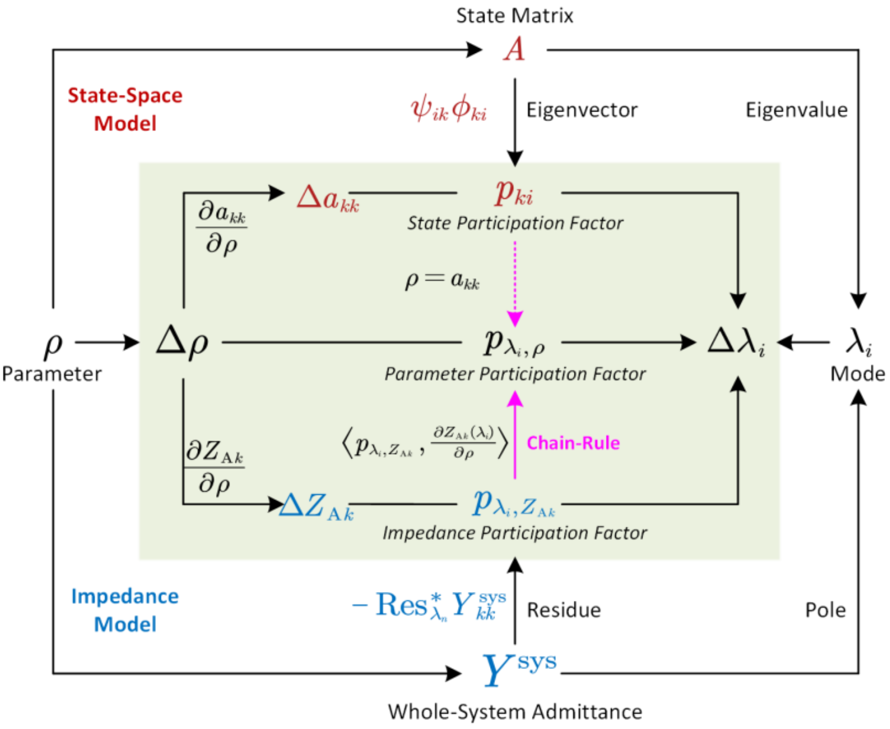


Figure 16 Relationship between participation factors in state-space and impedance models [44]

The information of $p_{\lambda,\rho}$ provides insights on how to tune a parameter to move λ in a desired direction in the complex plane. A special case is obtained when the selected parameter is a_{kk}, in which the corresponding parameter participation factor becomes the state participation factor -

$$p_{ki} = p_{\lambda_i,\rho} = \langle p_{\lambda,Z_{Ak}}, \frac{\partial Z_{Ak}(\lambda_i)}{\partial a_{kk}} \rangle_F \tag{32}$$

Equations (30) and (32) are called the chain-rule of participation factors. The impedance participation factor allows us to evaluate the participation of an apparatus in system oscillations through only black-box models. On top of this, the chain-rule yields the state and parameter participation factors, which further enables to look inside the black-box and trace root-causes to detailed parameters and states without disclosing the state equation. The relationship between the different types of participation factors in the state-space and impedance models are summarized in Fig. 4.

Based on this chain-rule of participation factors, the grey-box approach for power system stability analysis was developed in [44].

5.3.3 Grey box method

The grey-box approach works in a layered manner [40], [44]. Each layer has different transparencies in accordance with the available prior knowledge, as shown in Fig. 5. Depending on the availability of prior knowledge, more transparency can be obtained which would in turn would provide further information for root-cause analysis and troubleshooting.

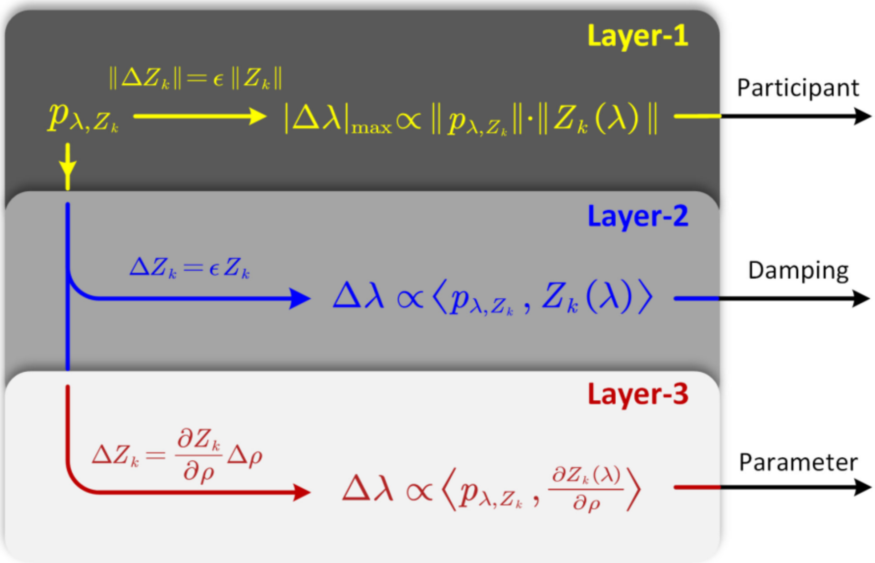


Figure 17 The three-layered grey-box [44]

In the 1st layer, the impedances of all the apparatus of the system Z_k, and the network admittance matrix (Y_N) are required. Using this prior knowledge, the impedance participation factor p_{λ,Z_k} is calculated using (25). Using the Cauchy inequality[3] on (26), the upper bound of eigenvalue variation $|\Delta\lambda|_{max}$ is calculated, which indicates the maximum

---

[3] https://en.wikipedia.org/wiki/Cauchy%E2%80%93Schwarz_inequality

possible *participation* of the corresponding apparatus. Hence in Layer 1, the potential participants in a mode are identified.

In the 2nd layer, with the same knowledge that is required in Layer 1, the directional information of Z_k and p_{λ,Z_k} is used to identify the negative *damping* contributors. It is assumed that any apparatus impedance perturbation (ΔZ_k) is aligned to its original impedance by a scaling factor. This when applied to (26) implies the impact of the apparatus impedance changes on the real part of $\Delta\lambda$, which would be indicative of the impact on system damping.

In the 3rd layer, additional information on the sensitivity of Z_k with respect to different parameters of the apparatus is required. This layer aims to investigate the root-cause of instability by providing information on the parametric sensitivity using (29) – (31).

Thus, different layers of the grey-box approach can be used depending upon how much information about the black box model is available.

5.3.4 Summary

The next work package will adopt this approach to undertake a proof-of-concept study using a test network. It will integrate a test black box model of a converter with the white box model of the rest of the network using the interfacing scheme of two-step local-to-global frame transformation as shown in Figure 14 and the whole system modelling method described in Section 5.2.2. This will be followed by applying the grey box approach to the developed model for participation analysis.

6 Automated identification

As discussed in previous Sections, SSO can be of various types, namely subsynchronous resonance (SSR), subsynchronous torsional interaction (SSTI) and subsynchronous control interaction (SSCI), depending on the components that interact with each other. For example, an SSTI occurs when a generator's torsional masses interact with PE-based HVDC converters [7] whereas SSCI occurs when power electronic control systems interact with other nearby power electronic devices [7]. If these oscillations are not adequately damped, they can lead to disconnection or equipment damage.

Grid Code requirement ensures that new connections do not introduce these types of oscillations to the system and do not deteriorate any oscillations originating from the system. This is verified by performing time-domain simulations for a set of scenarios to prove compliance with the grid code [10]. The number of scenarios studied is limited due to computational complexity and therefore it does not consider scenarios too much in the future. As the number of wind and solar connections increases it becomes essential to study a broader range of scenarios. This allows us to understand better the impact of future network changes, sources of uncertainty and variability on the occurrence of SSO events. Moreover, it enables better-informed decisions as a broader set of scenarios is covered. To enable screening of a large number of scenarios, automated identification procedures are necessary.

6.1 Overall approach

An overview of the proposed approach is presented in Figure 18. The process can be split into three main steps – (1) conversion of EMT model (including proprietary models) to frequency domain models, (2) scenario creation and filtering of scenarios for hidden threats using frequency domain techniques and (3) automated identification of SSO events from time domain simulations of filtered scenarios.

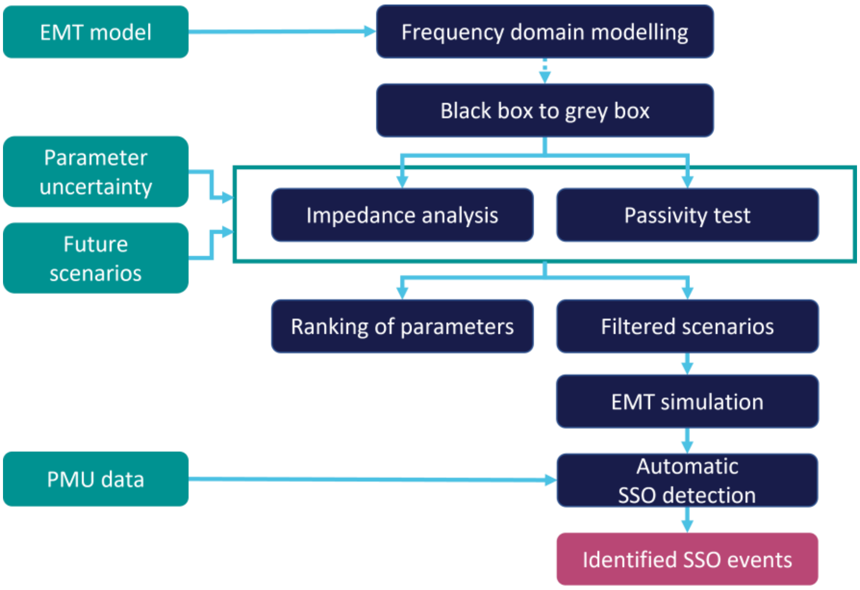


Figure 18 Overview of the proposed analysis process

For systems where non-proprietary (i.e., white box) model is available, it is possible to perform eigenvalue, sensitivity, and participation factor analysis (also known as modal analysis). This is realised by linearising the system around equilibrium and computing the eigenvalues and eigenvectors of the system state matrix $\boldsymbol{A}$. The location of the eigenvalues indicates which modes are present in the system and whether these are stable or could lead to oscillations when excited. By assessing how the eigenvalues change under the addition of components or variation of system parameters, predictions can be made about which scenarios can lead to SSO.

A different approach is to use impedance-based stability analysis, as discussed in Section 3.1.3. This method only requires knowing the frequency-dependent impedances of the converter and grid (works with proprietary controllers). This makes the technique highly attractive as it does not require manufacturers to disclose sensitive information about their design. Predictions about the stability are made by finding the gain and phase margins from the corresponding bode plots. A similar but slightly different approach uses the passivity criterion to evaluate the stability, which is applied to the system's total impedance rather than a ratio of the separate impedances.

Using these methods, it is possible to identify system parameters important for SSO identification and filter out scenarios that could contain hidden threats in the network. Note that all these methods are significantly faster than performing time-domain simulations on a diverse set of cases. This is because time domain simulation relies on solving equations over time whereas frequency domain methods linearise the system around an operating point and tries to identify the behaviour as a function of the frequency.

6.2 Functional requirements and attributes

We need to define a set of functional requirements for the automated identification algorithm. This will guide the attributes of the developed platform and help us to assess the suitability of the process to solve this particular problem.

Functional requirements:

1F. The algorithm needs to identify SSO events from time domain results

2F. The identification accuracy should be sufficiently high (>85%) for it to be used in future as a BAU tool

3F. The algorithm should work for a range of SSO frequencies (1Hz to 47Hz)

Attributes:

1A. *Good generalisation performance*: Because this algorithm needs to evaluate various scenarios and measurement points (PMU sources), its method needs to be sufficiently robust to handle different input sources.

2A. *Scalable*: The method should be scalable to larger sets of data without affecting the performance.

3A. *Automation*: Little to no human intervention required for the identification process.

4A. *Accessible*: The system should be developed using open-source software.

5A. (OPTIONAL) *Explainability*: The method should be interpretable (or transparent) so that it is easy to understand, maintain and improve the system.

6A. (OPTIONAL) *Execution speed*: Since a wide range of scenarios should be considered, the method should be fast. Moreover, the tuning of the process should also be achievable in a reasonable amount of time.

Various approaches exist to realise this, but most rely on signal processing techniques or machine learning approaches. We have adopted a hybrid approach using both feature engineering and a machine-learning-based classifier to achieve the requirements of the project. These are discussed in detail in Section 7.

7 Feature engineering

Data-driven methods, such as machine learning, are powerful because they allow performing various predictive tasks by capturing trends and relations from the data which humans might not be able to identify. However, using raw data is not always a good idea, even when much data is available. Deep neural networks can be used for complex tasks such as image recognition and generation [15, 16], computational photography [17], or autonomous driving [18, 19] but are notoriously hard to train. By transforming the data into new features, the performance of a model can be improved significantly in terms of accuracy and training time.

In this chapter, feature engineering for SSO detection is discussed. First, a brief description of the problem is provided in Section 7.1. Second, possible approaches from the literature are compared (Section 7.2). In Section 7.3, the implemented method is explained. Finally, the feature extraction is validated in Section 7.4.

7.1 Problem description

From the frequency domain methods, scenarios arise that could lead to SSOs. These scenarios are simulated using EMT simulations, and measurements are saved at various points in the network. An example of such a time-domain measurement can be seen in Figure 19. The signal is the A-phase current of a network in pu, containing an SSO starting around 80 s.

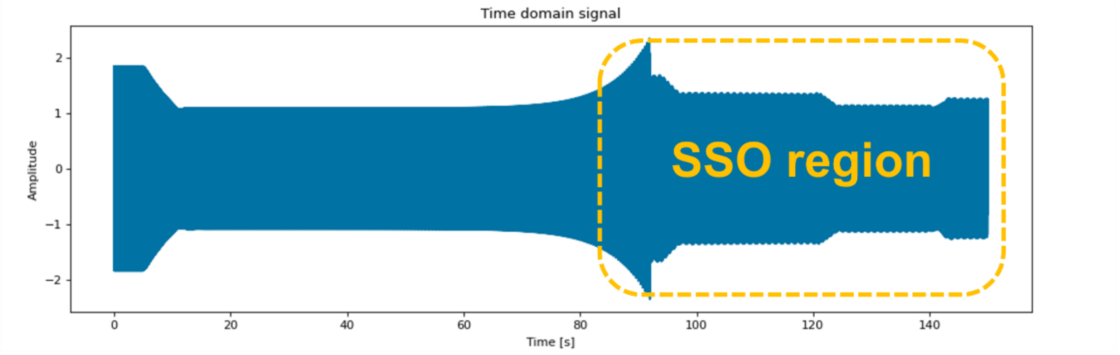


Figure 19 Example of a current measurement in pu containing an SSO. Created with data from [20]

The raw measurement is not straightforward to identify SSOs quickly. In the SSO region, a change in amplitude is seen. However, a change in the voltage or current levels is not unique to SSOs but can have various reasons, e.g., enforcing different setpoints, topology changes, and faults. [21] provides a detailed overview of mathematical models of power quality disturbances, showing which issues can lead to changes in amplitude. Moreover, a change in the oscillatory frequency could be observed. But to extract these changes, a

Fourier Transform is applied, which already moves into a different domain. In conclusion, the data needs to be manipulated to transform it into a more useful form.

This process of feature engineering should create features or metrics that are generic enough (requirement 1A) to identify SSOs (requirement 1F) without the need for human intervention (requirement 3A) from different datasets. The method should also be scalable (requirement 2A).

Optionally, the features should be explainable (requirement 5A) and quick to execute (requirement 6A). Note that if the method is sufficiently fast, it can be deployed on larger datasets and can, therefore, also satisfy requirement 2A.

Finally, the implementation should be accessible (requirement 3A), which can be realised by implementing the subsystem in an open-source language such as python.

7.2 Possible approaches

SSOs are becoming more prominent in the energy network, which has also caught the eye of the scientific community. This has led to the development of various methods that can be used to detect and understand SSOs. In 2020, a technical report was created by the IEEE Power & Energy Society to summarise the research [9]. [21] provides an excellent summary of methods that can be used to identify and qualify power quality disturbances. The approaches can be roughly divided into signal processing techniques and data-driven methods.

7.2.1 Signal processing methods

These types of methods originate from signal processing theory and usually involve applying a mathematical transform. The new domain allows analysing certain signal aspects more easily than in the original (time) domain. Various methods exist, which are briefly elaborated on below. [22, 23] show a side-by-side comparison of the different techniques.

The most famous example is the Fourier Transform (FT), which decomposes a signal into separate components of a single frequency. During an SSO event, the FT will show frequency components different from the fundamental frequency, making the FT a good feature for our purpose. Moreover, efficient implementations exist, such as the Fast-Fourier Transform (FFT), the one-sided FT and the Discrete Fourier Transform (DFT). The latter is usually used with real-world, i.e., sampled signals. Events in the signal can be detected by setting a static threshold for the magnitude of the frequency components [24] or by using a dynamic threshold where an autoregressive moving average is used to find the detection threshold from the measurements [25]. More complex alternatives also exist. [26] uses curve-fitting to interpolate the DFT spectrum, which allows for estimating inter-harmonic frequencies; [27] presents a method that helps decouple the SSO component from the supersynchronous oscillation component.

The downside of the FT is its inherent limitations, such as spectral leakage and the picket-fence effect [26]. Spectral leakage is the spreading of the peaks in the frequency spectrum caused by windowing. The Picket-fence effect is caused by the number of points in the frequency domain and hides the actual location of the peaks when the resolution is low.

The Wavelet Transform (WT) comprises scaled and shifted forms of a mother wavelet [8, 22]. Unlike the FT, the WT uses variable-sized windows to analyse and represent data across different scales [28]. Like the FT, it suffers from spectral leakage and the picket-fence effect and is highly influenced by noise [21]. Moreover, the wavelets are predefined and not self-adaptive [29], meaning the choice of wavelet is crucial for accurate estimation.

The S-Transform (ST) is a generalisation of the Short-Time Fourier Transform (STFT) and extends on the continuous WT to overcome some of their difficulties [30]. It can localise the variation in spectral characteristics [29]. However, the ST is computationally expensive compared to other methods making it unsuitable for real-time application [21].

Kalman Filters (KF) are in dynamic systems where an educated guess can be made about the next state of the system. A KF combines the information from the prediction with actual readings to improve the accuracy of the prediction. In power systems, they are therefore used mainly for parameter estimation [21]. [31] provides an excellent visualisation of how a KF works.

The Extended Kalman Filter (EKF) extends the KF to work on non-linear systems. The KF is very fast, making it suitable for real-time applications [21, 31]. The drawback is that the state-space representation of the system is needed, increasing modelling complexity and possibly limiting the number of scenarios that can be evaluated.

Prony Analysis (PA) is a method used to estimate signal contents by fitting a linear combination of exponential terms, each term representing a different oscillatory mode [8]. With PA, the amplitude, damping factor, frequency and phase angle are estimated. A broad description of the method can be found in [8]. A downside of PA is that the noise levels affect the estimation accuracy [23, 26].

Estimation of Signal Parameters via Rotational Invariance Techniques (ESPRIT) is similar to PA in the estimated parameters. The big difference is that ESPRIT uses the autocorrelation matrix of the signal to estimate the coefficients. A more detailed explanation is found in [32]. ESPRIT is better than FT because it also estimates the damping coefficient [32]. Moreover, according to [33], it provides a more accurate estimation of the frequency modes (together with the FFT). On the other hand, ESPRIT has a more significant computational burden because it needs to estimate the self-correlation matrix [26].

In the end, all these methods result in an alternative domain representation of the signal that can be used to distinguish between SSO and non-SSO regions, for example, by monitoring changes in parameters. Alternatively, the computed transforms can be used as features for data-driven methods (see Section 7.2.3).

7.2.2 Data driven methods

Machine learning has undoubtedly proven to be a powerful and versatile approach to problems with many data available. The fundamental difference with signal processing techniques is that the method to distinguish between SSO and non-SSO events is, therefore, also based on trends seen in the data rather than more fundamental (physical) theories. Below, some of the state-of-the-art data-driven approaches are discussed.

In [34], the authors inspected Phasor Measurement Unit (PMU) measurements for ambient and SSO regions. They realised that the amplitude of the phasors shows more prominent fluctuations in the case of SSOs, and a feature is formulated that reflects this variability. A Decision Tree (DT) classifier was used to learn the distinction between SSO and non-SSO regions.

A refinement of this concept is presented in [35]. Instead of one feature, three features are used. One feature is based on the envelope volatility, one on the periodicity of oscillations and one on the presence of stationary subsequences. The authors also opted to use a Support Vector Machine (SVM) classifier instead of a DT.

Both of these approaches work roughly the same. First, some features are derived based on the characteristics of the measurements during SSO events. Second, a machine learning classifier is used for prediction. The advantage of these methods is their simplicity making them interpretable. But this could be a double-edged sword and result in a poor generalisation performance if the approach is too simple.

It is also possible to use an entirely data-driven approach by training Convolutional Neural Networks (CNNs) [36, 37]. The networks can be 1D, in which case the input data will be the measurement time series, or 2D, in which case the input will be an image.

The advantage is that there is no need for specialised signal processing or hardcoded thresholds [36] to learn to detect SSO events. Another advantage is that CNNs are already used in other domains, and efficient implementations exist: they can be run on low-cost devices such as the Raspberry Pi [36] or an Nvidia Jetson [37], making deployment at the grid edge possible.

On the other hand, CNNs are notoriously hard to train. For example, the models proposed in [37] have 96k and 2.2M parameters, respectively; the SVM of [35] has four (one weight per feature plus a bias term). This severely impacts training time. Another issue is that the models depend highly on the training data. This issue is seen in other domains where Google's Vision AI is biased towards certain ethnicities [38, 39]. A model that is trained purely on data and has no (partly) interpretable underlying features derived from the data will fail to generalise when deployed on unseen data. This could result in failing to identify SSO events for data sources it has not seen. This could be different measurement points or PMUs of other manufacturers.

7.2.3 Hybrid approaches

There also exist hybrid approaches. Hybrid approaches combine different signal processing methods that extract features and might use a data-driven method to learn the distinction between SSO and non-SSO regions. [21, 29] are a good reading into different hybrid methods. A couple of interesting ideas are presented below.

A combination of the FFT and Empirical Wavelet Transform (see [40]) is used to decompose a signal into mono-frequency components in [29]. Then, a set of features is derived that characterises different events, and an SVM-based classifier is used to learn the boundaries between different event types. This approach is very similar to the one in [35], except that the features are more complex.

In [32], ESPRIT estimates the measured signal's amplitude, frequency phase and damping factor. Then, three Naïve-Bayes classifiers are trained on data subsets that use a different discretisation method. The Naïve-Bayes classifier learns to distinguish between the events based on the ESPRIT estimations.

[41] uses the ST to detect and localise disturbance, while the EKF was used to estimate the amplitude, frequency and harmonic contents. Simple thresholding is used to identify the type of disturbance in the network from the estimated amplitude.

7.3 Implemented approach

In the previous section, different methods were discussed that are suitable or can be adapted for detecting SSO regions. When choosing a method, the requirements of Section 6.2. can be used for filtering out fewer promising techniques. 1A is not satisfied when using approaches heavily dependent on input data (CNNs) or procedures easily influenced by external factors, e.g., noise (WT, PA). Moreover, some methods are slower (ESPRIT, ST) than others (FFT, KF), violating requirement 6A. Also, not all techniques have interpretable outcomes (requirement 5A).

In the end, it was chosen to use a hybrid approach based on [35]. Features 1 – 3 are created from this method. This approach is attractive because of its simplicity. The used features are derived from signal properties, meaning that the technique should work on different input sources (requirement 1A). Also, simplicity promises better scalability, explainability, and execution speed (requirements 2A, 5A, and 6A). This approach is realisable using standard python libraries, making it accessible (requirement 4A). But, special attention should be paid to ensure that the method performs as expected.

The features should be distinctive enough on their own. Nevertheless, it is chosen to use a machine learning classifier to predict the output. This is because the classifier considers the tree features together, possibly capturing relations between them; a simple threshold considers them individually.

Moreover, an approach based on the FFT is also implemented to identify whether a measurement includes SSOs. This is because the data sources are unlabelled. Feature 4 is derived from the FFT. An advantage is that this method works on a different domain (time domain) than the features from [35] (phasor domain), allowing for an extra check on the input. This should increase the versatility and overall accuracy of the SSO detector.

In the following sections, the working of the features is discussed. First, an overview is given in Section 7.3.1. and the features are described in separate subsections. A validation of the method is presented in Section 7.4.

7.3.1 Overview

A pipeline is needed to extract the features using the chosen methods. An overview is seen in Figure 20. Before computing the features, some extra steps are required.

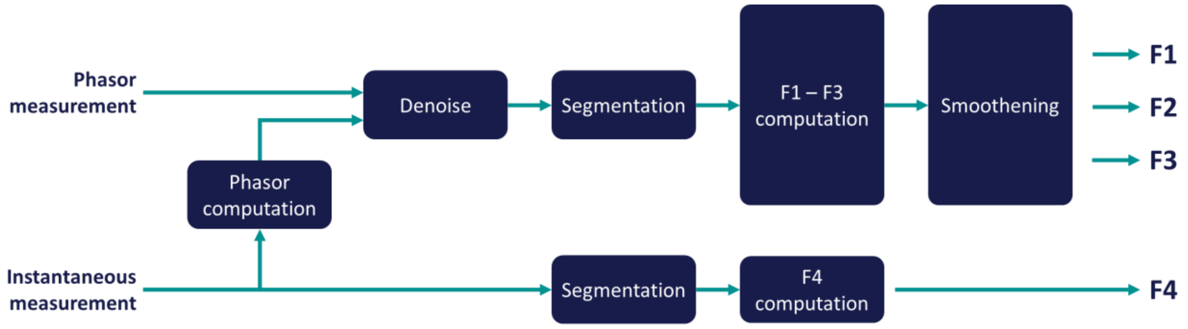


Figure 20 Overview of the feature engineering pipeline

First of all, it is essential to note that features 1 – 3 work on the phasor measurement. Using the phasor measurement allows the removal of the fundamental grid frequency from the signal, meaning that all oscillations are caused by undesired frequency components. This concept will become clearer once the features are introduced in Sections 7.3.2, 7.3.3, 7.3.4 and 7.3.5.

If the phasor measurement is unavailable or has a low sample rate, the phasor can be computed from the instantaneous signal. More details about this are included in Appendix A .

The FFT is computed from the signal's time domain (or instantaneous) measurement. Hence, it is only possible to calculate feature 4 when the time domain measurement is available.

Secondly, the phasor is denoised before computing features 1 – 3. This is done using a uniform filter or, in other words, a moving average filter. These features detect changes in the phasor, and if too much noise is present, they will capture trends caused by noise rather than by SSO events. Averaging the signal reduces this effect. Feature 4 does not require denoising because it considers only significant frequency components. The noise present in the signal will have a lower power density than the oscillatory components and will therefore remain undetected. More details about this are found in Section 7.3.5.

Thirdly, segmentation is performed on the measurements before computing the features. This is done by separating the measurement signal into overlapping segments of a predetermined duration. An example can be seen in Figure 21.

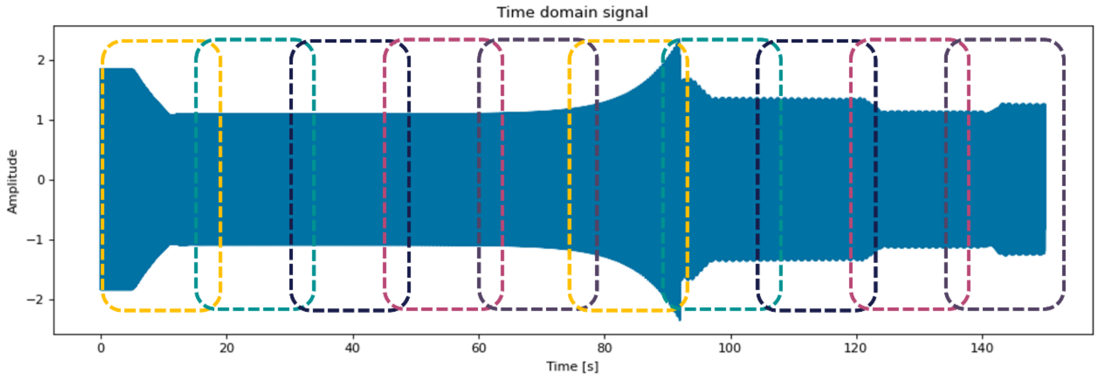


Figure 21 Example of the segmentation of a measurement. Created with data from [20].

There are three advantages to this approach. It reduces the occupied memory size as the features are stored per segment. It could also increase the accuracy as random fluctuations are averaged out when considering a segment, and it allows one to distinguish between different regions in the signal. The default window size is set for 1 second for features 1 – 3 and 4 seconds for feature 4. For the first three features, a shorter duration gives a more granular resolution but can capture trends caused by random fluctuations; a longer period reduces the memory size as the features are computed on fewer windows, but it averages out changes happening in the signal. For example, no change in the feature values would be observed in the SSO region of Figure 19 when only one segment is used. The chosen window size strikes a good balance in having a granular resolution but not capturing random fluctuations, and it is below the window sizes achieved with FT-based methods [27]. There is a trade-off in the FT between frequency resolution and the duration of the time measurement; this gives the segmentation duration of feature 4. With a window size of 4 seconds, a frequency resolution of 1 Hz is obtained with the chosen windowing function. This allows for a reduction of the measurement duration while having a sufficient good frequency resolution. More details about this trade-off are found in Section 7.3.5.

Finally, the output of features 1 – 3 is smoothened with a moving average filter. This produces a smoother transition between the SSO and non-SSO regions. It removes erroneous values of the features, e.g., a single segment giving values indicating SSOs while the surrounding segments predict non-SSO parts.

7.3.2 Feature 1 – Trend

Oscillatory signals are characterised by having a regular variation in magnitude, which periodically repeats itself. Non-SSO data does not exhibit this periodicity [35]. Feature 1 is created with this concept in mind and is appropriately called the trend. This feature decomposes the measurement into regions where the magnitude is constant or increasing and areas where consecutive points decrease in value and looks for a repetition or trend.

This is achieved as follows. Suppose that the signal is denoted by $\boldsymbol{A}$. The rising/constant and falling trend of the point $A[i]$ w.r.t. $A[i-1]$ is given by $B[i]$ as:

$$B[i] = \begin{cases} 1, A[i] \geq A[i-1] \\ -1, A[i] < A[i-1] \end{cases} \tag{1}$$

All the points that are given by (1) form $\boldsymbol{B}$. Next, all consecutive 1 or -1 are counted and stored in $\boldsymbol{C}$. For example, $\boldsymbol{B} = [1,1,1,-1,-1,1,1,1,-1,-1]$ would yield $\boldsymbol{C} = [3,2,3,2]$. After that, it is checked if there exists a sequence in the points in $\boldsymbol{C}$ i.e. a repetition in their values. This is done by using the following condition:

$$\begin{gathered} \exists m \geq 2, m \in \mathbb{N}_+ \\ C[j] = C[m+j], \forall j \ \leq m, j \in \mathbb{N}_+ \end{gathered} \tag{2}$$

All points that satisfy (2) are added to $\boldsymbol{D}$. The value of feature 1 is given by:

$$F1 = \sum_{D[i] \in \boldsymbol{D}} D[i] \tag{3}$$

Let us consider the example from Figure 22. This snippet is taken from the SSO region of the signal in Figure 19. First, $\boldsymbol{B}$ is constructed by finding the points that have a rising amplitude and a falling amplitude. Next, $\boldsymbol{C}$ is obtained by counting the consecutive points that are 1 (rising or constant) and -1 (falling). These are the numbers shown in Figure 22. It is possible to find repetition in $\boldsymbol{C}$ using $m = 2$ in (2): the first four values (11, 12, 11 and 12) satisfy the equation. Hence, these points are added to $\boldsymbol{D}$. Finally, the value of feature 1 is found by summing the points in $\boldsymbol{D}$, giving us $F1 = 46$.

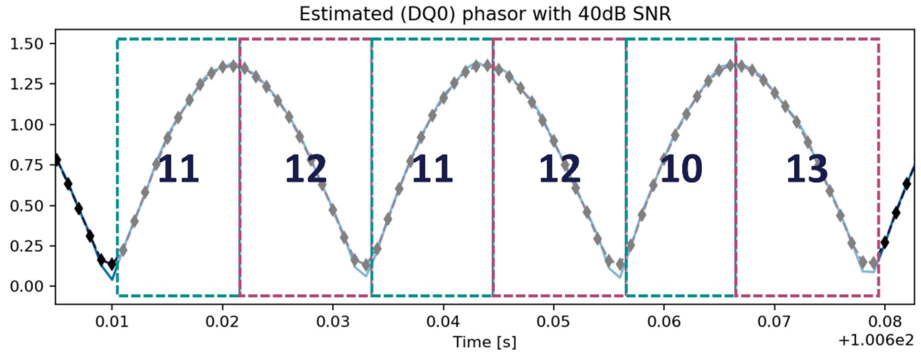


Figure 22 Example of the rising/falling trends found by feature 1 in a phasor signal containing SSO [20].

It is crucial to use the phasor measurement instead of the time measurement. To illustrate this, consider Figure 23, the instantaneous measurement taken from a non-SSO region. Because the current and voltage signals also consist of an oscillatory signal, a clear trend is visible. Feature 1 would have a value of 60 in this case, which is higher than the previously found 46. This would indicate that the segment contains an SSO, which is not the case.

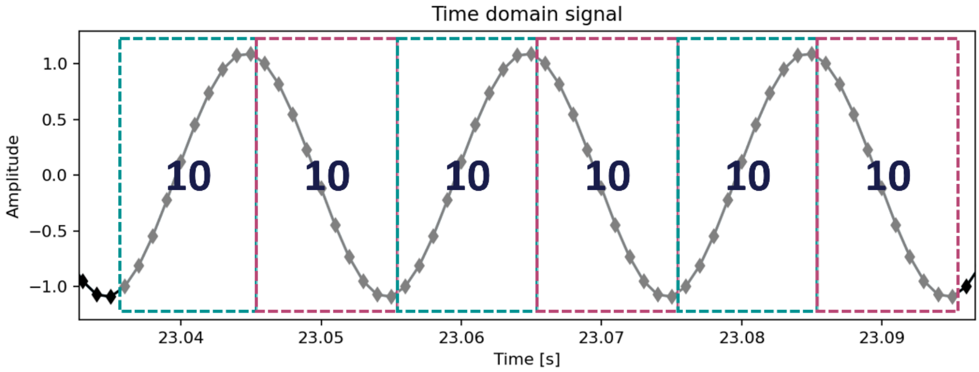


Figure 23 Example of the rising/falling trends found by feature 1 in an instantaneous signal containing no SSO. Created with data from [20].

If feature 1 is computed on the phasor magnitude instead (Figure 24), which has a constant value for a single frequency, there will only be random fluctuations because of noise which show little repetition. This means that the value of feature 1 computed with (3) will be lower than the one calculated in the SSO region. For example, feature 1 on the signal in Figure 24 will have a value of 13, caused by the sequence of 1's and 2's around 23.06 seconds.

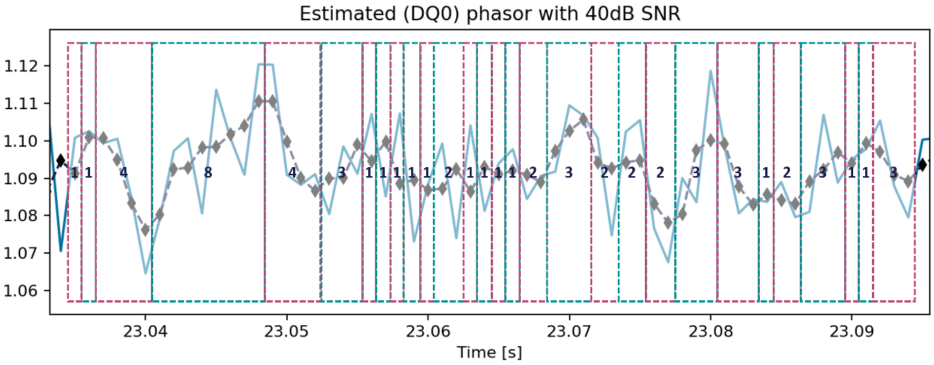


Figure 24 Example of the rising/falling trends found by feature 1 in the phasor signal containing no SSO. Created with data from [20].

Although the original method of [35] works well on synthetic signals, some improvements are needed to handle real-world data. These improvements involve the sample rate of the phasor and the amount of noise present in the signal.

Firstly, random noise fluctuations could result in $1 \leftrightarrow -1$ (or other low-valued) sequences. Note that an SSO would unlikely cause this as the change in its phasor magnitude has a well-defined and periodic form. Thus, segments that should have a zero value for feature 1 will get a non-zero value. This is also seen in Figure 24. It could become an issue for sources where the value of feature 1 during an SSO is also low. This can be solved by checking for sequences with (2) only for points that are above $\varepsilon_{c-significant}$. The threshold is given by the values of $C[i]$ in (non-)SSO regions.

Secondly, the non-significant $1 \leftrightarrow -1$ (or other low-valued) sequences are most likely to occur around maxima and minima in the phasor. In those regions, the change between two consecutive points becomes smaller. If the changes caused by noise become more significant than the change caused by the signal, the sequence in $\boldsymbol{C}$ is interrupted by insignificant $1 \leftrightarrow -1$ sequences. This will result in a lower value of feature 1, possibly failing to detect SSO events. A peak detection and sequence-removing algorithm are created to fix this issue. Using the derivative of the phasor, regions that could contain the maxima and minima are localised; these are the regions for which the derivative is sufficiently close to 0, i.e., where $\nabla A[i] \leq \varepsilon_{gradient}$. To prevent considering non-SSO areas (where the phasor is flat with noise fluctuations), only maxima and minima with values higher than $\varepsilon_{c-significant}$ are considered.

If an insignificant $1 \leftrightarrow -1$ sequence is found around two significant maxima/minima, the sequence is removed, and the number of deleted points is split equally between the nearby maxima/minima. If the number of removed points is odd, one sample is subtracted to obtain an even number. For example, if the considered region has $\boldsymbol{C} = [1, 2, -1, 1, -1, 1, \dots, 23, 1, -1, 1, 32, 24, \dots]$ and $\varepsilon_{c-significant} = 6$, the algorithm would update C as $[1, 2, -1, 1, -1, 1, \dots, \mathbf{24}, \mathbf{33}, 24, \dots]$. The sequence at the start is unaltered as it is below the significance threshold.

Finally, the original method only works well when the upward and downward trends repeat after the same number of sample points. If this is not the case, i.e., when the period of the SSO is not an integer multiple of the sampling period, there will be a slight variation in the counted points. This is seen in Figure 22, where the last values have changed by one point per trend. This issue is likely to occur in real-life data as the sampling is not perfectly aligned with the period of the SSO. To fix this issue, the equality constraint of (2) is relaxed to:

$$|C[j] - C[m + j]| \leq \varepsilon_{c-similar} \tag{4}$$

The relaxation in (4) allows for the number of consecutive points in $\boldsymbol{C}$ to deviate slightly.

7.3.3 Feature 2 – Envelope Volatility Index (EVI)

The second characteristic of oscillatory signals is that the magnitude change is constant within the repetition period of the SSO. If the amplitude of the SSO were to decrease, the perturbation would not be perceived as it is well-damped. If there is insufficient damping, an oscillation will be the result.

Feature 2 is called the EVI, as it looks at the variations in the signal's envelope. This is done using the following approach [35]. Again, suppose that the measurement is denoted by $\boldsymbol{A}$. First, all peaks and valleys of a signal segment are detected –

$$\begin{aligned}\boldsymbol{B_2} &= \{A[i] \mid A[i] \geq A[i-1], A[i] \geq A[i+1], \forall i\} \\ \boldsymbol{C_2} &= \{A[i] \mid A[i] \leq A[i-1], A[i] \leq A[i+1], \forall i\}\end{aligned} \tag{5}$$

Then, feature 2 is computed using the following equation:

$$F2 = \sqrt{\frac{var(\boldsymbol{B_2})}{var(\boldsymbol{A})} \cdot len(\boldsymbol{B_2})} + \sqrt{\frac{var(\boldsymbol{C_2})}{var(\boldsymbol{A})} \cdot len(\boldsymbol{C_2})} \tag{6}$$

Equation (6) can be interpreted quite intuitively. If the variance in the found peaks and valleys is high, and when there are more peaks and valleys, the value of feature 2 will also be increased. In the case of an SSO signal, the phasor signal will be periodic and have a well-defined envelope. This means there is little variance in the peaks and valleys and relatively few peaks and valleys, resulting in a low value of feature 2. For non-SSO signals, the phasor magnitude should be flat. Any found peaks and valleys will be caused by either noise in the signal or other interferences. In [35], it was found that non-SSO measurements have more peaks and valleys than their SSO counterpart and that they show a more significant variance in their envelope, resulting in a higher value of feature 2. In (6), the variance of the peaks and valleys is divided by the data variance, meaning the relative variation of the envelope is used. In non-SSO regions, the absolute variations caused by noise are minor when compared to the SSO region. However, the relative deviations in the envelope are significant, while for SSO regions, the deviations are negligible.

Let us consider the same example. In Figure 25, the 80 ms phasor measurement contains an SSO. In this segment, there are only three peaks and four valleys, which have a slight deviation in their amplitude. Because the variance in the measurement is large, the terms $\frac{var(\boldsymbol{B_2})}{var(\boldsymbol{A})}$ and $\frac{var(\boldsymbol{C_2})}{var(\boldsymbol{A})}$ of (6) will be small. Also, $len(\boldsymbol{B_2})$ and $len(\boldsymbol{C_2})$ are 3 and 4 respectively. This results in a small value of feature 2.

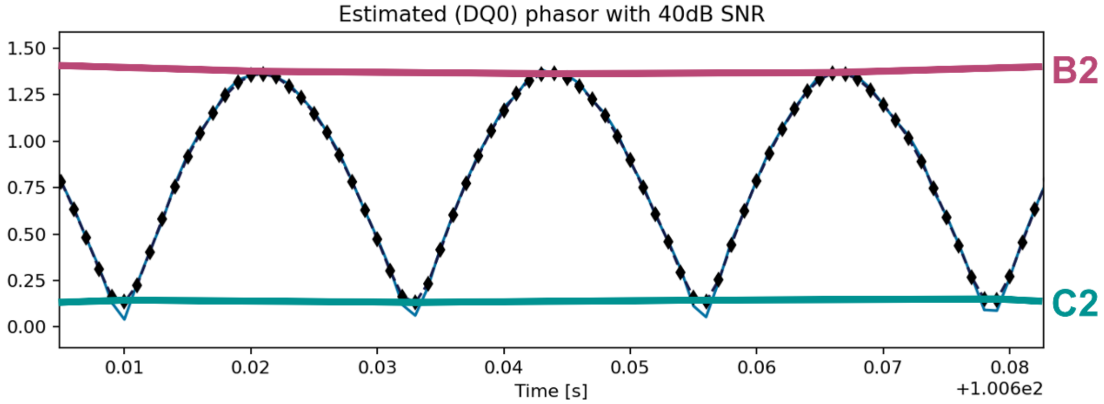


Figure 25 The envelope of a phasor signal containing an SSO. Created with data from [20].

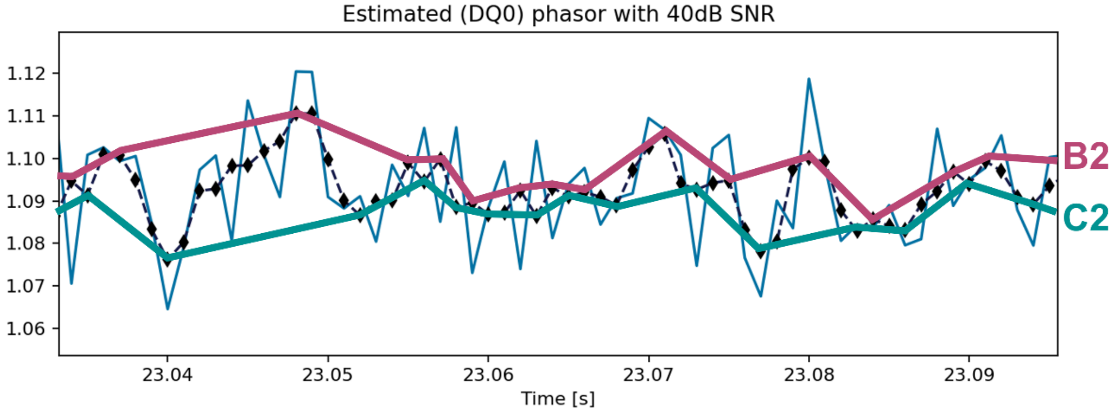


Figure 26 The envelope of phasor signal containing no SSO. Created with data from [20].

In Figure 26, there are many peaks and valleys: $len(\boldsymbol{B_2})$ and $len(\boldsymbol{C_2})$ are 15 and 16 respectively. Also, because the range of the data is much smaller, $var(\boldsymbol{A})$ is also small. This results in larger relative deviations in the values of the peaks and valleys. Hence, the terms $\frac{var(\boldsymbol{B_2})}{var(\boldsymbol{A})}$ and $\frac{var(\boldsymbol{C_2})}{var(\boldsymbol{A})}$ are significant. This means that feature 2 will have a considerable value.

One adaptation must be made for feature 2 to work correctly on real-world data. Depending on the size chosen for segmentation, one of the variances may not be defined. In Figure 27, the segment's duration is too short to find any peaks or valleys. This means that the variances $var(\boldsymbol{A})$, $var(\boldsymbol{B_2})$ and $var(\boldsymbol{C_2})$ are undefined (NaN).

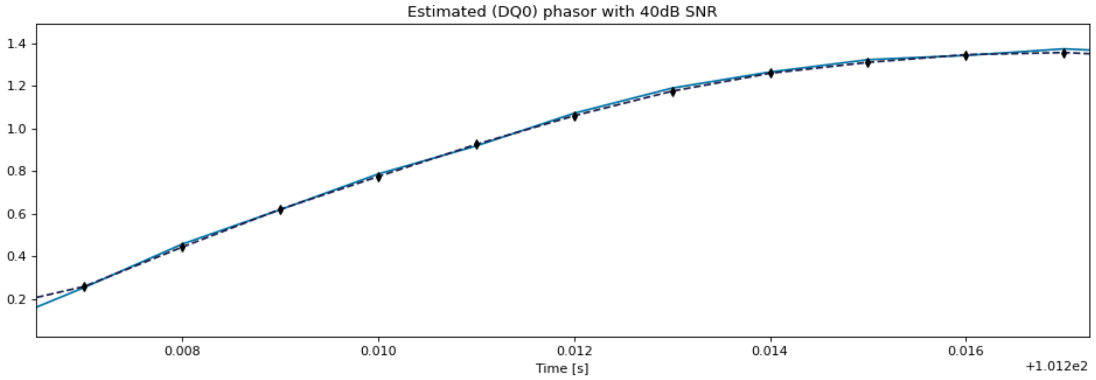


Figure 27 A too-small segmentation of the phasor measurement results in no peaks and valleys. Created with data from [20].

Because the expected signal properties during (non-)SSO regions are known, it is possible to evaluate the different scenarios and infer the value of feature 2:

1. If $var(\boldsymbol{A}) = 0$, it means that the phasor magnitude is flat. This is expected to happen only during non-SSO regions because the estimated phasor aligns with the (presumed) grid frequency. Therefore, feature 2 gets a low value ($F2 = 0$) when $var(\boldsymbol{A}) = 0$.
2. If $var(\boldsymbol{B_2}) = 0$ and/or $var(\boldsymbol{C_2}) = 0$ and $var(\boldsymbol{A}) \neq 0$, feature 2 is computed with (6).
3. If $var(\boldsymbol{B_2}) = \text{NaN}$ and/or $var(\boldsymbol{C_2}) = \text{NaN}$ and $var(\boldsymbol{A}) \neq 0$, a NaN is returned. The variances will not be defined if there are no points in $\boldsymbol{A}$ that satisfy (5) i.e., no minima or maxima. This is the case in regions that are monotonously increasing or decreasing, such as in Figure 27. Because it is unknown if this increase is due to oscillations in the phasor magnitude or due to noise, the expected value of feature 2 is unknown. Instead, an error is raised to use longer segments.

In most cases, scenario two will hold, meaning that feature 2 is computed normally.

7.3.4 Feature 3 - Length of Stationary Subsequences (LSS)

The third feature is the length of stationary subsequences, which looks at how much the adjacent amplitudes of the phasor change. During SSO events, the amplitude changes significantly, while for non-SSO measurements, there are more stationary subsequences [35].

Feature 3 is computed as follows. First, the set $\boldsymbol{B_3}$ is constructed –

$$\boldsymbol{B_3} = \left\{ A[i] \mid \frac{|A[i] - A[i-1]|}{A[i]} \leq \varepsilon_{LSS}, \forall i > 0 \right\} \tag{7}$$

Then, feature 3 is found as –

$$F3 = len(\boldsymbol{B_3}) \tag{8}$$

In (8), the term $\frac{|A[i]-A[i-1]|}{A[i]}$ can be interpreted as the difference normalised by the current sample value $A[i]$. For signals in the SSO region, this relative change will be more significant since it is caused by the oscillatory trend in the signal. This means that there will be fewer points below the threshold, and thus feature 3 will have a lower value. In non-SSO measurements, the noise will cause slight changes in the phasor magnitude, resulting in a more negligible difference. The result is that more points are in $\boldsymbol{B_3}$ giving a higher value for feature 3.

The threshold ε_{LSS} is highly dependent on two factors, namely, the sample rate and range of the data. The same simulation measured at a higher sampling rate will give points closer to each other, meaning that the threshold is met more easily. Also, if a source has a smaller data range than another, the threshold is reached more often since the measurement points are closer. Thus, it is essential to choose ε_{LSS} appropriately for each source.

7.3.5 Feature 4 – Fourier Transform Index (FTI)

The Fourier Transform (FT) is used to determine the frequencies in a signal. The FT computed on a time-domain signal gives its frequency spectrum. A peak at each present frequency in this spectrum will indicate its Power Spectral Density (PSD). The higher the peak, the more significant that frequency component is in the signal. This means that measurements with SSOs will show peaks not caused by the fundamental grid frequency.

Feature 4, the Fourier transform index, is created with this relationship in mind. The FTI is an index computed based on the periodogram of a signal:

$$F4 = \sum_{f_i \in F_{sig}} \frac{PSD(f_i)}{PSD(f_{grid})} \tag{9}$$

In (9), F_{sig} represents the set containing all significant frequencies without the grid frequency $(f_{grid} \notin F_{sig})$.

For signals without SSOs, there will be only one peak at f_{grid}. In that case, feature 4 will have a value of 0 because there are no peaks to sum. When a signal does contain SSOs, $F4 \neq 0$. The higher the value of feature 4, the higher the PSDs of the frequencies or the more frequencies there are. A value of 1 means that all significant frequencies have as much energy in the signal as the fundamental frequency.

The FT suffers from spectral leakage. In the ideal spectrum, each frequency would be represented by a Dirac delta function at each frequency, and more dominant frequencies would have a higher impulse. However, the Discrete Fourier Transform (DFT) or, more specifically, the efficient Fast Fourier Transform (FFT) is used when working with sampled signals. The FFT implicitly assumes that the measured signal repeats itself outside the measured interval [42], making the signal periodic. The FT of a periodic function gives a discrete spectrum, meaning that the computed spectrum is only defined at certain frequencies [43]. Those frequencies are given by:

$$f_{bins} = \left\{ k \frac{F_s}{N_{samples}} \middle| 0 \leq k \leq N - 1 \right\} \tag{10}$$

In (10), $N_{samples}$ is the number of samples of the input signal and is alternatively represented by $\frac{t_{segment}}{T_s}$. Moreover, the spacing between the frequencies is given by

$$\Delta f = \frac{F_s}{N_{samples}} = \frac{1}{t_{segment}} \tag{11}$$

(11) tells us that the longer the segment, the closer the estimated frequencies can be in the FFT; it also tells the lowest frequency. Intuitively, this makes sense as a signal of 1 s cannot estimate a frequency of 0.5 Hz; the period of that frequency is 2 seconds, after all.

One of the frequencies may not be directly represented by a frequency bin. In that case, the power at that frequency will be spread out over the neighbouring bins (see the experiments in [42-44]). The frequency energy is leaked across the spectrum, i.e., spectral leakage.

This effect is most dominant when the recorded measurement does not contain an integer multiple of the oscillation period (e.g., $t_{segment} = 1.5\, T_{SSO}$). For example, this could happen when segmenting the measurement. Because the FFT implicitly assumes that this segment is repeated, a sharp jump is present in the signal. This indicates that different frequency components are present in the signal apart from the SSO and fundamental frequencies, causing spectral leakage.

A windowing function is applied to reduce this effect. [45] presents an overview of window functions and their properties. A windowing function is 0 at the start and end, smoothening the sharp jumps. Depending on the used windowing function, the spectral leakage to nearby frequencies is reduced compared to not using any windowing function (also known as the rectangular window).

Another effect that can occur is the picket fence effect caused by spectral leakage. As seen in (10) earlier, the FFT is only well-defined at specific frequencies. This means that the discrete frequencies obtained in the frequency domain could hide the actual location of the peak and, thus, the frequency.

This is like looking at the frequency spectrum through the cracks of a picket fence, hiding parts of it. A solution is to zero-pad the signal, i.e., add trailing zeros to the measurement. The effect is that the frequency spectrum is interpolated between the original bins, allowing for better estimation of the actual frequency (see also the demos in [43, 44]).

It is crucial to realise that zero-padding the signal does not increase the frequency resolution of (11). Instead, it interpolates the existing frequencies of the FFT [43]. The measurement must be recorded for a longer duration to increase the frequency resolution, giving more frequencies in the FFT where the PSDs are computed [43, 44].(10)

The exact location of the peaks is not so important for detecting SSOs. Nevertheless, it is essential to obtain an accurate spectrum to avoid considering unwanted frequencies. Because only the extremes are detected in (9), they have to be accurate. Luckily, the spectral properties of the windowing functions are known [45]. The preferred window is the Hamming window (see Figure 28), as it has a narrower main lobe width (MLW) compared to the Blackman window, and its side lobe height (SLH) is lower than that of the rectangular window. The former is crucial to allow for the estimation of multiple close-by frequencies. The latter is required to reduce the spectral leakage to nearby frequencies.

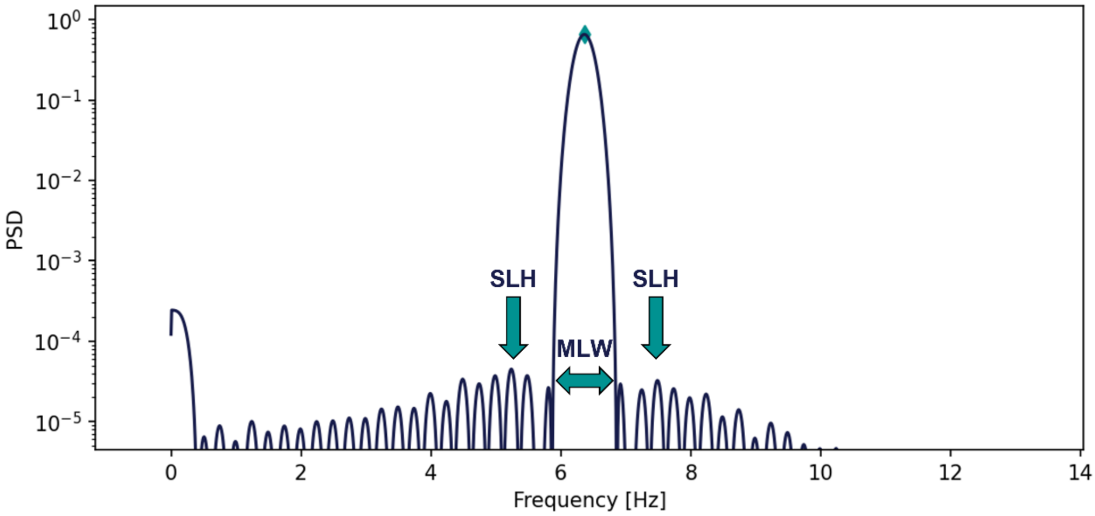


Figure 28 The spectral properties of the Hamming window (MLW and SLH). Created with data from [20].

The suggested segment durations are 10 seconds to have a sufficient frequency resolution [27]. However, SSOs hardly remain unchanged within a segment of this duration. For example, the SSO could be damped, resulting in a lower peak. This is because the PSD is computed over the entire segment, yielding the average PSD of the particular frequency. Also, the frequency could shift slightly, creating different peaks and effectively spreading the energy over the spectrum.

A shorter segment duration is used. This duration is chosen based on the desired spacing between two frequencies $f_{min\text{-}res}$. By ensuring that the MLW is less or equal to this minimum frequency resolution, it will become possible to find two peaks close to each other. This is because the main lobes of the two frequencies will not overlap; only the side lobes will interfere, which have a negligible contribution (-43 dB for the Hamming window [45]).

The MLW is known for the window:

$$MLW = \frac{C_{window}}{N_{window}} \tag{12}$$

In (12), C_{window} is a constant specific to the window type and can be found in [45]. The MLW is given in normalised (angular) frequency. This can be converted into Hz as:

$$MLW_{Hz} = MLW \cdot \frac{F_s}{2\pi} = \frac{C_{window}}{2\pi \cdot N_{window} \cdot T_s} \tag{13}$$

The following boundary should be enforced to achieve the desired frequency resolution:

$$f_{min-res} \geq MLW_{Hz} \tag{14}$$

To figure out the desired segment duration, (13) and (14) are combined:

$$\frac{C_{window}}{2\pi \cdot f_{min-res}} \leq N_{window} \cdot T_s = t_{segment} \tag{15}$$

For example, if a frequency resolution of 1 Hz is desired, the minimum duration of the segment is 4 seconds when using a Hamming window. This is confirmed by Figure 28, where a segment duration of 4 seconds was used. This frequency resolution allows using smaller segments, ensuring that feature 4 is not averaging measurements too much. However, a relatively long duration is required to distinguish between the different frequencies. This is because the SSO frequency range is limited (5 – 45 Hz [7]), and the large frequency resolution obtained when using short durations (e.g. $\Delta f = 5\text{ Hz}$) is quickly too coarse for this application.

In Figure 29, a periodogram computed from the SSO region of the measurement in Figure 19 is shown. It is possible to identify two peaks, one belonging to the grid frequency and one caused by the SSO. To find the significant frequencies F_{sig} in (9), the following steps are performed:

1. First, the grid frequency f_{grid} is identified from the signal. This step is required since the grid frequency is never precisely 50 Hz but is allowed to deviate by ± 0.5 Hz [46].
2. The frequency range $f_{grid} \pm \frac{f_{min-res}}{2}$ is blocked from finding any peaks.
3. The significant peaks are localised within the specified frequency range. In Figure 29, different thresholds are shown that can be used. The default threshold is found as:
$$\varepsilon_{f-significant} = \max\left(\mu_{PSD} + 3 \cdot \sigma_{PSD}, 10^{\frac{SLH+SLM}{10}}\right) \tag{16}$$
The first term is used to ignore the noise floor but instead take only significant frequencies, making feature 4 robust against different sources with varying noise characteristics.
The second term is to ensure that the found peak is not caused by spectral leakage of the windowing function. A safety margin (SLM) of 5 dB is added to the SLH to create a sufficient distinction between the side lobes and frequencies with low energy content.
Note: the significant frequencies found are spaced by at least $f_{min-res}$.

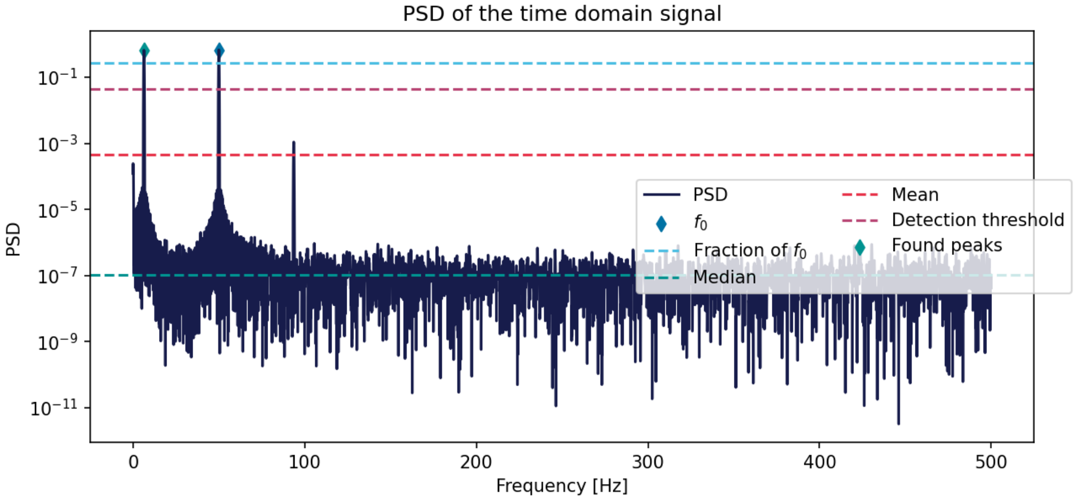


Figure 29 The periodogram of the example measurement segmented between 115.5 – 119.5 seconds. Different thresholds for the significant frequencies are shown. Created with data from [20].

The measurement segment of Figure 29 will give a value of $F4 \approx 1$ as there is one peak found with a PSD of approximately that of f_{grid}. Thus, feature 4 successfully indicates that this segment contains an SSO event.

7.4 Validation

The features will be validated by using measured signals that contain SSOs. No real PMU data was available to test the features. Luckily, one IEEE dataset was found with three (simulated) SSO events [20]. This is not enough to properly validate the features and train a classifier. Hence, two other data sources are used to generate data: a proprietary SSO generator and a CIGRE testbench model.

In real-world measurements, noise is present. Also, system events such as load steps or tap changes change voltage and current levels. The pipeline of Figure 20 is extended to simulate these events and make the generated data more representative. This is done by adding perturbation sources, as seen in Figure 30.

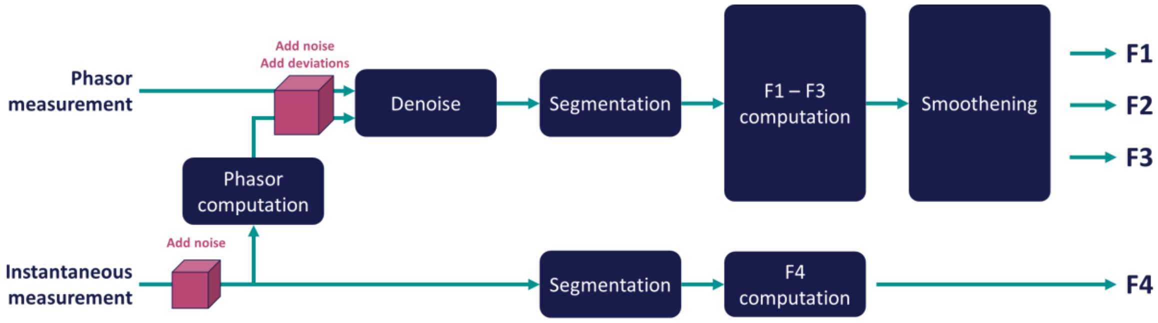


Figure 30 The feature engineering pipeline extended with perturbation sources.

The remainder of this section is structured as follows. First, the theoretical behaviour is presented. Then, each dataset is discussed in more detail, and the computed features are shown. For more information about tuning the features, see Appendix B.

7.4.1 Theoretical behaviour

The four features have interpretable behaviour, allowing us to reason how they will behave under different signals. Three types of oscillations are considered – well damped, poorly damped and increasing amplitude.

Well damped oscillations are those which have an amplitude that decreases quickly to 0. This means that they will be damped quite fast and will, therefore, not cause any harm to system components. These oscillations are considered of the non-SSO type. Feature 1 will have a low value because the phasor is constant. Features 2 and 3 will have high values as the signal has a volatile envelope (noise dominates the phasor) and since the relative change between samples is small. Feature 4 will be 0 as the energy contents of the oscillations are small due to their short duration.

Poorly damped oscillations are the opposite and will persist in the system for longer. These are also the oscillations that the features would detect since they are present for a significant duration in the measurement. The phasor would oscillate as an SSO superimposed on the grid frequency, resulting in a high value of feature 1 and low values of features 2 and 3. Moreover, feature 4 would be non-zero as there would be a significant peak in the periodogram caused by the SSO.

Oscillations with increasing amplitude (undamped oscillations) remain in the system until something disconnects or breaks. Feature 1 would detect a trend even when the amplitude increases since the period of oscillation remains the same, meaning the number of sample points in the upward and downward regions stays the same. Features 2, 3 and 4 will work the same as for the poorly damped oscillations.

7.4.2 IEEE Dataset

The dataset from IEEE [20] contains three measurements with SSOs. The first measurement is used throughout this report to illustrate the working of the features. It is a simulated system of a Doubly-Fed Induction Generator connected to a parallel compensator and then coupled through a weak transmission line with the grid.

The second measurement is a simulation of the actual Hami power grid in Northwest China with a high wind power penetration. The third dataset is PMU data recorded during an actual SSO event in the North China power grid.

The features were calculated with the default segment duration of 1 second for features 1 – 3 and a minimum frequency resolution of 1 Hz (the equivalent of 4-second segments). These are also the default values of the features. Moreover, the threshold for feature 3, ε_{LSS}, is set to 0.0035. Also, some small steps are added to the phasor.

The results of the feature computation are seen in Figure 31. From the figure, there is a clear transition between the non-SSO (0 – 70 seconds) and SSO (70 – 150 seconds) regions. This means that for this measurement, the features are distinctive enough.

It is worth noting that the exact moment of transition differs slightly between the features. This is not an issue, as the goal is to identify SSOs in measurements from a scenario and not to detect the exact moment when the SSO occurs. From the computed features, it is still possible to conclude that a significant portion of the measured signal contains undesired SSOs.

Furthermore, because the time domain measurement is three-phase, feature 4 is computed on each phase individually. This is done to detect SSOs even if they occur in a single phase.

Computing the features on the other measurements also shows promising results. However, these measurements do not contain a precise moment where the SSOs are excited, making it more difficult to validate the features. In that case, it is checked whether the features produce the correct range of values: features 1 and 4 should be high and features 2 and 3 should be low. This is indeed the case; hence, the features work well on this dataset.

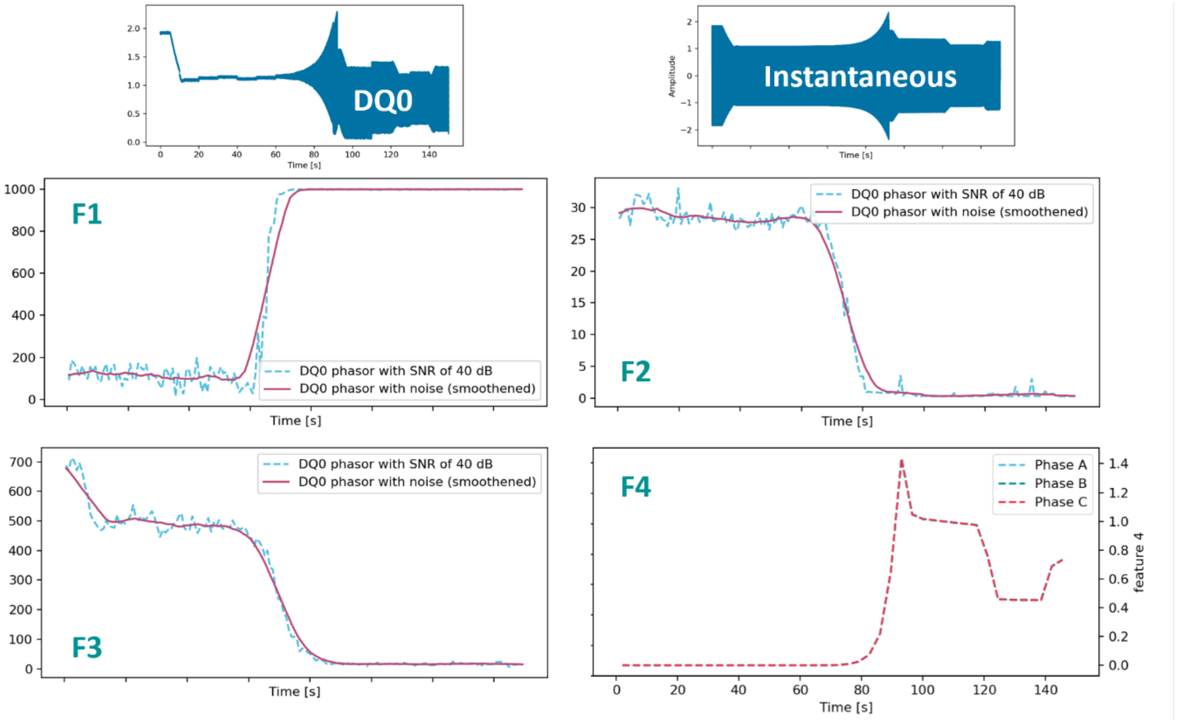


Figure 31 Features computed on the first dataset of [20].

7.4.3 SSO generator

The SSO generator model consists of three three-phase sinusoidal generators added together and passed through a PMU measurement block. Three quantities can be specified for each generator: frequency, amplitude, and phase. One generator will serve as the grid source, while the other two will be switched on to add oscillations. The block diagram is depicted in Figure 32. This model will be used to generate artificial measurements to test the features. The significant advantage is that the model allows precise control of the signal contents meaning that the measurements are well-defined. A drawback is that it lacks complex system dynamics, making it less representative.

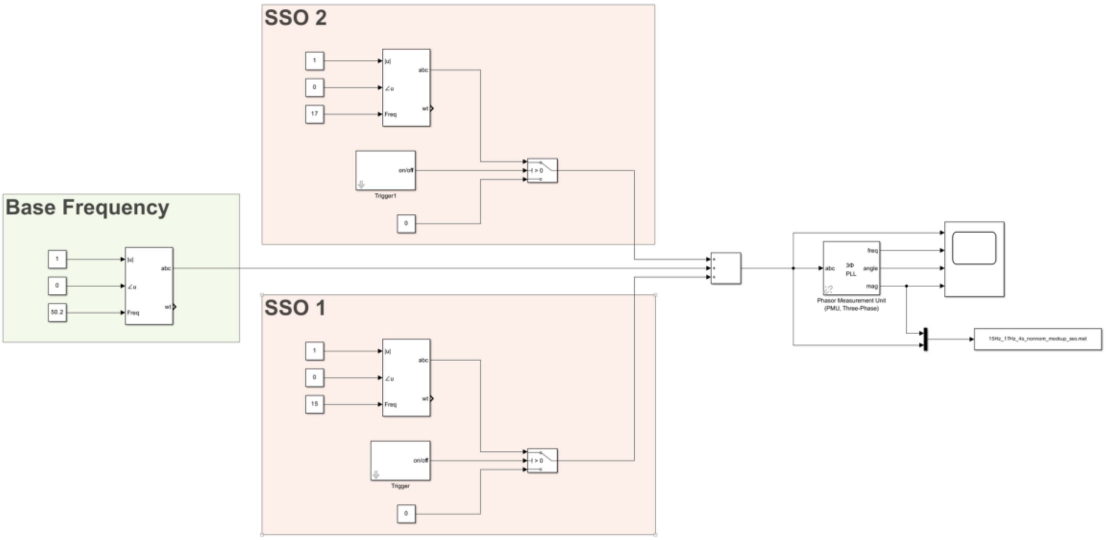


Figure 32 Schematic of the proprietary SSO generator.

The *Base Frequency* block represents the grid. It creates a constant three-phase voltage of frequency f_{grid}. Ideally, the grid frequency is around 50 Hz, but minor deviations occur. Because the detector should work all the time, we have to include these frequencies as well. Deviations between 49.2 – 50.5 Hz are acceptable [47]; larger variations are harmful to the grid and mitigated as much as possible. This means that the most often occurring grid frequencies lie within these limits, meaning it is possible to sample grid frequencies representative of the real system.

National Grid makes the measurements of their grid frequency openly available on their website. The measurements from the past year are downloaded to account for seasonal variations, and a probability distribution is fit. Because there is a dip around 50 Hz (never a perfect frequency), the normal distribution cannot be used. A double gaussian distribution was tried, but the implemented optimisation problem was not good. In the end, a normal kernel distribution is fitted to the data, giving the distribution in Figure 33, making it possible to sample realistic values of f_{grid}.

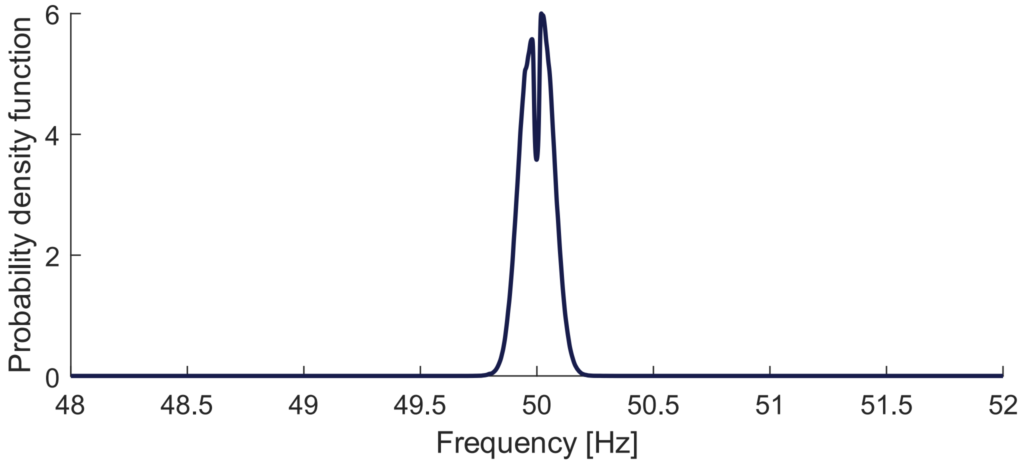


Figure 33 Distribution of the grid frequencies of National Grid between October 2021 and September 2022. Fitted using data from [46].

The blocks *SSO 1* and *SSO 2* consist of switched voltage sources which simulate the occurrence of SSOs. Three parameters need to be specified: the frequency, the amplitude and the phase. The frequency is determined by sampling a uniform distribution between 1 and 48 Hz, representing the SSO range; a more accurate range would be from 5 – 45 Hz [7], but if the features work well on this superset, they should also perform as desired in the actual range. An extra check is added to ensure that the frequencies of the SSO sources are at least $f_{min-res}$ apart. The exact amplitude of SSO events is unknown because few events are recorded. SSOs could have amplitudes that are smaller, similar or larger than that of the grid, depending on their cause. The amplitude is sampled from a uniform distribution between 0.01 (power of -40 dB) and 2 (power of 6 dB) to ensure that the features work robustly under a wide range of scenarios. The amplitude of the second SSO source is only non-zero in case of double SSO events. It is presumed that these occur less often than a single SSO. A probability limits these events; the double SSO event occurs with a chance of 0.1. The phase is set to 0.

The exact timing of the SSO event is not important because the features work independently on the multiple measurement segments. The event start time and duration are also sampled randomly from a uniform distribution, mimicking the stochasticity of the occurrence of SSO events. Since feature 4 requires the SSOs to have a duration of at least one period, the minimum duration is three seconds (three periods of a 1 Hz signal).

The features were calculated with the default segment duration of 1 second for features 1 – 3 and a minimum frequency resolution of 1 Hz (the equivalent of 4-second segments). These are also the default values of the features. Moreover, the threshold for feature 3, ε_{LSS}, is set to 0.0035. Noise and deviations are added with the sources (Figure 30) to make the simulations more realistic.

The computed features on one generated SSO are seen in Figure 34. Features 1 and 4 peak during the SSO event, while features 2 and 3 dip, as expected. Similar behaviour is seen for the other simulated measurements, validating that the features work for this data source.

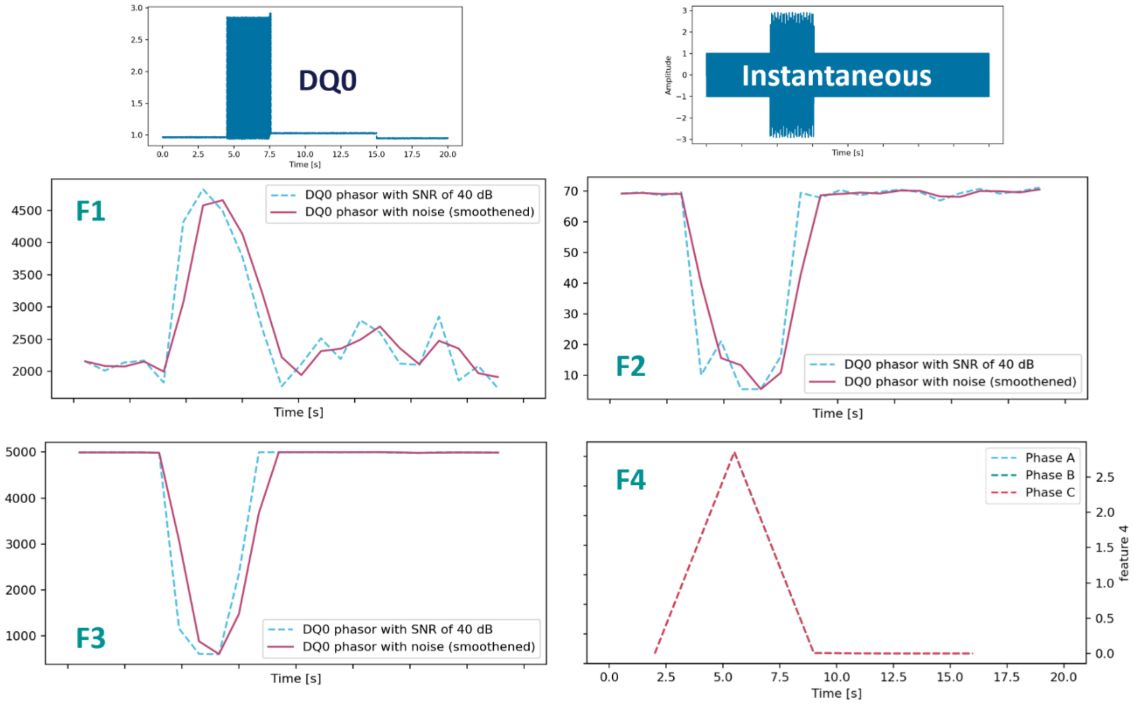


Figure 34 Features computed on signal generated SSO

7.4.4 CIGRE test bench

Dynamic stability is a crucial topic to keep power grids reliable. CIGRE working group on *Multi-frequency stability of converter-based modern power systems* is working on this topic. The network model used in this section is based on the Cigre benchmark system developed by the Cigre WG.

In [11, 48], various methods are described that can identify and possibly mitigate frequency instabilities in converter-based power systems. A test bench was created to investigate and validate the effect of the proposed methods. The system is depicted in Figure 35. It consists of two identical Wind Turbine Generator (WTG) strings, and inverters connect both to the grid. There are several cable segments which can cause interactions with the PE. The grid is modelled as a voltage source with a grid impedance.

The testbed is at its *"stability boundary, i.e. the system is poorly damped, and even small parameter variations can lead to instability"* [48]. This is also seen from the eigenvalue analysis, showing multiple undamped oscillatory modes. Note that the modes are not exclusively from the SSO range but also include interactions in the harmonic range.

A more extensive analysis of the CIGRE test bench is included in Appendix C.

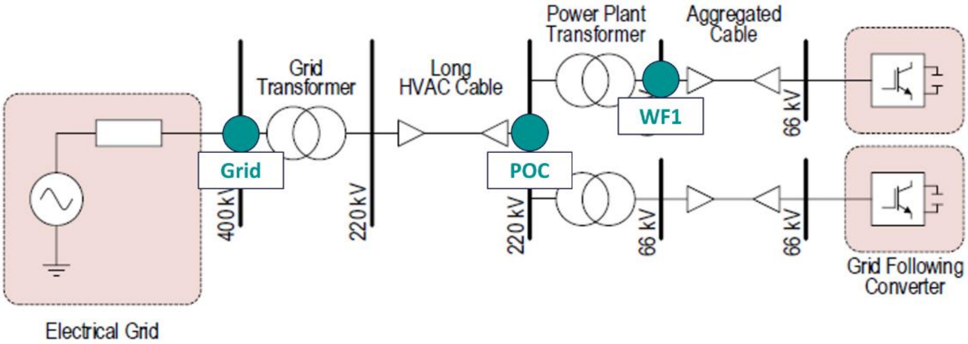


Figure 35 Simple representation of the CIGRE test bench. Image from [48].

The authors list the grid's Short-Circuit Ratio (SCR) as the main contributor to the SSO. The damping of the DC voltage control loops in the aggregated WTG strings and the Phase Lock Loop (PLL) decreases with the SCR and becomes unstable below $S_n \leq 1100\ \mathrm{MVA}$ and $S_n \leq 750\ \mathrm{MVA}$ respectively [48]. This means that by varying S_n, different scenarios with oscillations can be created.

Steps are applied in the grid voltage, AC and DC voltage references of the inverters, and the DC side's power to excite these oscillations. Measurements are recorded at three points (see Figure 35): one WTG string (WF1), the point of coupling of the strings (POC) and the grid side (Grid). Moreover, both the current and voltage are measured. This is done to test the features on different signals at different points in the system where the SSOs might be more or less prominent. This means that six measurements are recorded per scenario.

Unlike the other data sources, it is unclear what frequencies are present in the CIGRE test bench. This complicates the feature validation. A spectrogram is computed on each measurement to analyse the measurement’s content. The spectrogram is calculated using Welch’s method for frequency bins of 5 Hz, allowing a time resolution of 0.2 seconds. This spectral resolution suffices as only the occurrence of oscillations is of interest, not the exact frequencies.

Note that MATLAB’s spectral analyser uses an exponential averaging method, meaning that the frequencies are spread out a bit in time; with a forgetting factor of 0.2, past measurements contribute 17% to the current value. For more details about the spectrum analyser, see [49].

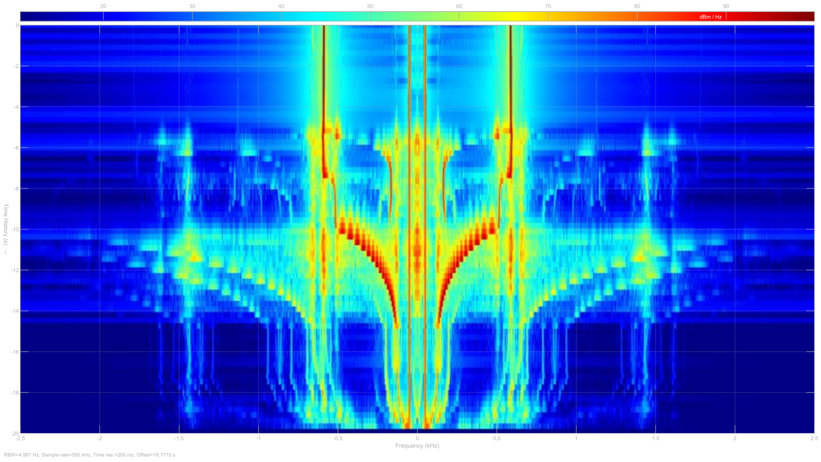


Figure 36 Spectrogram of the current at WF1 for S_n=800 MVA

In Figure 36, two continuous vertical lines can be seen around 50 Hz, corresponding to the grid frequency. After 0.2 seconds, the first event kicks in, causing oscillations in both the SSO and harmonic range. Several other events occur, causing the oscillatory frequencies to shift (e.g., at 5 seconds). From such a spectrogram, it can be concluded that the entire scenario contains oscillations.

The features were calculated with the default segment duration of 1 second for features 1 – 3 and a minimum frequency resolution of 1 Hz (the equivalent of 4-second segments). Moreover, the threshold for feature 3, ε_{LSS}, is set to 0.002. Only noise is added because the deviations are already present in the system, caused by the various events.

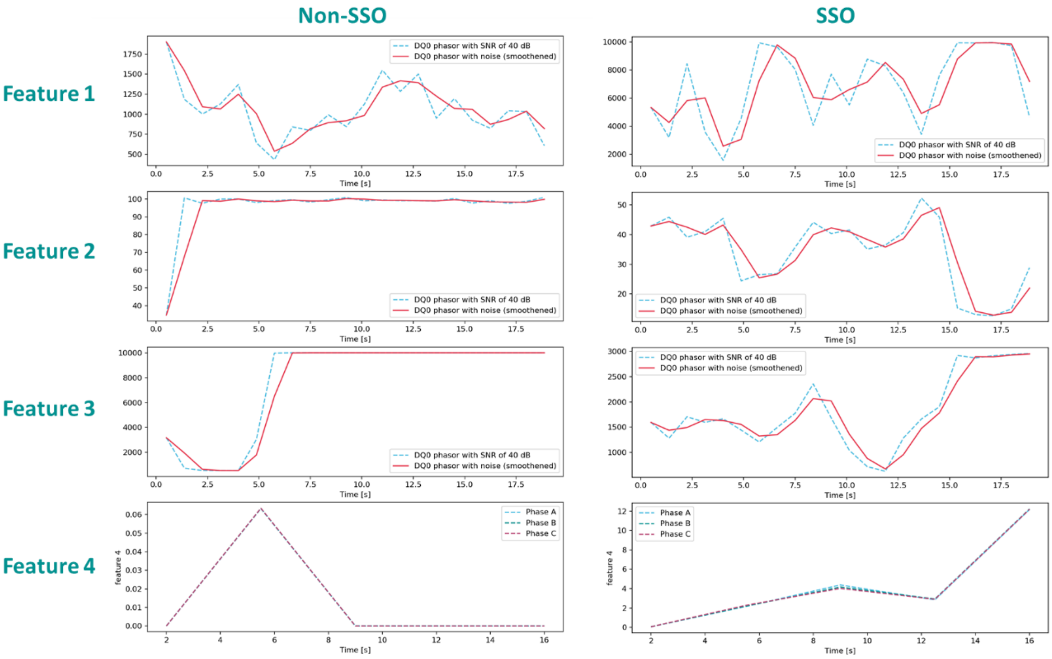


Figure 37 Features computed on the current at WF1 for S_n=3000 MVA (non-SSO) and S_n=800 MVA (SSO).

The computed features on the current at WF1 for two scenarios are seen in Figure 37. The spectrograms showed that the scenarios were either full of oscillations or only had the grid frequency, meaning no transition was present in the features. To determine if the features are distinctive, a non-SSO event is compared with an SSO event. It can be seen from the values that feature 1 and 4 have a higher value while features 2 and 3 have a lower value during the oscillations. Similar results are obtained for the other measurements. This validates the working of the features as they can create a different range of values for SSO and non-SSO events.

7.4.5 Conclusions

It is demonstrated in Figure 31, Figure 34 and Figure 37 that the features work as expected and produce different values for SSO and non-SSO events. Hence, requirement 1F is satisfied. Requirements 1A and 3A are also met because the features work on various data sources with minimal tuning. Moreover, the features can be explained intuitively, meaning that requirement 5A is achieved. The features are implemented in python, fulfilling requirement 4A.

8 Machine learning classifier

The method presented in Section 7 can transform the input measurement into features with distinct values for SSO and non-SSO events. But the validation in Section 7.4 shows that the features produce a different range for each data source. Also, the moment when the values change can differ per feature. And perhaps most importantly, only computing the features is insufficient to detect SSOs from the measurements because manual intervention is still needed to determine the knee-point (violating requirement 3A), making the method poorly scalable (not complying with requirement 2A).

Different approaches exist to fully automate detection, which were briefly discussed in Section 7.2. A simple thresholding mechanism is insufficient as it considers each feature individually rather than as an ensemble. For example, if feature 1 is high and feature 2 is low, it is expected that feature 3 also has a low value. The simple threshold would not know this relation, and the threshold would need to be determined manually.

A better idea is to train a machine learning classifier, which automatically learns the distinction in feature values between SSO and non-SSO regions from the data. This method is a bit slower than the simple threshold if the training time is considered. However, the prediction speed is fast once a good model is trained (satisfying requirements 2A, 3A and 6A). Special attention must be paid to ensure that the model can handle different data sources (requirement 1A).

This chapter will discuss the development of a machine learning classifier. Scikit-learn is one of the most popular python libraries for machine learning [50] and will be used for implementation, complying with requirement 4A. Section 8.1 presents the general approach and pipeline, and Section 8.2 discusses the need for scaling the features before using them in the classifier. The classification algorithms are briefly discussed in Section 8.3. A voting procedure is required to determine the final output (Section 8.3.4). Finally, in section 8.5, the implemented classifier is validated.

8.1 Pipeline

For some tasks, it is sufficient to have only the classifier. However, more components are required in most cases to create a working classifier. For example, scaling is almost always used to improve the accuracy and convergence speeds of algorithms that calculate distances between data, such as SVM [51]. Also, it could be desired to reduce the dimensionality of data by using methods like Principal Component Analysis (PCA). When combined, a pipeline is formed.

It is essential to think about how to take the features and predict the occurrence of an SSO. It is known from section 7.3.1 that the features are computed on segments of the measurement, yielding multiple points in time where a prediction can be made. The computed values can be considered independent or as a sequence, affecting which machine learning method is suitable.

When the calculated features are considered independent, each segment is considered a sample for the classifier. The training is done on all computed feature values, maximising the accuracy of predicting a single element. An advantage of this method is that it yields many data points for training the data. However, since the goal is to determine whether a measurement contains an SSO event, some voting mechanism is needed.

It is also possible to consider the entire measurement as a sample, meaning that the segments are considered to be related to each other. The advantage of this method is that temporal relations are captured. But there are two drawbacks. First, this method could make the classifier sensitive to the moment of SSO occurrence. Second, a strategy needs to be developed to handle different segment durations between features 1 – 3 and feature 4 and different measurement durations. This is because the classifier is most likely trained on a fixed number of inputs. It is chosen to use the first method as the drawbacks of the second method violate requirements 1A and 2A while using a voting system improves the explainability of the method (requirement 5A).

The pipeline of the current classifier is seen in Figure 8.1 and consists of three main elements: scaling, two classifiers and a voting mechanism. Each of them is elaborated on in more detail in the following sections. The pipeline is implemented as a scikit-learn classifier class called *SSOClassifier*.

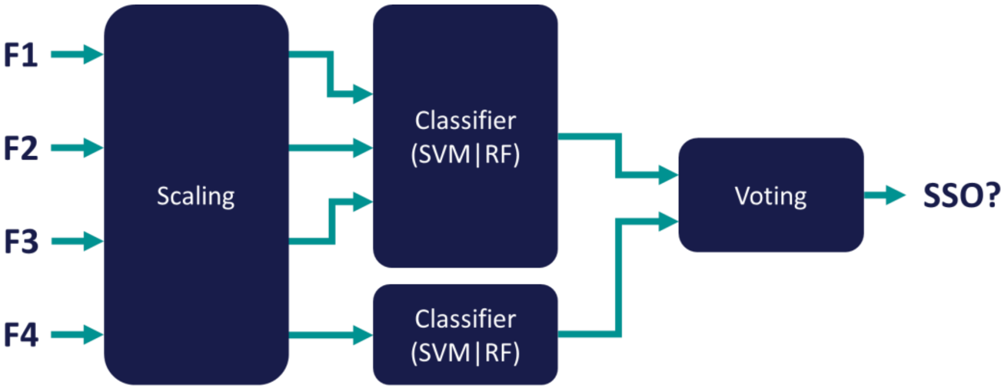


Figure 8.1 Overview of the classifier's pipeline

First, a scaling step is implemented. This is because the features could have a different range depending on the measurement source and as required for training SVMs. More details about the scaling are found in Section 8.2.

The current pipeline consists of two classifiers. Having two classifiers is required by the way the segmentation of the measurement works for the different features. Features 1 – 3 have a default segmentation of 1 second to ensure granular detection while also averaging out the signal a bit. On the other hand, feature 4 uses 4-second segments as required by (15) to achieve the desired frequency resolution. This means that for the same measurement, there will be more computed values of features 1 – 3 (see Figure 8.2) than feature 4 (as depicted in Figure 8.3). The easiest solution is to train two separate classifiers, one handling features 1 – 3 and the other making predictions on feature 4. This also adds redundancy to the detector. The training of the classifiers is elaborated in Section 8.3

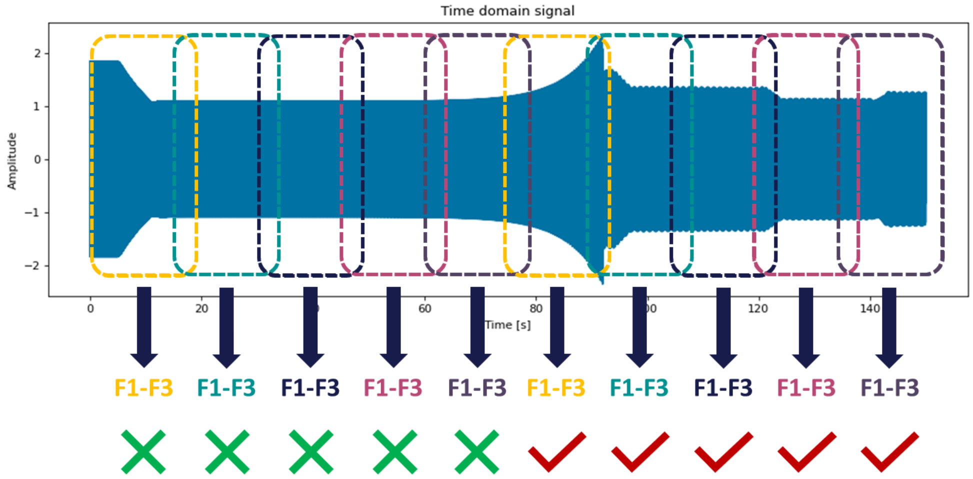


Figure 8.2 Computation of features 1 – 3 on a measurement. Created with data from [20].

The automatic detection of SSOs works as follows. First, the features are computed on a measurement using the pipeline in Figure 20, meaning that multiple values of the features are obtained. The features are scaled and inputted to the respective classifier as separate samples, yielding a prediction on each segment. Finally, the predictions are passed through a voting system to produce a single label for the measurement, classifying it as SSO or non-SSO (Section 8.4).

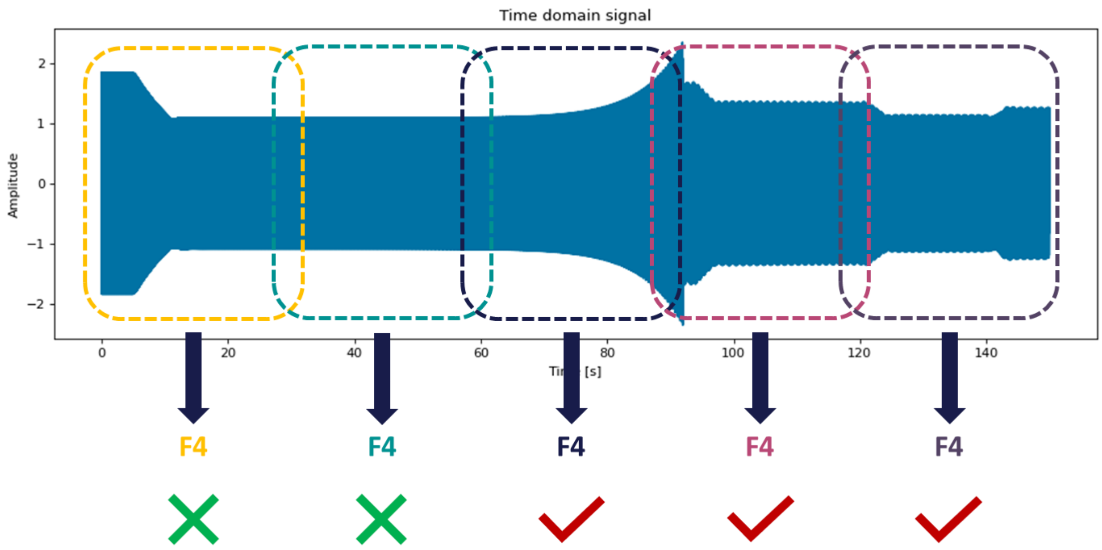


Figure 8.3 Computation of feature 4 on a measurement. Created with data from [20].

8.2 Scaling

Scaling plays a crucial role in machine learning. Sometimes different datasets produce different ranges of features, resulting in erroneous predictions and lowering the model's overall accuracy. Also, some algorithms, like SVMs, require scaling to converge quickly and correctly. For more information about scaling, see [51].

The computed features work well because they satisfy most requirements and distinguish between SSO and non-SSO events. However, the features should work for different data sources to have a good normalisation performance (requirement 1A). While their basic functionality remains the same, the produced output range could differ significantly. For example, feature 1 attains higher values for sources with a higher sample rate because more points will have the same trend; similarly, feature 3 will be higher for non-SSO regions because more points satisfy (7). Also, a more volatile envelope will mean a higher value of feature 2. Finally, measurements with more frequencies will yield a higher value for feature 4.

Implementing a scaling step reduces the impact of different data sources by moving the features' values to an identical range. For example, normalisation scales the data to the interval [0, 1]; standardisation makes the data zero mean and unit variance. The former is implemented to preserve the separation between data classes and ensure a similar range. This is done using the formula:

$$\boldsymbol{F}_{\boldsymbol{scaled}} = \frac{\boldsymbol{F} - F_{min}}{F_{max} - F_{min}} \tag{17}$$

In (17), F_{min} is the smallest and F_{max} is the largest feature value.

Scaling works because the lower boundary of the features is well-defined; features 1 and 4 are 0 for non-SSO events, while features 2 and 3 are 0 for SSO events. Scaling the data ensures that regardless of the range of the features in a different source, the extremes are aligned to the range [0, 1], meaning the learned decision boundaries can be used. This satisfies requirement 1A.

8.3 Classification algorithms

Classification is a task where one wishes to predict to what class an observation (or sample) belongs. Various methods exist to do this using machine learning. In this case, the classifier is trained using supervised learning, meaning the model is told what class observations belong during training.

As mentioned in Section 8.1, a separate classifier is implemented for features 1 – 3 and feature 4. Various classifiers exist, such as SVM, Decision Trees (DTs), Naïve-Bayes (NB), K-Nearest Neighbours (KNN) and Neural Networks (NNs). It is also possible to use a combination of them, also known as ensemble methods; a Random Forest (RF) combines multiple DTs and averages their prediction.

This section discusses three classifiers: the SVM, DT and RF. These classifiers are considered for the SSO detector. A brief explanation of each classifier is given, including its tuneable parameters and a comparison is made. For more information about the other classifiers, see [52].

8.3.1 Support Vector Machine

SVMs is a classifier that learns a boundary between two classes by finding a maximum separating hyperplane that separates the classes (see Figure 8.4). The advantage of having a maximum separating hyperplane is that it is unbiased against the two classes. This is because a hyperplane closer to one class would predict the other more often.

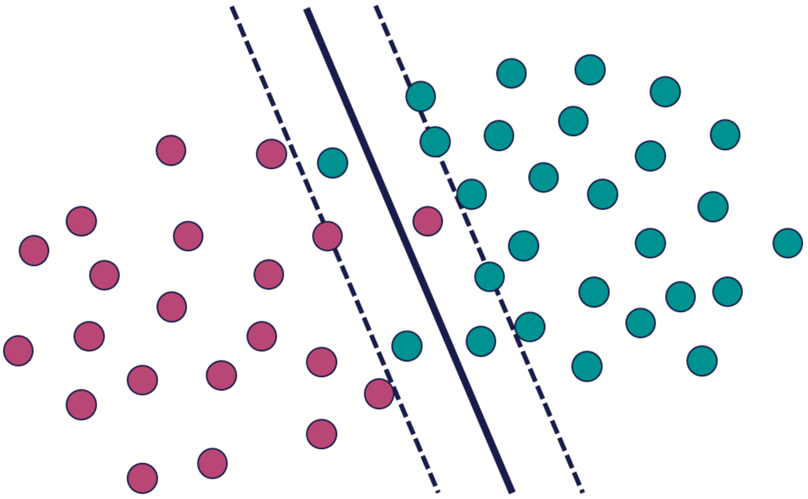


Figure 8.4 SVM classifier distinguishing between two classes by finding a maximum separating hyperplane.

The hyperplane is found by solving the following optimisation problem [53]:

$$\begin{aligned} &\min_{w,b,\zeta} \frac{1}{2} w^T w + C \sum_{i=1}^{n} \zeta_i \\ \text{subject to} \quad & y_i (w^T \phi(x_i) + b) \geq 1 - \zeta_i, \\ & \zeta_i \geq 0, i = 1, \ldots, n \end{aligned} \tag{18}$$

In (18), the term $\frac{1}{2} w^T w$ is trying to maximise the separating hyperplane, while the term $C \sum_{i=1}^{n} \zeta_i$ is penalising for the misclassification of points. The parameter C is also known as the regularisation parameter. A higher value of C places more importance on the correct classification of points i.e., the second term of (18).

Also, a kernel function $\phi(x_i)$ is used, which allows learning a non-linear decision boundary by mapping the points to a higher dimension. Different kernels exist, including the linear, Gaussian Radial Basis Function (RBF), polynomial and sigmoid kernels. For more information about the kernels, see [53].

More information about SVMs, including the mathematical formulation, is found in [54].

8.3.2 Decision Trees

A decision tree is a method that existed before machine learning to help with decision-making. It uses a tree-like structure to make choices based on rules to arrive at the final class. In this case, it would derive thresholds for each feature and build a network that can be followed.

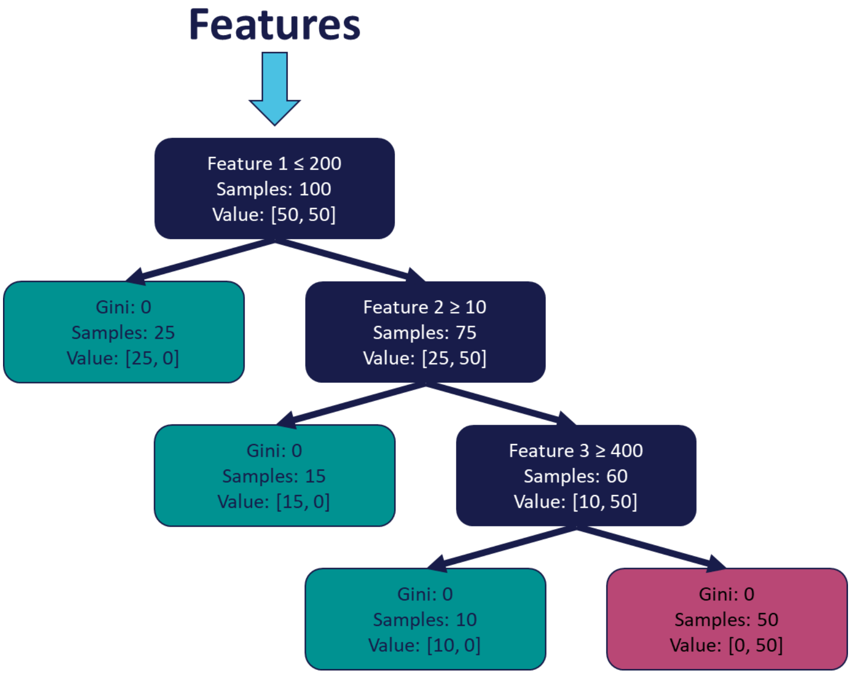


Figure 8.5 Example of a decision tree for the IEEE dataset. Disclaimer. This tree is for illustrational purposes only and is not actually fitted on the data.

An example of a DT is seen in Figure 8.5. The computed features enter from the top. First, a split is performed on the value of feature 1. If it is below 200, the segment is classified as non-SSO (first class). If it is above 200, the value of feature 2 is checked. This process is repeated until a sample is classified as non-SSO or SSO.

A splitting criterion is used to automate constructing a DT, for example, the Gini index or entropy. Both are a way to express the purity at a node.

The Gini index does this by *"calculating the amount of probability of a specific feature that is classified incorrectly when selected randomly"* [55]. A low Gini index means a higher purity node, while a high value indicates the random distribution of elements across classes.

Entropy is *"a measurement of the impurity or randomness in the datapoints"* [55]. Similar to the Gini index, a low value indicates that only one class is present, while a high value indicates randomness. Entropy is used to measure the information gain at a node.

It is also possible to define the structure of the tree by limiting the way it can grow. A maximum depth can be specified, reducing how deep a tree can grow and thus limiting the number of decision nodes. Alternatively, the maximum number of leaf nodes can be specified.
The minimum number of samples required to split an internal node can be set, ensuring

that splits are significant rather than individual samples, making the model more general. A similar effect can be realised by specifying the minimum impurity decrease in decision nodes.

By specifying max features considered for splitting a node, the building of a tree can be sped up when there are many features.

For more information about decision trees, their working and their mathematical implementation, see [56].

8.3.3 Random Forest

An RF is an ensemble method that combines multiple DTs to predict the output. Each tree is slightly different because it is trained using random resampling of the training data with replacement (also known as bagging). Also, each DT might use a subset of features, adding to the randomness.

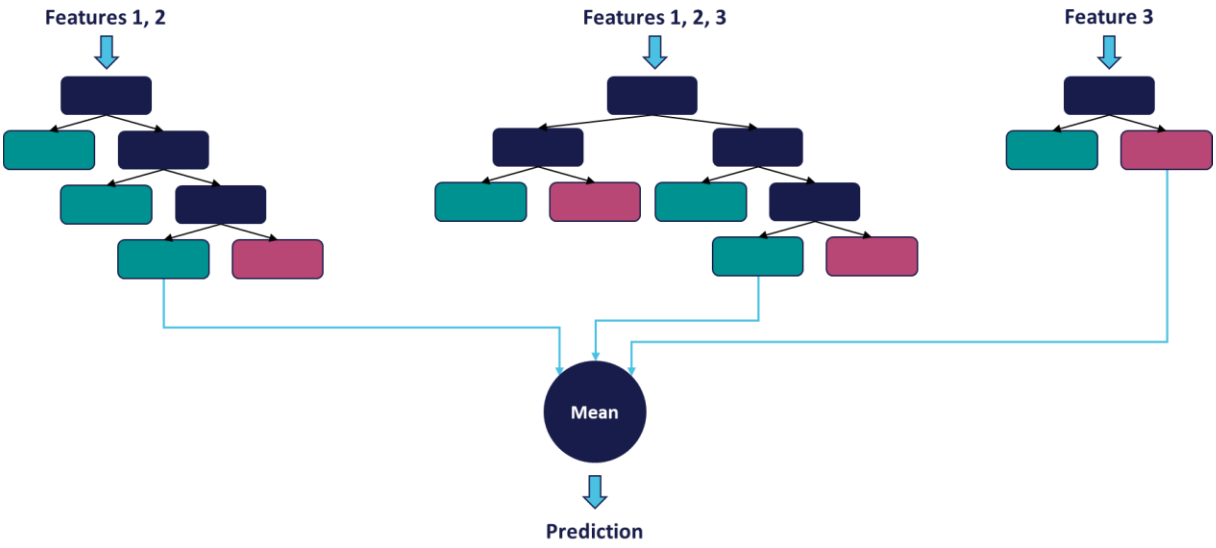


Figure 8.6 Example of a random forest consisting of three DTs.

A second variant exists called Extreme RF (ERF). It adds more randomness to RF by using random thresholds when determining the splitting nodes instead of the most discriminative threshold. This means the split is derived randomly instead of choosing the one that minimises the Gini index or maximises the information gain. This slightly increases the model's bias but reduces the variance even more.

Because an RF consists of DTs, it also shares the parameters of the DT. Moreover, the number of estimators can be specified that will be fit using the parameters for a single tree.

For more information about RF and ERF, their working and mathematical implementation, see [57].

8.3.4 Comparison

These three classifiers can all be used for the same task, and their methods are explainable as opposed to using NNs, satisfying requirement 5A. There are also some differences.

SVMs tend to generalise well and require little tuning to perform well. These are desired properties as they satisfy requirements 1A and 3A. For large datasets, efficient implementations exist that use subsets of the data, making the SVM scalable (requirement 2A). However, they are sensitive to different feature ranges, requiring a scaling step.

DTs are very explainable because of their simple decision-making structure. Moreover, they have few parameters and those only limit the form (requirements 3A and 6A). However, DTs tend to have generalisation difficulties, especially when a tree is large; this violates requirement 1A.

An RF solves the generalisation issue by creating multiple DTs randomly. Their average behaviour will reduce the variance of the model, allowing it to generalise better (requirement 1A). Also, the ensemble will still ensure good performance even if several DTs are bad models. Two downsides of the RF are that it is less explainable than a single DT (requirement 5A) and it costs more resources to train (requirement 6A). Luckily, the fitting can be sped up as the computations of the various DTs are independent, allowing for parallelisation.

8.4 Voting procedure

The segmentation of the measurement results in multiple calculated values of the features. Each computation of the features is considered as a separate sample to the classifier, resulting in multiple predictions per measurement. A voting procedure is implemented to produce a single label for the measurement:

1. Use features 1 – 3 to predict SSOs using the first classifier. This gives the predictions y_{123}, having the same length as the number of segments for features 1 – 3.
2. Use feature 4 to predict SSOs using the second classifier. This gives the predictions y_{fft}, having the same length as the number of segments for feature 4.

3. Compute the ratio of segments predicted to contain an SSO to the total number of segments, N_{SSO}.
4. If the ratio $N_{SSO} \geq \varepsilon_{voting}$ with $\varepsilon_{voting} \in [0,1]$, label the measurement as SSO. A lower value for ε_{voting} makes the detector more sensitive to SSO events because fewer SSO predictions are needed to flag a window as SSO; this also means it is more susceptible to false positives, reducing its precision.

8.5 Validation

The validation of the machine learning classifier consists of three main steps. First, the data is created using simulations of the CIGRE test bench and transformed into the appropriate form (section 8.5.1). Next, the model is trained and tuned to determine the best parameters. A suitable strategy to do this is discussed in section 8.5.2. Finally, in section 0, the performance is evaluated on different data splits.

8.5.1 Creating and preparing the data

The features are computed on measurements from the CIGRE test bench to test the pipeline of Figure 8.1. It was chosen to use this data source because it allows for the creation of various scenarios (as opposed to the IEEE dataset) while implementing complex system dynamics, which the SSO generator lacks.

Different scenarios are generated by varying the value of S_n. From each scenario, current and voltage measurements are taken at the three points specified in Figure 35, giving six signals per scenario. The test bench is simulated for $S_n = \{3000, 1500, 1300, 1100, 1000, 900, 800, 750, 700\}$ MVA. When decreasing the value of S_n, the SCR is also decreased, giving rise to oscillatory modes.

Spectrograms (such as the one in Figure 36) are computed for the six measurements to determine the frequency components present in the data. This allows for adding labels to the data, which will be used to train the classifiers. The measurement is considered to have oscillations when there is a frequency with a PSD higher than or at most -30 dB below $PSD(f_{grid})$; a lower value could be caused by spectral leakage of the Hamming window.

Labels are added per segment. This is done by specifying the simulation time where there is an oscillation (see Table 8.1) and creating an artificial signal. Then, this signal is segmented using the same duration as the feature computation. A label is assigned based on the value of the signal. If the signal is mostly 1 (SSO) inside a segment, the assigned label is 1 (SSO). This process is illustrated in Figure 8.7. Two vectors are constructed that contain the labels, called $\boldsymbol{y}_{123}$ and $\boldsymbol{y}_{fft}$.

The features are computed using the default settings, i.e., a segment duration of 1 second for features 1 – 3 and a minimum frequency resolution of 1 Hz, corresponding to a 4-second window for feature 4. The threshold ε_{LSS} of feature 3 is set to 0.002.

Table 8.1 Overview of the simulation times when oscillations occur in the different scenarios.

S_{grid} [MVA]	Grid current	Grid voltage	POC current	POC voltage	WF current	WF voltage
3000	n.a.	n.a.	n.a.	n.a.	n.a.	n.a.
1500	0 – 6 s	0 – 6 s	0 – 6 s	0 – 6 s	0 – 6 s	0 – 6 s
1300	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s
1100	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s
1000	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s
900	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s
800	0 – 15.5 s	0 – 20 s	0 – 15.5 s	0 – 20 s	0 – 20 s	0 – 20 s
750	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s
700	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s	0 – 20 s

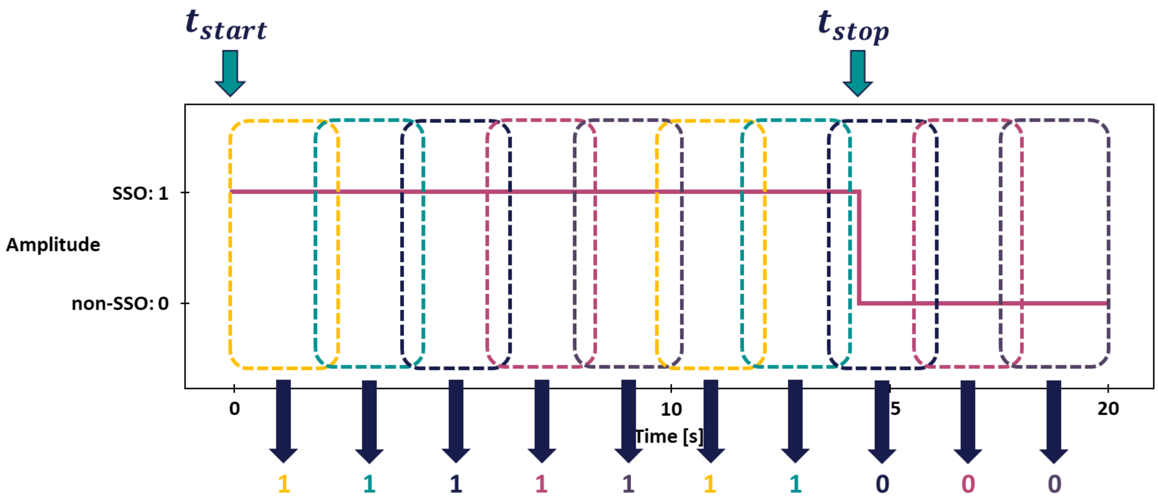


Figure 8.7 Example computation of the labels on the grid current with $S_n = 800\ MVA$.

The data must be structured in a matrix $\boldsymbol{X}$ where each row corresponds to a sample and each column to a feature. Because the computed features on each segment are considered independent, a single measurement will result in multiple samples for matrix $\boldsymbol{X}$. Also, two matrices are created, $\boldsymbol{X}_{123}$for features 1 – 3 and $\boldsymbol{X}_{fft}$ for feature 4. This is because feature 4 uses a different segment duration. Note that $\boldsymbol{X}_{fft}$ also has three columns because feature 4 is computed on each phase.

8.5.2 Cross-validation, performance metrics and grid search

The first classifier can be fitted using $\boldsymbol{X}_{123}$ and $\boldsymbol{y}_{123}$ and the second using $\boldsymbol{X}_{fft}$ and $\boldsymbol{y}_{fft}$. Usually, the data is split into three sets: training, validation and test set. The training set is used to fit the classifier, the validation set is used to evaluate its performance and find the optimal parameters, and the test set is used to evaluate the model on new data. However, this reduces the number of samples, and the performance depends on the particular splits.

Cross-validation (CV) is a solution to this problem. In the simplest version, K-fold CV, the data is split into k partitions. If desired, the data is first shuffled. Then, the model is trained using $k-1$ folds; validation is performed on the held-out fold. This process is repeated, and a different fold is retained each time, resulting in a total of k different splits on which the model is evaluated. The performance of the model is the average performance across all splits. A model has low variance when the performance across the splits does not vary significantly; moreover, it has a low bias when it shows excellent performance on the different splits.

A particular version of the CV is used: stratified K-fold. This method is visualised in Figure 8.8. The key difference is that stratified K-fold creates the folds on each class, meaning that the percentage of each class in the splits is equal to that of the entire dataset.

Having this property is crucial for problems with an imbalance between the classes. Measurements will mostly not contain oscillations because the operators prevent them as much as possible, resulting in class imbalance.

Note: It can be seen from Table 8.1 that the data contains more oscillations, resulting in an opposite imbalance. Nevertheless, it was shown that the features generalise well, meaning they can also produce the correct range of values for the non-SSO class. The classification algorithms can compensate for this imbalance during training [53], meaning they will distinguish between the two classes equally.

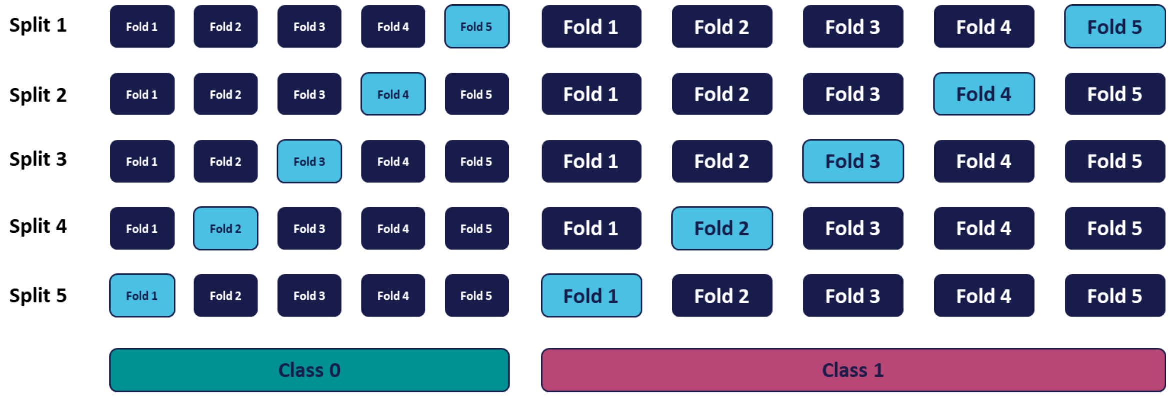


Figure 8.8 Visualisation of stratified K-fold cross-validation.

Scikit-learn implements various performance metrics [58] to evaluate the prediction performance of a model. The performance metric of interest is balanced accuracy which is the average accuracy of the model on each class. It is essential to use the balanced accuracy to have an accurate image when working with unbalanced problems. Otherwise, a model could be accurate while constantly misclassifying the minority class. This performance metric will also be used during CV and tuning.

A classifier's performance also depends on the used parameters. For example, using a too-low value for the regularisation parameter C in SVMs could result in a decision boundary that misclassifies too many points. Evaluating the parameters of a classifier to find the ones that perform the best is also known as grid search. This is done by trying multiple parameters and storing the model's performance, allowing one to choose the best parameters later. For more information about grid search, see [59].

Exhaustive grid search is used together with CV to tune the model. In an exhaustive grid search, a range of values is tried for each parameter. The model is then evaluated using the CV method, yielding the average performance of the classifier for those particular combinations of parameters. The parameter values are then chosen based on the combination that gives the best performance.

It is unknown how the real-world dataset will look. The best thing that can be done to ensure a good generalisation performance (requirement 1A) is to try different orderings of the dataset. This final evaluation of the model with the optimal parameters is performed by varying the random seed. This approach is like rerunning the CV multiple times, but the results are reproducible by using fixed random seeds.

8.5.3 Performance evaluation of the model

The first evaluated classifier models are based on SVMs. The only parameters that are varied are C, and the kernel coefficient γ as these are the only parameters used in the linear and RBF kernels, simplifying the tuning.

C is the regularisation parameter, and a higher value places more importance on the correct classification of points.

γ is the kernel coefficient for the RBF kernel. The effect of γ is illustrated in [60], but generally, a higher value will make the boundary curve more around the classes. The results of the initial grid search are seen in Figure 8.9 for features 1 – 3 and in Figure A.24 and Figure A.25 for feature 4.

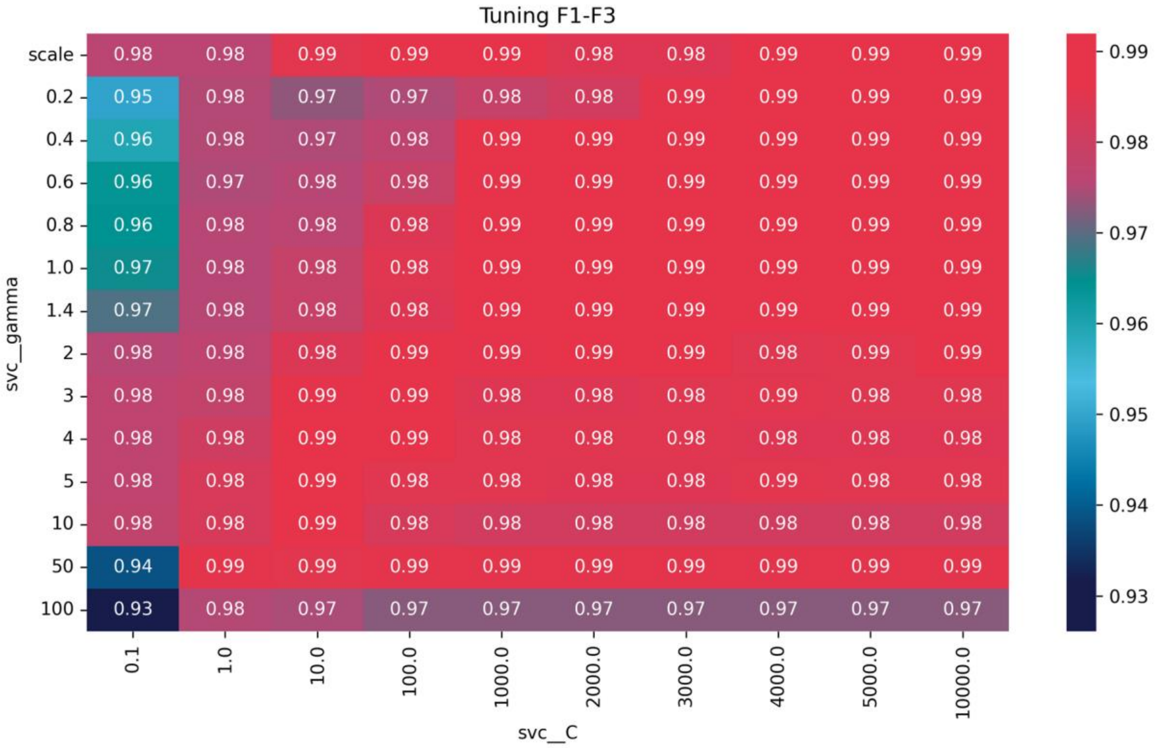
SVM has excellent results for features 1 – 3, yielding accuracies above 93%. For $C \geq 1$, the accuracies are 97% or higher; the default value of $C = 1$ is kept ensuring maximum separation. γ shows little influence and is therefore left to its default value *scale*. This means that this classifier type is fast to tune, satisfying requirements 3A and 6A.

The SVM classifier shows a much poorer performance for feature 4. Only with $C \geq 4000$, an accuracy above 80% is obtained. This can be explained by looking at the values of feature 4 (Figure A.26). Feature 4 will always be around 0 for measurements that do not contain SSOs. On the other hand, the values of the SSO class can range significantly. Misclassifying points of the non-SSO class will result in smaller penalties in (18) because the distance to the decision boundary is closer, meaning a higher value of C is needed to account for misclassification.

However, even with a high value for C, the SVM classifier shows high variance (see Figure A.27). This indicates that the model is dependent on the data split, resulting in bad generalisation performance (up to 15% accuracy loss). This can be explained as follows. Based on the data split, some points are missing near the actual decision boundary (Figure A.28) because they are held back for testing. The resulting decision boundary is then placed in the wrong position. Once the classifier is tested, it will misclassify the points resulting in lower accuracy. This effect can be more substantial depending on the particular split, meaning SVM might not be the best classifier for feature 4.

To fix this, an RF is considered as the classifier for feature 4. The varied parameters are the number of estimators and the maximum depth. The results of the tuning are shown in Figure A.29. The RF is a much better classifier yielding a performance of around 95%; the accuracy also seems to be similar for the different parameters. This means it is possible to train a simpler RF consisting of fewer and shallower trees, reducing the computational complexity, and improving the execution speed (requirement 6A). The RF is probably working better because a simple threshold is sufficient to distinguish between the values of feature 4.

However, a simpler RF is susceptible to the drawbacks of DTs. For example, the trees tend to be overfitted to the data. This is seen by a tree performing multiple splits on the same variable while only one would be sufficient (see Figure A.30). Moreover, the fitted trees do not consider each phase as equally important. This means that the RF is biased towards detecting SSOs on certain phases.



svc__gamma \ svc__C	0.1	1.0	10.0	100.0	1000.0	2000.0	3000.0	4000.0	5000.0	10000.0
scale	0.98	0.98	0.99	0.99	0.99	0.98	0.98	0.99	0.99	0.99
0.2	0.95	0.98	0.97	0.97	0.98	0.98	0.99	0.99	0.99	0.99
0.4	0.96	0.98	0.97	0.98	0.99	0.99	0.99	0.99	0.99	0.99
0.6	0.96	0.97	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
0.8	0.96	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
1.0	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
1.4	0.97	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.99	0.99
2	0.98	0.98	0.98	0.99	0.99	0.99	0.99	0.98	0.99	0.99
3	0.98	0.98	0.99	0.99	0.98	0.98	0.98	0.99	0.98	0.98
4	0.98	0.98	0.99	0.99	0.98	0.98	0.98	0.98	0.98	0.98
5	0.98	0.98	0.99	0.98	0.98	0.98	0.98	0.99	0.98	0.98
10	0.98	0.98	0.99	0.98	0.98	0.98	0.98	0.98	0.98	0.98
50	0.94	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99	0.99
100	0.93	0.98	0.97	0.97	0.97	0.97	0.97	0.97	0.97	0.97

Figure 8.9 Grid search with CV for SVM as the classifier for features 1 – 3. The values inside the cells represent the balanced accuracy for different parameters.

It is still desired to have a classifier based on RF as that one works substantially better. By combining knowledge about the working of the features with the results from the grid search, an optimised RF is created (Figure 8.10). The RF consists of three shallow DTs, each performing a single split on feature 4 from a different phase. After that, there is a vote on the output of the three DTs. The implemented schemes are *all* (all classifiers have to predict an SSO), *majority* and *one*.

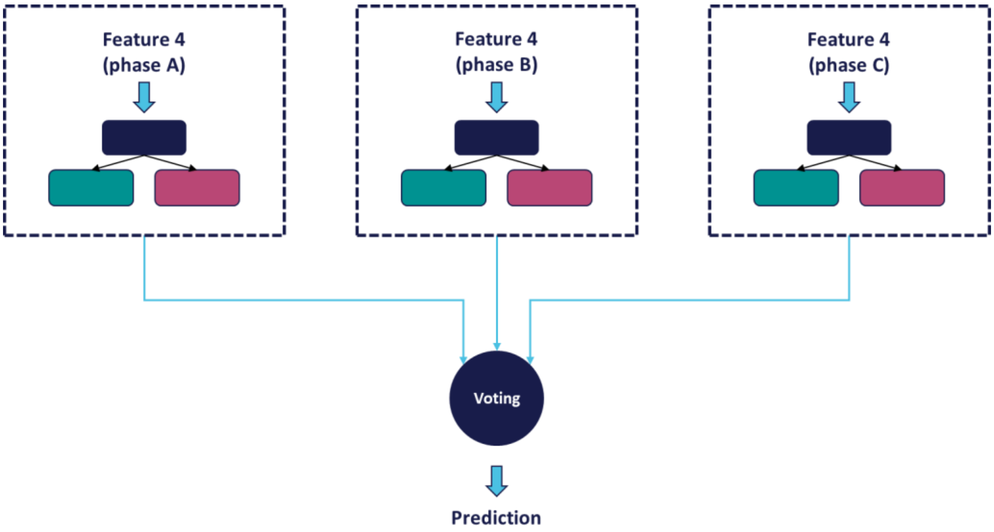


Figure 8.10 Optimised RF consisting of three simple DTs, one on each feature.

This optimised RF is the most attractive classifier for feature 4 out of the three. It is highly explainable because of its simple structure (requirement 5A). Moreover, it does not require any tuning and is much faster to fit than the RF (satisfying requirements 3A and 6A).

Finally, the generalisation performance of the implemented classifiers is evaluated.

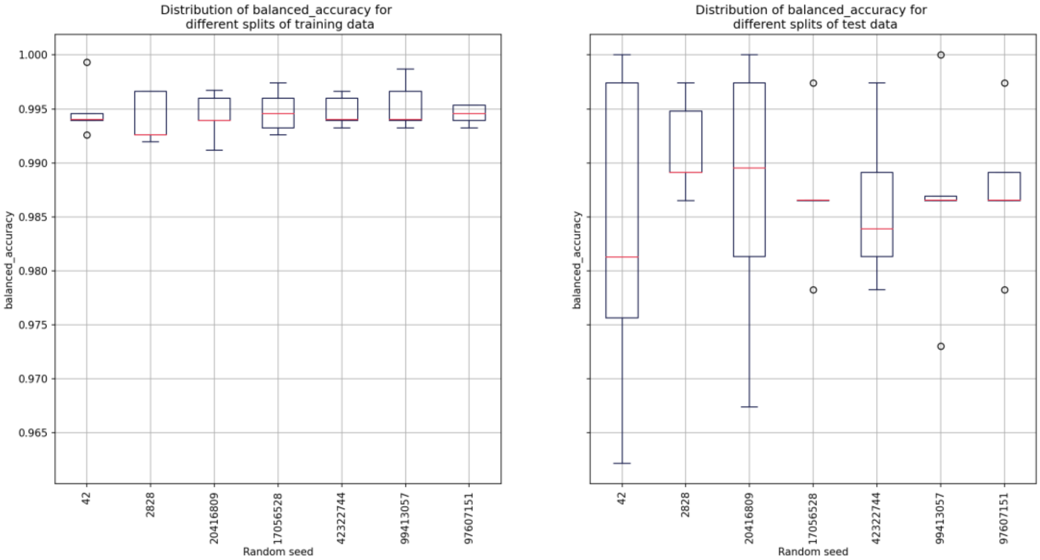


Figure 8.11 Performance evaluation of the tuned SVM classifier of features 1 – 3.

From Figure 8.11, it can be seen that the SVM classifier has high accuracy (98%) and low variance (± 4%) for different data splits. Also, it indicates a good generalisation performance, satisfying requirement 1A.

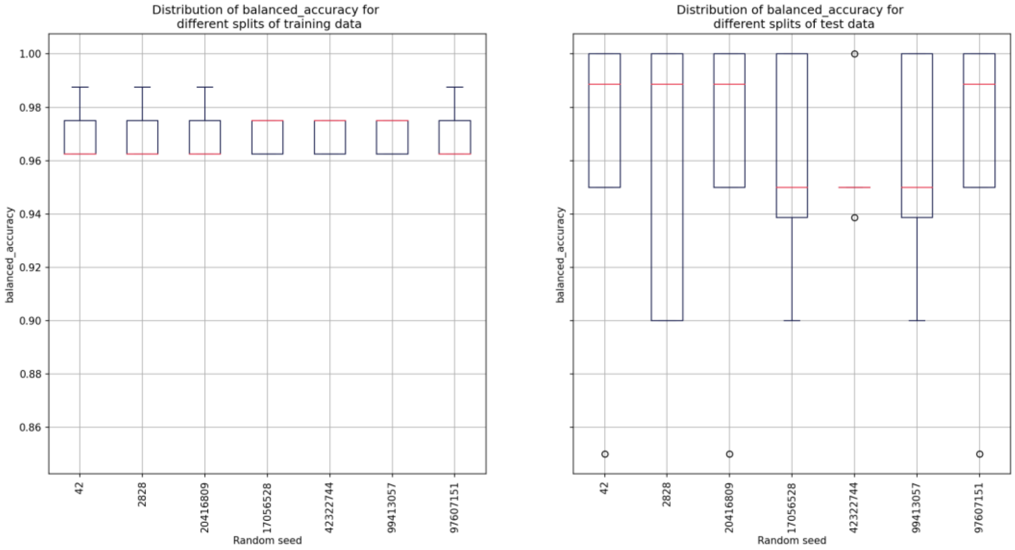


Figure 8.12 Performance evaluation of the optimised RF classifier of feature 4.

The optimised RF classifier has a slightly lower accuracy with an average of 97% accuracy, but it is still outstanding and higher than with SVM. It does show a little more variance (± 7.5%), but it is less than with the SVM-type classifier. This means that this classifier also has a good generalisation performance, complying with requirement 1A.

The performance of the classifiers is further discussed with the help of a confusion matrix in Figure 8.13. The true labels correspond to the ground truth i.e., the SSO/non-SSO outcome based on the judgement of an engineer by inspecting time domain simulation results. The predicted labels are the outcome of the classifiers which are trained using the four features (Section 7.3). The results show that the number of misclassifications is very low. As an example, the number of SSO events classified as non-SSO by features 1-3 are just 2 out of 288 SSO events. This corroborates with the accuracy figures shown in Figure 8.11 and Figure 8.12 .

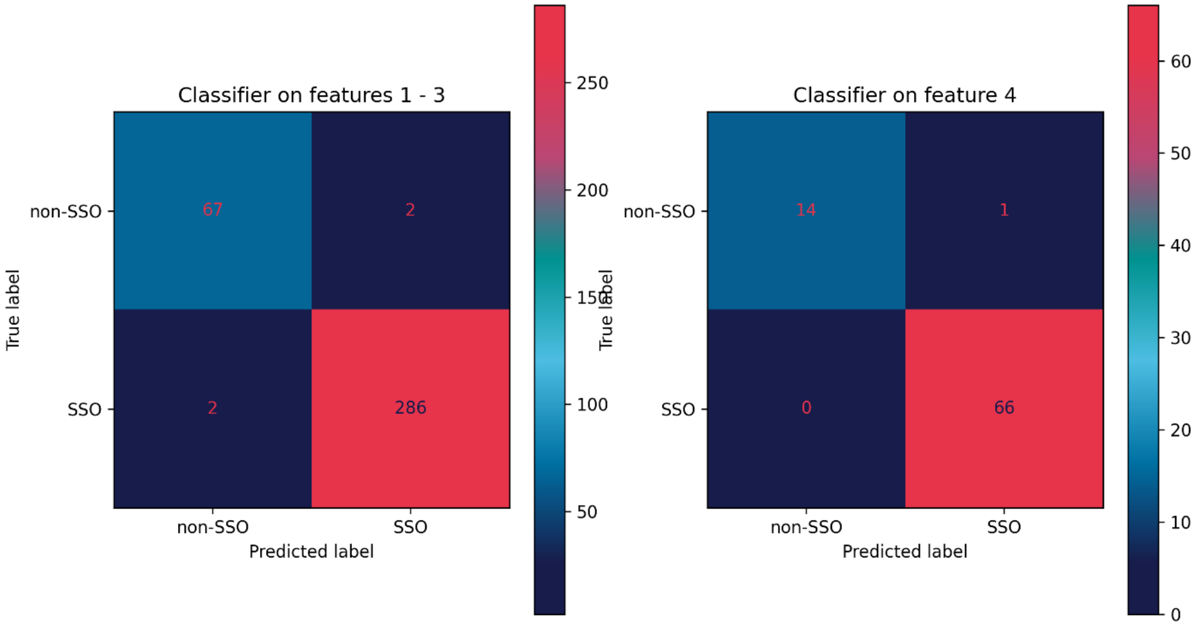


Figure 8.13 Confusion matrix for features 1 to 4 showing the performance of the classifiers

Both classifiers have an outstanding performance in predicting SSOs on both instantaneous and phasor measurements. Having this capability means that the SSO detector can work on various sources and is scalable (requirements 1A and 2A), meaning that the design satisfies the requirements.

8.6 Summary

The automatic detection of SSOs is realised using two steps. First, features are engineered based on the measurements. Then, a machine learning classifier is used to distinguish between SSO and non-SSO events.

Feature engineering is done by segmenting the measurement and computing four features on the parts. The first three features work on the phasor domain, while the last uses instantaneous measurement.

The features can be explained intuitively, satisfying requirement 5A. Also, it was proven that the features produce distinctive ranges of values between SSO and non-SSO regions for different data sources (requirement 1F), showing a good generalisation performance of the features (satisfying requirement 1A). Moreover, little tuning was required to achieve this, showing that requirement 3A was also met

The fitted machine learning classifier is based on Support Vector Machines (SVM) and an optimised Random Forest (RF). Both methods are explainable, especially the latter and hence satisfy requirement 5A. Because of this simplicity, fitting both classifiers is quick and easy, meaning requirements 3A and 6A are met.

The SVM classifier was used to predict using features 1 – 3. It has high accuracy (98%), and low variance (± 4%) for different data splits.

The optimised RF was designed to determine SSO occurrence using feature 4. It was found to have a slightly lower accuracy of 97%. Also, it does show a little more variance (± 7.5%). Both classifiers have an outstanding performance in predicting SSOs on both instantaneous and phasor measurements (requirement 1F). This capability means that the SSO detector can work on various sources and is scalable (requirements 1A and 2A).

This automatic SSO detector is implemented using python and some of its most popular libraries, meaning it is accessible (requirement 4A).

8.7 Recommendations

It was seen during feature validation that the default parameters might not always be suitable for all sources. Therefore, it is recommended to tune the features beforehand to ensure a distinct range is produced between SSO and non-SSO regions.

Scaling using normalisation might be insufficient when different data sources produce a different spread of the values. In those cases, another scaling method could be used. Alternatively, the data could be resampled to ensure an even sample rate across the sources.

A different voting procedure can be implemented depending on the performance, e.g., voting on each classifier separately and producing an SSO label if one classifier predicts an SSO.

The classifiers are trained only once before deploying them for prediction, also known as offline learning. If scenarios arise where the classifier is not performing as expected, an online learning scheme can be added where the classifier is retrained using new data. Scikit-learn implements a stochastic gradient descent (SGD) classifier that can be refitted later.

Finally, the implemented detector generalises so well that it can detect all oscillations. It is recommended to add a low-pass filter when detection of only the SSO range is desired.

9 Conclusion

The identification of subsynchronous oscillations (SSO) is becoming increasingly important as more wind farms are connected to the electrical power system. However, SSO caused by controller interactions is challenging to identify without studying a wide range of scenarios. EMT simulation has a significant computational burden and cannot be used to scan through a high number of scenarios. By combining frequency domain techniques, automating EMT studies, and using machine learning methods, it is possible to develop a framework that can screen a large number of scenarios and identify hidden threats without significant manual intervention. In the next stage of the project, the methods identified will be implemented in Python and used with the Cigre benchmark model to create a proof-of-concept tool.

10 References

[1] Prabha Kundur, *Power System Stability And Control by Prabha Kundur*. New York: McGraw-Hill, 1994. Accessed: Feb. 03, 2023. [Online]. Available: https://docs.google.com/file/d/0ByS7mm27UFt9Q1pHazdpRE1aaEU/edit?resourcekey=0-rpispiP7Pavmul4LZJEyow

[2] Z. Jiang, L. Yang, Z. Gao, and T. Moan, 'Numerical Simulation of a Wind Turbine with a Hydraulic Transmission System', *Energy Procedia*, vol. 53, pp. 44–55, 2014, doi: https://doi.org/10.1016/j.egypro.2014.07.214.

[3] K. R. Padiyar, *Analysis of Subsynchronous Resonance in Power Systems*. Springer US, 1999. [Online]. Available: https://books.google.co.uk/books?id=QMSELoMjsg0C

[4] R. G. Farmer, A. L. Schwalb, and E. L I Katz, 'PROJECT REPORT ON SUBSYNCHRONOUS RESONANCE ANALYSIS AND SOLUTIONS', 1977.

[5] P. M. Anderson, B. L. Agrawal, and J. E. van Ness, *Subsynchronous Resonance in Power Systems*. Wiley, 1999. [Online]. Available: https://books.google.co.uk/books?id=bYdNE_uAdZAC

[6] R. N. Damas, Y. Son, M. Yoon, S. Y. Kim, and S. Choi, 'Subsynchronous Oscillation and Advanced Analysis: A Review', *IEEE Access*, vol. 8, pp. 224020–224032, 2020, doi: 10.1109/ACCESS.2020.3044634.

[7] D. Kidd and P. Hassink, 'Transmission operator perspective of Sub-Synchronous Interaction', in *PES T&D 2012*, 2012, pp. 1–3. doi: 10.1109/TDC.2012.6281627.

[8] X. Xie, X. Zhang, H. Liu, H. Liu, Y. Li, and C. Zhang, 'Characteristic Analysis of Subsynchronous Resonance in Practical Wind Farms Connected to Series-Compensated Transmissions', *IEEE Transactions on Energy Conversion*, vol. 32, no. 3, pp. 1117–1126, Sep. 2017, doi: 10.1109/TEC.2017.2676024.

[9] M. Beza and M. Bongiorno, 'On the risk for subsynchronous control interaction in type 4 based wind farms', *IEEE Trans Sustain Energy*, vol. 10, no. 3, pp. 1410–1418, Jul. 2019, doi: 10.1109/TSTE.2018.2889181.

[10] I. P. E. Society and I. P. E. Society. P. S. E. Committee, *Analysis and Control of Subsynchronous Resonance: Presented at the IEEE Power Engineering Society 1976 Winter Meeting and*

Tesla Symposium. IEEE, 1976. [Online]. Available: https://books.google.co.uk/books?id=VVt1JgAACAAJ

[11] J. Adams, C. Carter, and S. H. Huang, 'ERCOT experience with Sub-synchronous Control Interaction and proposed remediation', in *Proceedings of the IEEE Power Engineering Society Transmission and Distribution Conference*, 2012. doi: 10.1109/TDC.2012.6281678.

[12] N. Rostamkolai, R. J. Piwko, E. v Larsen, D. A. Fisher, M. A. Mobarak, and A. E. Poitras, 'Subsynchronous interactions with static VAR compensators-concepts and practical implications', *IEEE Transactions on Power Systems*, vol. 5, no. 4, pp. 1324–1332, 1990, doi: 10.1109/59.99384.

[13] J. R. Ramamurthy and L. J. Bohmann, 'Wind Plant Interaction in a 345-kV Series-compensated Power System—Electromagnetic Transient Program Modeling and Event Analysis', *Electric Power Components and Systems*, vol. 44, no. 15, pp. 1745–1756, Sep. 2016, doi: 10.1080/15325008.2016.1198943.

[14] G. Irwin, A. Jindal, and A. Isaacs, *Sub-synchronous control interactions between type 3 wind turbines and series compensated AC transmission systems*. 2011. doi: 10.1109/PES.2011.6039426.

[15] X. Xie, X. Guo, and Y. Han, 'Mitigation of Multimodal SSR Using SEDC in the Shangdu Series-Compensated Power System', *IEEE Transactions on Power Systems*, vol. 26, no. 1, pp. 384–391, 2011, doi: 10.1109/TPWRS.2010.2047280.

[16] L. Wang, X. Xie, Q. Jiang, H. Liu, Y. Li, and H. Liu, 'Investigation of SSR in Practical DFIG-Based Wind Farms Connected to a Series-Compensated Power System', *IEEE Transactions on Power Systems*, vol. 30, no. 5, pp. 2772–2779, 2015, doi: 10.1109/TPWRS.2014.2365197.

[17] J. Sun *et al.*, 'Renewable energy transmission by HVDC across the continent: system challenges and opportunities', *CSEE Journal of Power and Energy Systems*, vol. 3, no. 4, pp. 353–364, 2017, doi: 10.17775/CSEEJPES.2017.01200.

[18] H. Liu *et al.*, 'Subsynchronous Interaction Between Direct-Drive PMSG Based Wind Farms and Weak AC Networks', *IEEE Transactions on Power Systems*, vol. 32, no. 6, pp. 4708–4720, 2017, doi: 10.1109/TPWRS.2017.2682197.

[19] 'Wind Energy Systems Sub-Synchronous Oscillations: Events and Modeling PREPARED BY THE Analytic Methods for Power Systems (AMPS) Committee Transient Analysis and Simulations Subcommittee (TASS) Wind SSO Task Force PES-TR80', 2020.

[20] 'ESO Technical Report - FINAL2'.

[21] R. Ortega, J. Romero, P. Borja, and A. Donaire, 'PID Passivity-Based Control of Nonlinear Systems with Applications', New Jersey, 2021.

[22] Y. Gu, W. Li, and X. He, 'Passivity-Based Control of DC Microgrid for Self-Disciplined Stabilization', *IEEE Transactions on Power Systems*, vol. 30, no. 5, pp. 2623–2632, Sep. 2015, doi: 10.1109/TPWRS.2014.2360300.

[23] B. L. Agrawal and R. G. Farmer, 'Use of Frequency Scanning Techniques for Subsynchronous Resonance Analysis', *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, no. 2, pp. 341–349, 1979, doi: 10.1109/TPAS.1979.319312.

[24] J. Adams, C. Carter, and S.-H. Huang, 'ERCOT experience with Sub-synchronous Control Interaction and proposed remediation', in *PES T&D 2012*, 2012, pp. 1–5. doi: 10.1109/TDC.2012.6281678.

[25] W. Ren and E. Larsen, 'A Refined Frequency Scan Approach to Sub-Synchronous Control Interaction (SSCI) Study of Wind Farms', *IEEE Transactions on Power Systems*, vol. 31, no. 5, pp. 3904–3912, 2016, doi: 10.1109/TPWRS.2015.2501543.

[26] B. Badrzadeh, M. Sahni, Y. Zhou, D. Muthumuni, and A. Gole, 'General methodology for analysis of sub-synchronous interaction', *IEEE Transactions on Power Systems*, vol. 28, no. 2, pp. 1858–1869, 2013, doi: 10.1109/TPWRS.2012.2225850.

[27] P. Mars, M. Miskiewicz, P. Błaszczyk, and T. Kuczek, 'Modelling of ABB solar inverters in power system simulations', 2018. https://new.abb.com/news/detail/9998/modeling-of-abb-solar-inverters-in-power-system-simulations (accessed Feb. 03, 2023).

[28] X. Wang and F. Blaabjerg, 'Harmonic Stability in Power Electronic-Based Power Systems: Concept, Modeling, and Analysis', *IEEE Trans Smart Grid*, vol. 10, no. 3, pp. 2858–2870, 2019, doi: 10.1109/TSG.2018.2812712.

[29] S. Shah and L. Parsa, 'Impedance Modeling of Three-Phase Voltage Source Converters in DQ, Sequence, and Phasor Domains', *IEEE Transactions on Energy Conversion*, vol. 32, no. 3, pp. 1139–1150, 2017, doi: 10.1109/TEC.2017.2698202.

[30] A. Rygg, M. Molinas, C. Zhang, and X. Cai, 'A Modified Sequence-Domain Impedance Definition and Its Equivalence to the dq-Domain Impedance Definition for the Stability Analysis of AC Power Electronic Systems', *IEEE J Emerg Sel Top Power Electron*, vol. 4, no. 4, pp. 1383–1396, 2016, doi: 10.1109/JESTPE.2016.2588733.

[31] Y. Zhu, 'Impedance Model Analysis and Measurement for Power System Stability', Imperial College London, London, 2022. Accessed: Feb. 03, 2023. [Online]. Available: https://spiral.imperial.ac.uk/handle/10044/1/100373

[32] Y. Han *et al.*, 'Modeling and Stability Analysis of LCL -Type Grid-Connected Inverters: A Comprehensive Overview', *IEEE Access*, vol. 7, pp. 114975–115001, 2019, doi: 10.1109/ACCESS.2019.2935806.

[33] J. Sun, 'Impedance-Based Stability Criterion for Grid-Connected Inverters', *IEEE Trans Power Electron*, vol. 26, no. 11, pp. 3075–3078, 2011, doi: 10.1109/TPEL.2011.2136439.

[34] B. Wen, D. Boroyevich, P. Mattavelli, R. Burgos, and Z. Shen, 'Impedance-based analysis of grid-synchronization stability for three-phase paralleled converters', in *2014 IEEE Applied Power Electronics Conference and Exposition - APEC 2014*, 2014, pp. 1233–1239. doi: 10.1109/APEC.2014.6803464.

[35] C. Zhang, M. Molinas, A. Rygg, and X. Cai, 'Impedance-Based Analysis of Interconnected Power Electronics Systems: Impedance Network Modeling and Comparative Studies of Stability Criteria', *IEEE J Emerg Sel Top Power Electron*, vol. 8, no. 3, pp. 2520–2533, 2020, doi: 10.1109/JESTPE.2019.2914560.

[36] A. Rygg, M. Molinas, E. Unamuno, C. Zhang, and X. Cai, 'A simple method for shifting local dq impedance models to a global reference frame for stability analysis', Jun. 2017, [Online]. Available: http://arxiv.org/abs/1706.08313

[37] Y. Gu, Y. Li, Y. Zhu, and T. C. Green, 'Impedance-Based Whole-System Modeling for a Composite Grid via Embedding of Frame Dynamics', *IEEE Transactions on Power Systems*, vol. 36, no. 1, pp. 336–345, 2021, doi: 10.1109/TPWRS.2020.3004377.

[38] N. Cifuentes, M. Sun, R. Gupta, and B. C. Pal, 'Black-Box Impedance-Based Stability Assessment of Dynamic Interactions Between Converters and Grid', *IEEE Transactions on Power Systems*, vol. 37, no. 4, pp. 2976–2987, 2022, doi: 10.1109/TPWRS.2021.3128812.

[39] E. Ebrahimzadeh, F. Blaabjerg, X. Wang, and C. L. Bak, 'Bus Participation Factor Analysis for Harmonic Instability in Power Electronics Based Power Systems', *IEEE Trans Power Electron*, vol. 33, no. 12, pp. 10341–10351, 2018, doi: 10.1109/TPEL.2018.2803846.

[40] Y. Zhu, Y. Gu, Y. Li, and Timothy. C. Green, 'Impedance-based Root-cause Analysis: Comparative Study of Impedance Models and Calculation of Eigenvalue Sensitivity', *IEEE Transactions on Power Systems*, p. 1, 2022, doi: 10.1109/TPWRS.2022.3179143.

[41] M. Amin and M. Molinas, ‘Small-Signal Stability Assessment of Power Electronics Based Power Systems: A Discussion of Impedance- and Eigenvalue-Based Methods’, *IEEE Trans Ind Appl*, vol. 53, no. 5, pp. 5014–5030, 2017, doi: 10.1109/TIA.2017.2712692.

[42] Y. Wang, X. Wang, F. Blaabjerg, and Z. Chen, ‘Harmonic Instability Assessment Using State-Space Modeling and Participation Analysis in Inverter-Fed Power Systems’, *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 806–816, 2017, doi: 10.1109/TIE.2016.2588458.

[43] Z. Huang, Y. Cui, and W. Xu, ‘Application of Modal Sensitivity for Power System Harmonic Resonance Analysis’, in *2007 IEEE Power Engineering Society General Meeting*, 2007, p. 1. doi: 10.1109/PES.2007.386295.

[44] Y. Zhu, Y. Gu, Y. Li, and T. C. Green, ‘Participation Analysis in Impedance Models: The Grey-Box Approach for Power System Stability’, Feb. 2021, [Online]. Available: http://arxiv.org/abs/2102.04226

Appendix A : Phasor Computation

Features 1 – 3 are computed on the phasor of a signal. Situations can occur where the phasor is unavailable, or the sample rate is insufficient. It is possible to compute the phasor from the instantaneous measurement in that case. This section describes a simple method based on the Clarke and Park transforms. [61] presents an extensive summary of more advanced synchro phasor estimation algorithms that use the DFT.

A three-phase measurement has four dimensions: phase A, phase B, phase C and a time vector. The Clarke transform can be used to obtain a modified frame for analysing the unbalanced three-phase problems. The power-variant Clarke transform is given by:

$$\begin{bmatrix} V_\alpha \\ V_\beta \\ V_\gamma \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{1}$$

The inverse transform is given by:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \cdot \begin{bmatrix} V_\alpha \\ V_\beta \\ V_\gamma \end{bmatrix} \tag{2}$$

When all components are balanced, $V_\gamma = 0$, effectively reducing the dimension from four to three.

However, the problem is that the vector given by $\boldsymbol{v}(\boldsymbol{t}) = \begin{bmatrix} V_\alpha \\ V_\beta \end{bmatrix}$ is rotating with a frequency of f_{grid}. This makes the design of control schemes more difficult as an oscillatory control signal needs to be created. This can be solved by rotating the $\alpha\beta - \mathrm{axes}$ with the grid frequency f_{grid}, creating the $\mathrm{dq} - \mathrm{axes}$. If the vector $\boldsymbol{v}(\boldsymbol{t})$ is viewed in this new reference frame, it appears to be stationary (see Figure C.1). This means that a DC signal is created from an AC signal. The combination of the Clark and Park transforms is also known as the DQ0 transformation. This transformation allows the designing of control schemes for Power Electronics (PE), facilitating the creation of converters and DC subsystems in the grid.

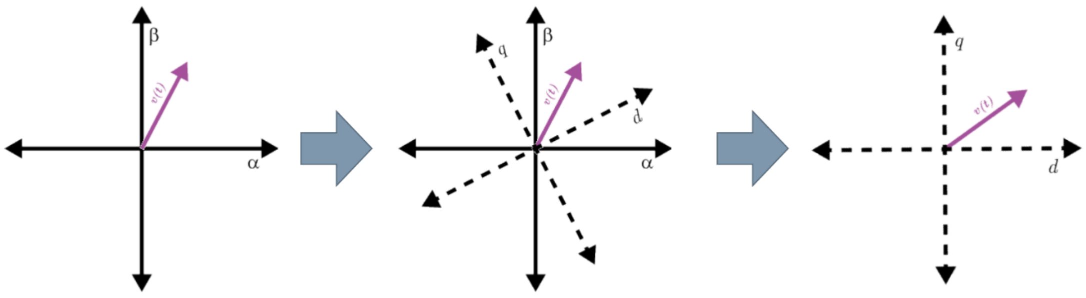


Figure C.1 Effect of the Park transformation. Image from [62]

The Park transformation is given by:

$$\begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_\alpha \\ V_\beta \\ V_\gamma \end{bmatrix} \tag{3}$$

The inverse transform is given by:

$$\begin{bmatrix} V_\alpha \\ V_\beta \\ V_\gamma \end{bmatrix} = \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} \tag{4}$$

If (1) and (3), and $(1)(2)$ and (4) are combined, the DQ0 transformation is obtained as:

$$\begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{5}$$

The inverse transform is given by:

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \frac{2}{3} \cdot \begin{bmatrix} 1 & 0 & 1 \\ -\frac{1}{2} & \frac{\sqrt{3}}{2} & 1 \\ -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta) & \sin(\theta) & 0 \\ -\sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} \tag{6}$$

In Figure C.2, the estimated DQ0-phasor magnitude is compared to the PMU measurement block from Simulink. Two things can be noticed.

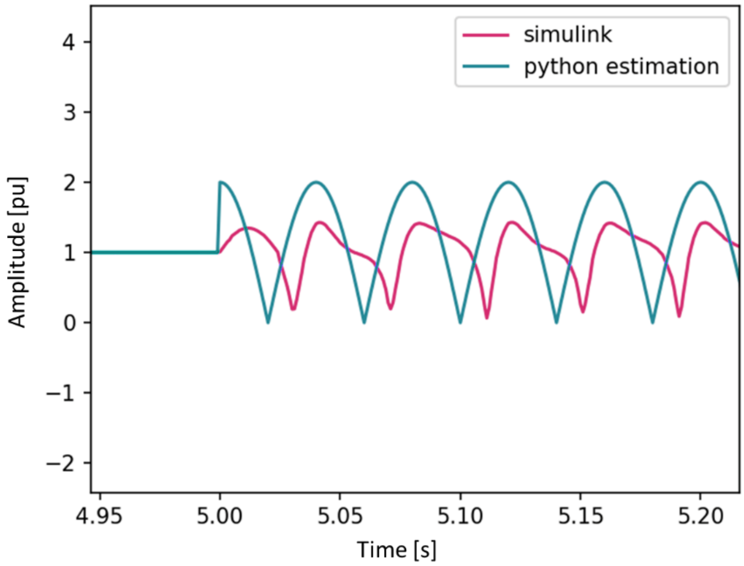


Figure C.2 Comparison of the DQ0 phasor with Simulink's three-phase PMU measurement block.

First, the implementation in python suddenly jumps while the Simulink implementation is slowly increasing. This is because the PMU measurement block uses a moving average, adding some delay to the estimated phasor. This also explains the slight difference in magnitude. This difference is not so significant as there are still oscillations visible in the phasor, allowing features 1 and 2 to work correctly. Feature 3 should work even better on the unaveraged phasor as the change in magnitude between two subsequent samples is larger. Feature 4 is unaffected as it is not computed on the phasor.

The second difference is that the PMU measurement block of Simulink uses a Phase Lock Loop (PLL) to lock the $\mathrm{dq-axes}$ to the actual grid frequency f_{grid}; the python implementation assumes $f_{grid} = 50\ \mathrm{Hz}$. The grid frequency is used to estimate the angle θ when computing the Park transformation. If there is a mismatch between the guessed frequency and the actual frequency, the vector $\boldsymbol{v}(\boldsymbol{t})$ will not be stationary anymore but will rotate. This will not affect the magnitude but rather the phase. Features 1 – 3 are unaffected by this as they rely on the phasor magnitude, not the phase.

Since the feature computation is unaffected by these differences, it can be concluded that the implemented phasor computation is good enough.

Appendix B: Defugging features

This appendix shows how to approach debugging the features when they do not produce a distinct range of values between SSO and non-SSO regions.

Feature 1

Feature 1 counts and compares the number of points in regions with upward/constant and downward trends. Depending on the type of source for the measurements, the default values might not work as expected. For example, Figure B.3 shows the values of feature 1 when computed on a synchro phasor measurement instead of the DQ0-phasor. Note that this is the same simulation as Figure 19. There is no distinction between the region containing SSOs and the non-SSO region. This is because the standard parameters ($\varepsilon_{c-significant} = 9$ and $\varepsilon_{c-similar} = 6$) are for a source with a higher sample rate (1000 Hz), while the synchro phasor is computed using 100 Hz.

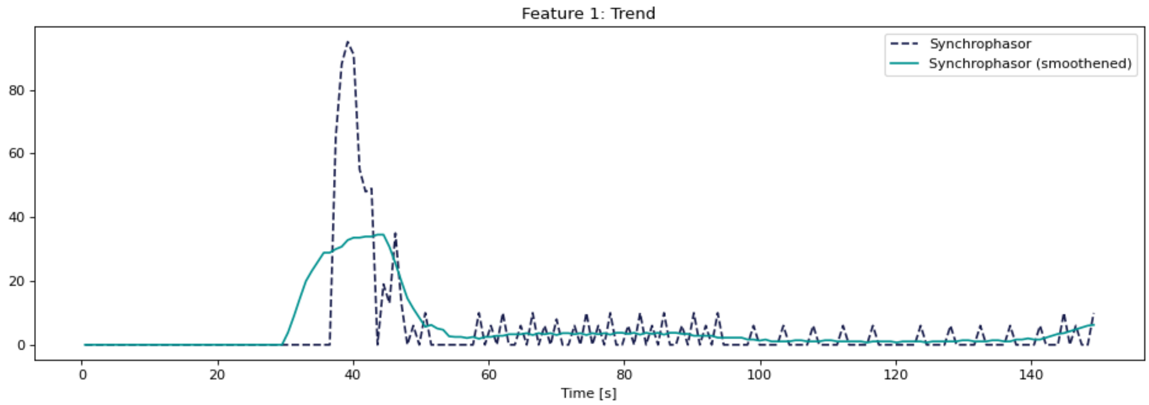


Figure B.3 Untuned feature 1 for a synchrophasor measurement. Created with data from [20].

After zooming in on the region of interest with the debug plot, Figure B.4 is obtained. The thing that is most likely causing the low value of feature 1 is that the values of $\boldsymbol{c}$ (i.e. the number of consecutive points that have the same trend) are low. Moreover, there are a lot of $1 \leftrightarrow -1$ sequences present in the signal. This is because the low sample rate of the signal results in the sampled points being constantly from different regions (i.e. upward/constant or downward trends). This is also confirmed by the derivative, which is close to 0 all the time, indicating minima or maxima.

A higher sample rate is preferred to improve the accuracy of feature 1 because having more sample points makes the trends more visible. If the time measurement is available and has a higher sample rate than the frequency measurement, it is possible to compute the DQ0-phasor and use that one for computing feature 1.

Another thing to try is removing the $1 \leftrightarrow -1$ sequence. In an SSO region, there are fewer true peaks and valleys (see also Section 7.3.3). This means there are more $1 \leftrightarrow -1$ points between two significant extremes. The peak detection and sequence-removal algorithm will increment the significant peaks more than in non-SSO regions because the number of removed points is larger. This could be sufficient to produce a clear distinction between the regions. To check this, $\varepsilon_{c-significant}$ is lowered to 2; also, $\varepsilon_{c-similar}$ is put to 1. Finally, the threshold for considering the derivative equal to 0, $\varepsilon_{gradient}$, is lowered to 0.005 as this is slightly lower than the significant peaks. This means the sequences are only removed for points with a lower derivative than the peaks.

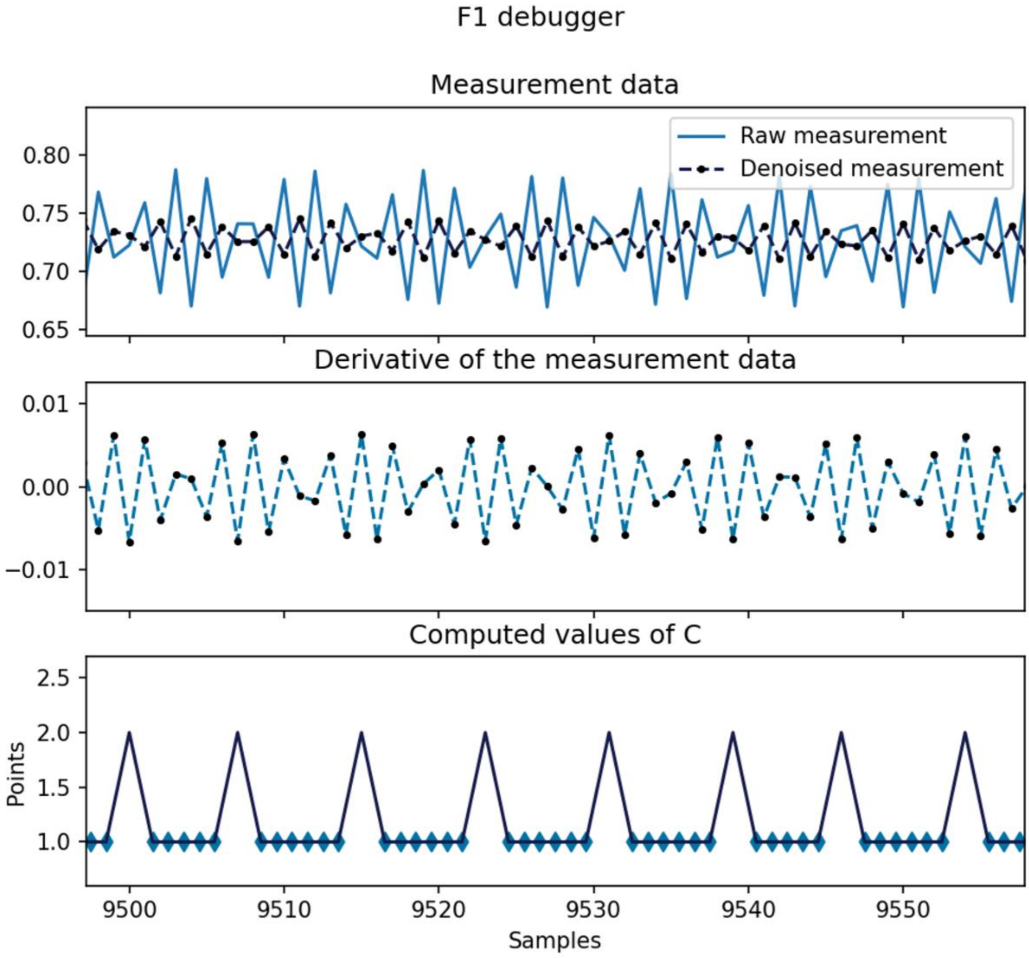


Figure B.4 Debug plot for untuned feature 1 zoomed in on the SSO region. Created with data from [20].

The result is seen in Figure B.5 where the SSO range now has a slightly different value than the non-SSO region. Note: This approach is discouraged as the data source contains one sample per region of upward/downward trends. It is better to use a data source with a higher sampling rate.

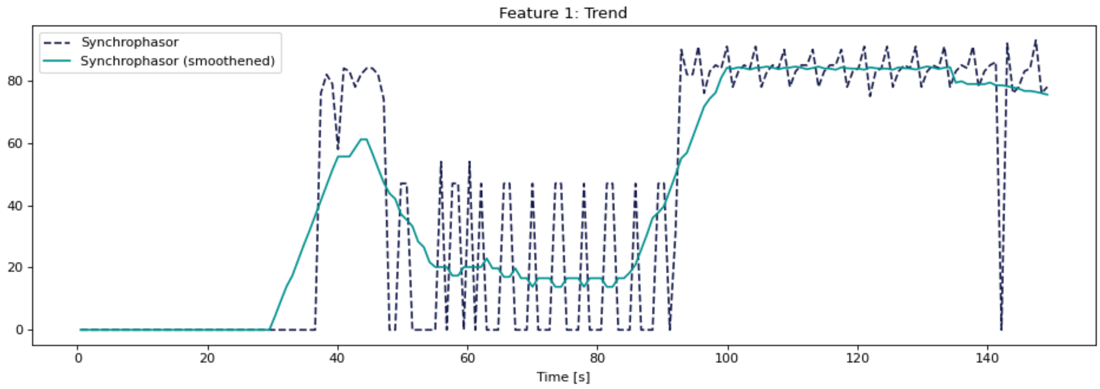


Figure B.5 Tuned feature 1 for a synchrophasor measurement. Created with data from [20].

A second example is seen in Figure B.6 where two SSOs are imposed on a grid source. The first SSO occurs from 5 to 7.5 seconds and the second SSO is between 6.5 and 9 seconds.

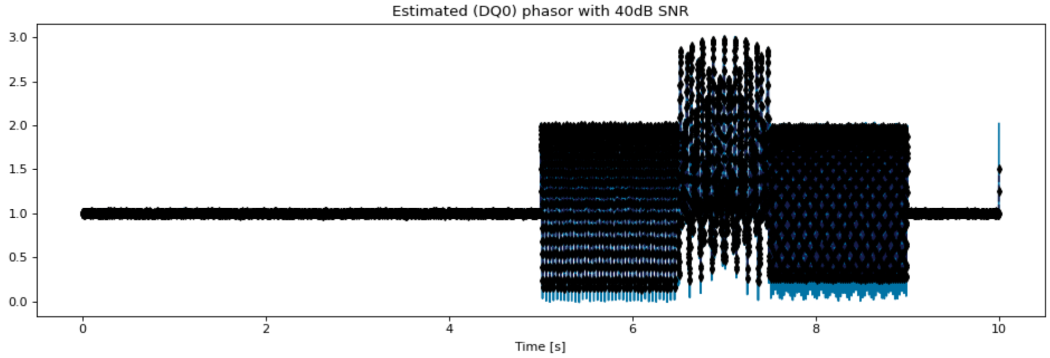


Figure B.6 DQ0-phasor of two SSOs superimposed on a constant grid frequency.

If feature 1 is computed without any improvements (significance thresholds, sequence removal and comparison thresholds), the obtained values are:

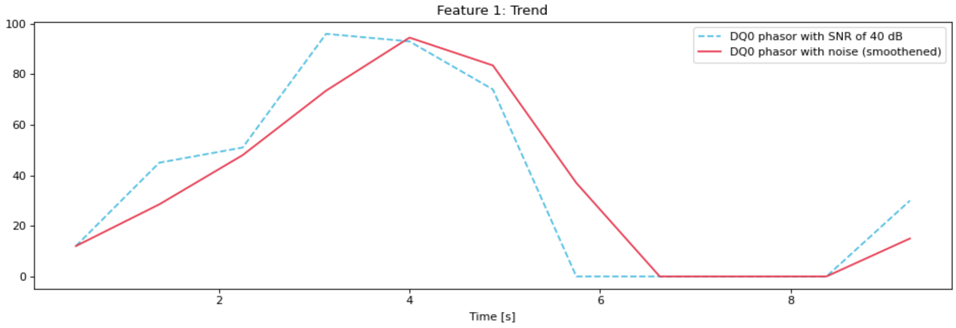


Figure B.7 Untuned feature 1 computed from the DQ0-phasor.

Feature 1 does not work as expected because its value is high when there is no SSO and low during the SSOs. This situation can occur when the default parameters of feature 1 are not correctly tuned for the measurement source. To fix this, let us look at the values in $\boldsymbol{C}$ using the debug plot:

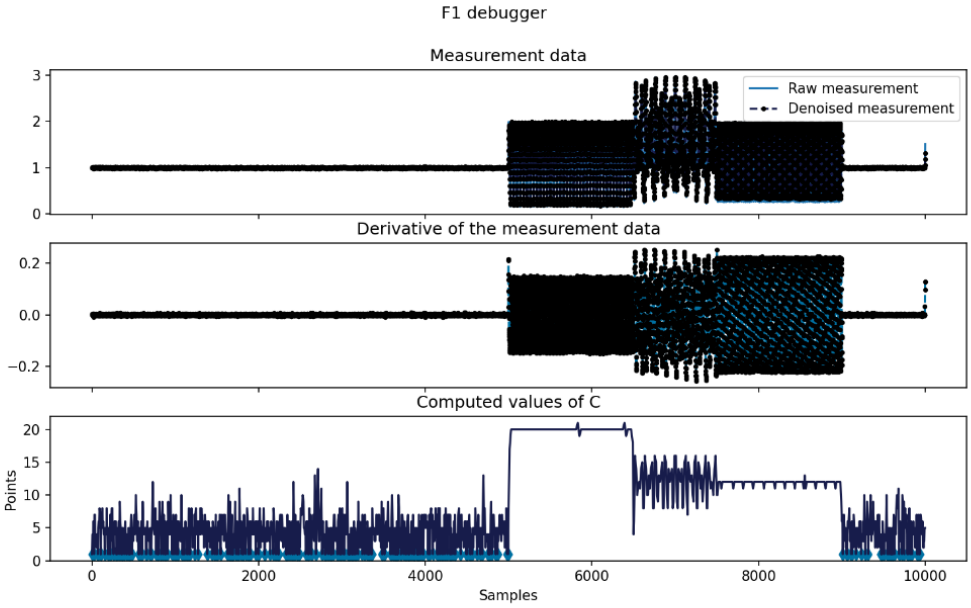


Figure B.8 Debugger plot of the untuned feature 1 computed from the DQ0-phasor.

It can be seen from Figure B.8 that there is a clear distinction between the SSO and non-SSO region. This is seen from the computed values of $\boldsymbol{C}$. The previous computation of feature 1 fails because there are some mismatches between the values of C, and it considers sequences in the ambient data. This can be fixed by setting the appropriate thresholds. $\varepsilon_{c-significant}$ is set to 9 as all ambient data points are below that value and all points in the SSO region are above the value.

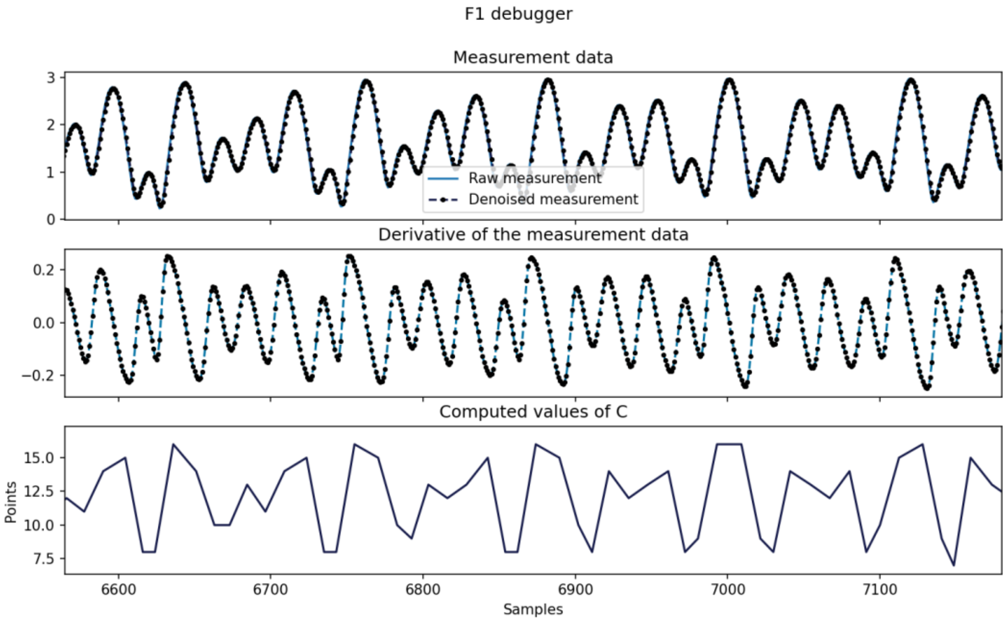


Figure B.9 Debugger plot of (untuned) feature 1 zoomed in on the SSO region.

We can set the similarity threshold by zooming in on the SSO region. It is visible from Figure B.9 that there is a trend in the phasor, but there is a slight variation in the values. By choosing $\varepsilon_{c-similar} = 3$, it is possible to find these trends.

Finally, the threshold for considering the derivative equal to 0, $\varepsilon_{gradient}$, is kept at the default 0.05 as this creates a good distinction between the derivative at the peaks/valleys and the rest of the signal.

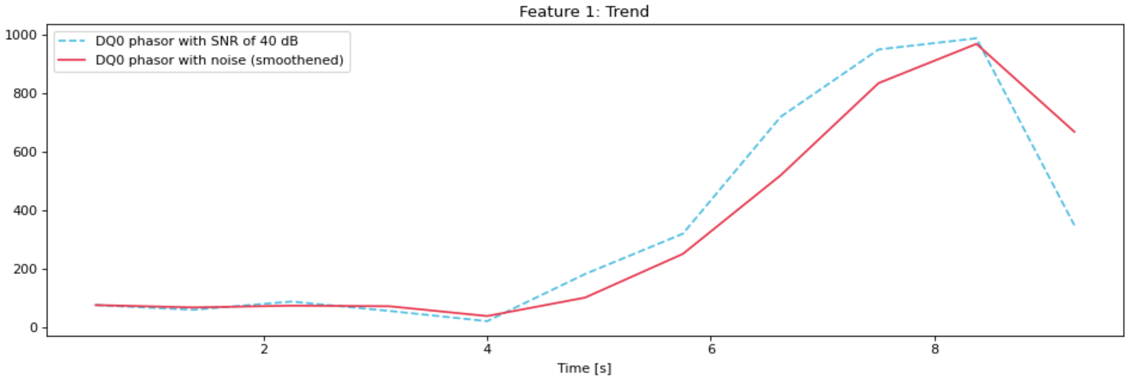


Figure B.10 Tuned feature 1 computed from the DQ0-phasor.

In Figure B.10, the result of the tuning is shown. It can be seen that feature 1 is now indeed showing a high value for the SSO region and a low value for the non-SSO region, which is as expected.

Feature 2

Feature 2 requires almost no debugging, as increasing the duration of the segments solves the undefined variances. When enabling its debugging, plots are returned for the regions that result in $F2 = \mathrm{NaN}$. For example, the signal in Figure B.11 will give undefined values for feature 2 when using a segment duration of 1 s. It is possible to evaluate using the debug pots whether the behaviour is expected (i.e. a monotonous region) or caused by some other issue. In this case, the change in phasor magnitude is probably caused by the initialisation of a simulation, making the measurement invalid. Hence, discarding the segments containing this region is better as they do not depict the data accurately.

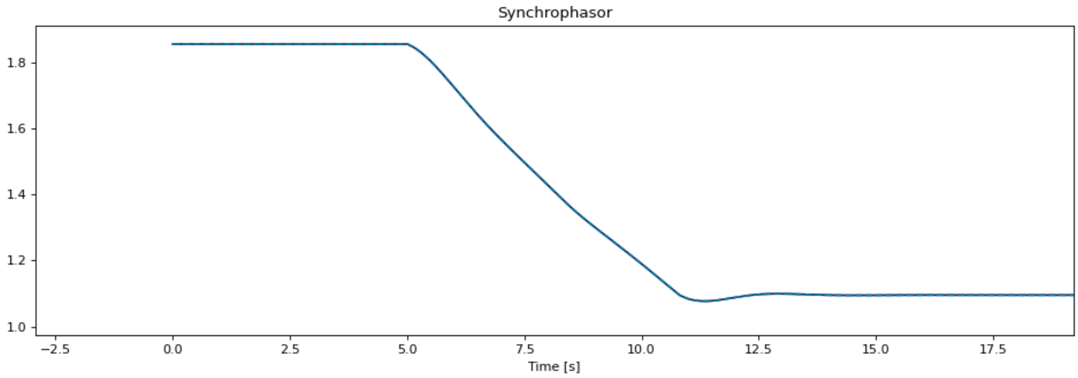


Figure B.11 A too-small segmentation of the synchrophasor measurement results in no peaks and valleys. Created with data from [20].

Feature 3

As mentioned in Section 7.3.4, the threshold ε_{LSS} should be set correctly for each source. Consider the phasor of Figure B.6. With the default $\varepsilon_{LSS} = 1$, the computed values of feature 3 are:

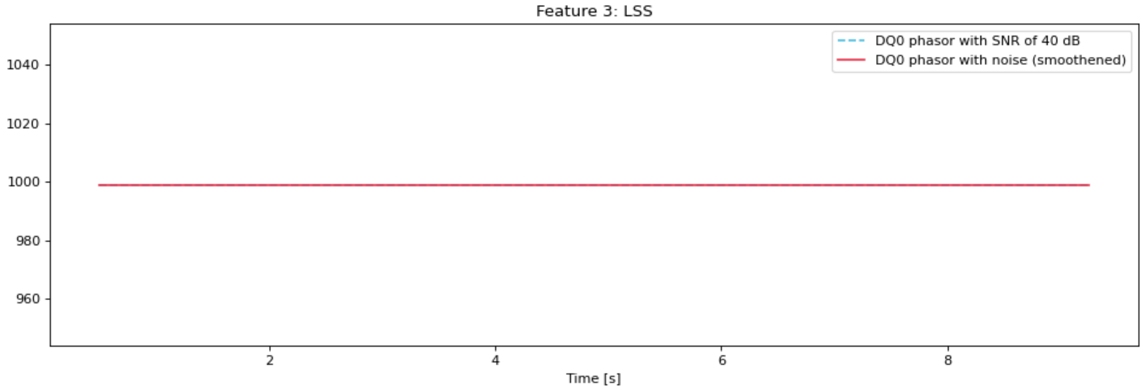


Figure B.12 Feature 3 when the threshold ε_{LSS} is not tuned for the particular measurement source.

In Figure B.12, feature 3 shows no distinction between the SSO and non-SSO regions. Upon inspecting the debugger plot (see Figure B.13), it is seen that the default threshold is too high; both regions are below the chosen threshold.

To tune the feature, ε_{LSS} is set to 0.05, which is between the values of the non-SSO region and those of the SSO region. The result is seen in Figure B.14, where there is an apparent dip in feature 3 during the SSO event.

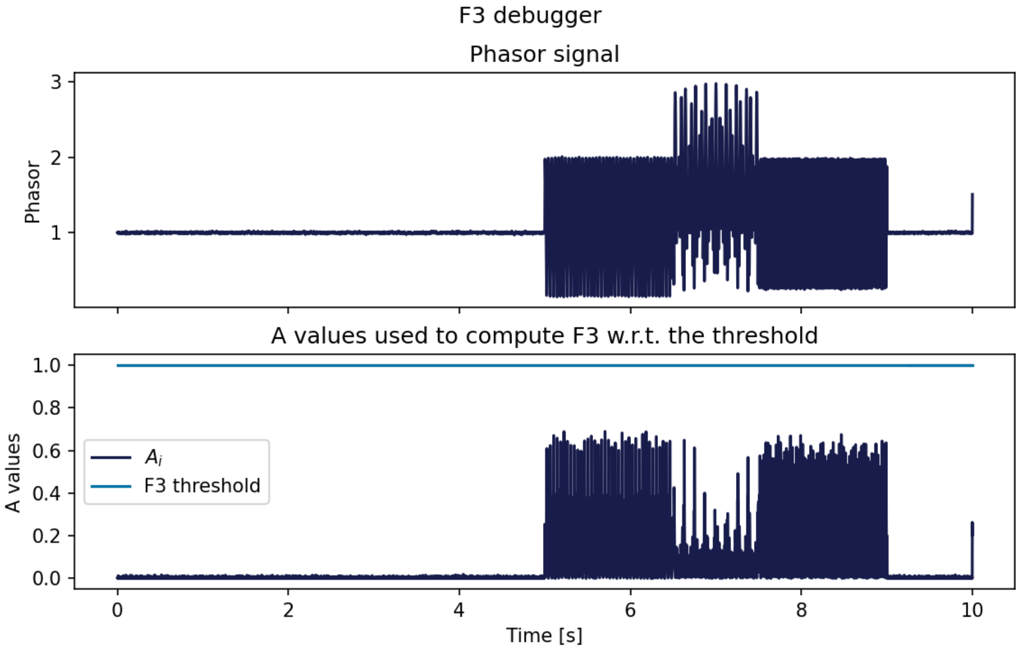


Figure B.13 Debugger plot of feature 3.

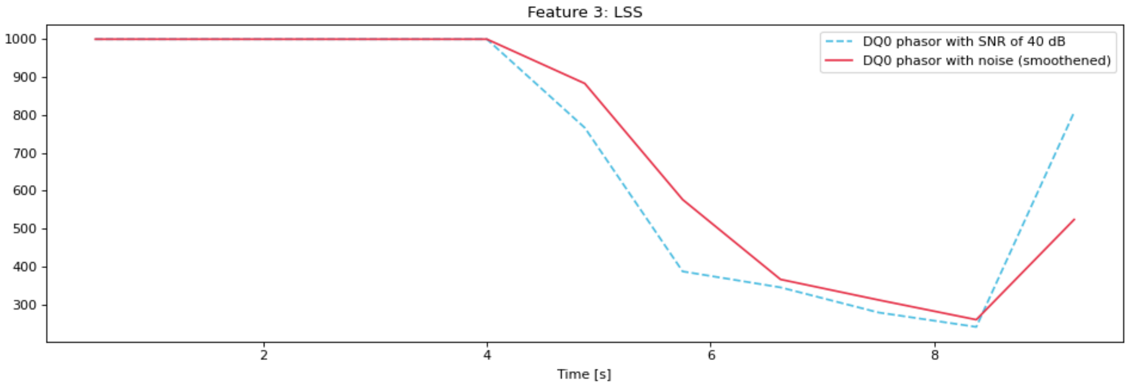


Figure B.14 Feature 3 when the threshold ε_{LSS} is correctly chosen.

Feature 4

Feature 4 should require little tuning to perform well because the underlying method (FFT) does not require any parameter tuning. When feature 4 does not perform as expected, it is probably caused by failing to find the correct peaks in the spectrum.

Figure B.15 shows the periodogram of a signal where the safety margin of the side lobe (SLM) is too low (0 dB). The peak detection algorithm will also pick up some of the side lobes caused by spectral leakage of the peak at f_{grid}. To fix this, the SLM is increased to 5 dB, effectively rejecting the side lobes (see Figure B.16).

It is important to note that the higher the SLM, the more rejection there is against noise and side lobes of the windowing function. However, this means that frequencies with lower energy content are also rejected. Unless the SLM is unreasonably high, this should not be an issue as the lower energy content frequencies are otherwise masked by the spectral leakage. In such a case, it is better to use a windowing function with lower side lobes (such as the Blackman window); there will be a trade-off in the MLW.

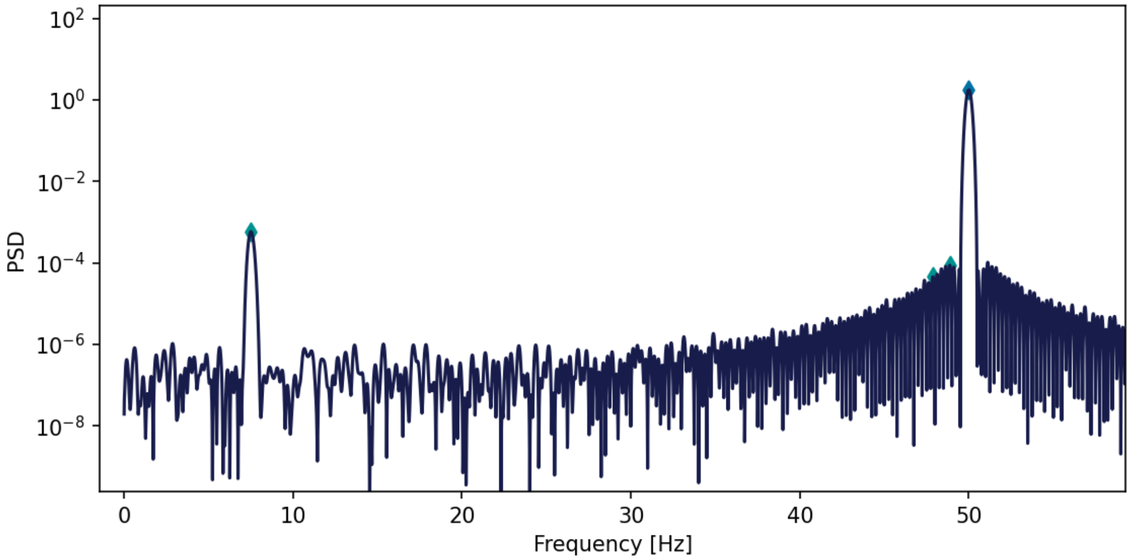


Figure B.15 Feature 4 when the threshold SLM is too low. Created with data from [20].

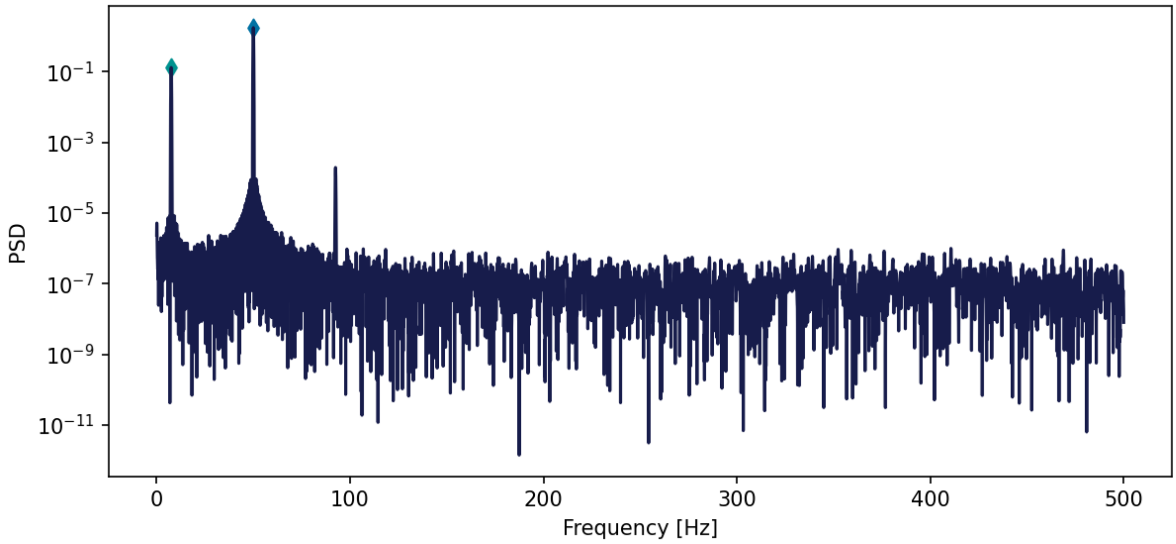


Figure B.16 Feature 4 when the threshold SLM is well-chosen. Created with data from [20].

If feature 4 fails to capture a frequency, it is also possible that the segment duration is too short. It is crucial to have at least one period of the lowest frequency present in the measurement. An intuitive explanation is that until an entire period is measured, it is impossible to know the signal's frequency. After all, the signal could consist of two frequencies and fulfil its period later. Because the lowest possible frequency is 5 Hz, only 0.2 seconds are needed, which is satisfied with $f_{min-res} = 1\ \mathrm{Hz}$. However, this must be kept in mind if a different frequency resolution is desired. It was found that two periods yield a better estimation.

A final consideration is that using a longer measurement when computing feature 4 could fail to detect SSO frequencies. For example, consider an SSO with a duration of 1 second inside a 10-second segment. Even if the SSO has the same power as the grid component, its PSD will be 1/10th of that at f_{grid}. This is because using longer durations averages out the energy of the frequency components. In those cases, a shorter segment should be used. This could increase $f_{min-res}$, but feature 4 would still find a peak, meaning its value is non-zero, thus, creating a distinction between an SSO and non-SSO event.

Appendix C: Cigre benchmark model

CIGRE WG on *Multi-frequency stability of converter-based modern power systems* has developed a benchmark system to study SSO phenomenon. In [11, 48], various methods are described that can identify and possibly mitigate frequency instabilities in converter-based power systems. The Cigre benchmark model is used in MATLAB Simulink to investigate and validate the effect of the proposed methods.

Overview

The benchmark system is depicted in Figure 35. It consists of two identical Wind Turbine Generator (WTG) strings, and inverters connect both to the grid. There are several cable segments which can cause interactions with the PE. The grid is modelled as a voltage source with a grid impedance.

Two different implementations of the test bench are provided: case 1 and case 2. Below, some of the differences are discussed.

General controller architecture

In the first model, the angle used for the dq0-transformation is computed using a PLL. In the second model, the angle is presumed to be from a constant sinusoid at the grid frequency.

Also, in the first case, the (AC) voltage is not regulated in a separate block. In the second case, a separate PI controller exists for this purpose. The second case probably implements a grid-forming controller, while the first models a grid-following controller [63]. An overview of the grid-following controller is seen in Figure D.17 The implemented grid-following controller of the first case. Image from [11].

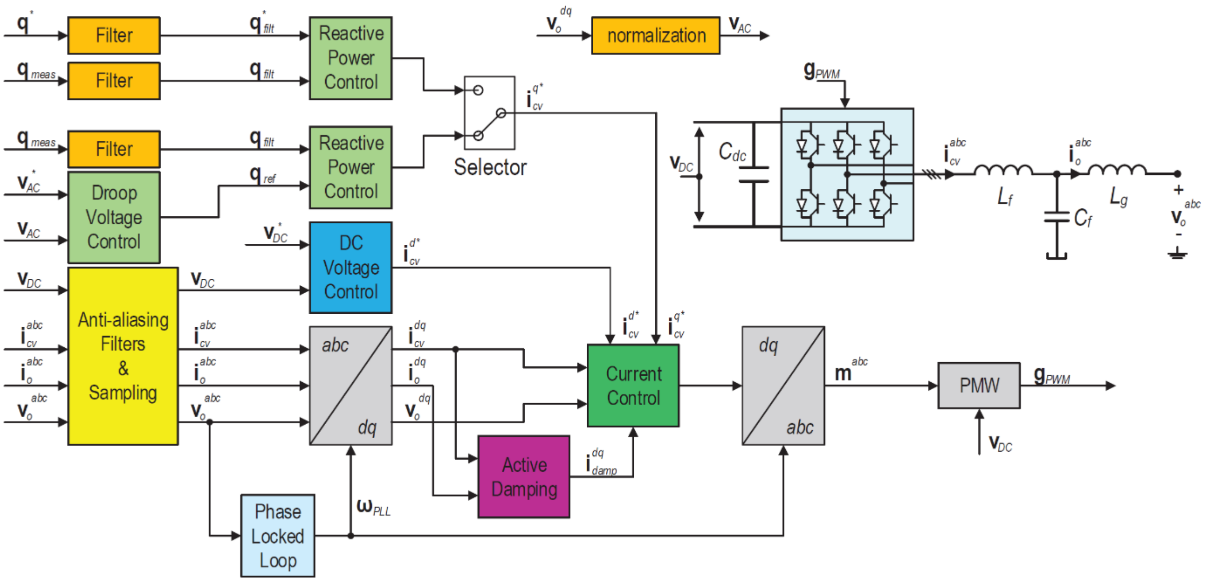


Figure D.17 The implemented grid-following controller of the first case. Image from [11].

The reactive power control loop

The reactive power setpoint is computed using the voltage droop controller and the measured reactive power in the first model. In the second case, a setpoint is given and implemented.

The method is not important for the creation of oscillations. However, case 1 allows defining the setpoints from a MATLAB script, while in case 2, the setpoints must be set inside Simulink. This makes case 2 unattractive for the simulation of multiple scenarios.

The current control loop

The current control loop is used to implement the desired power outputs (setpoints in i_{dq}) in the system. These setpoints are computed from the AC-voltage droop controller and the DC-voltage (1st model) or directly from PQ setpoints passed through a voltage regulator (2nd model).

The current control loop in the first model has a variable gain K_{ffv} which can be changed to move the poles into a stable region (see the section below). In the second case, this parameter is absent, meaning an equivalent gain of $K_{ffv} = 1$. This should result in unstable operation.

Also, the difference between the measured current and the current setpoint is handled differently. In the first case, it is passed through a PI controller; in the second case, a P controller is used.

Moreover, in the first model, the difference in current over the transformer is passed through the active damping gain and used as well in the computation of the reference *dq*-voltage. In the second model, this is done only over i_{cv} (the current output at the WTG string).

Finally, a decoupled i_{cv} is used in the first model to compensate for the interaction between the axes. If properly implemented, this should reduce the effect that a change on one axis has on the other axis by predicting the interaction and subtracting it from the other. Note: this only works well under balanced conditions. If the system is imbalanced, some coupling will occur. This is not done in the second model.

The output stage

An output stage converts the computed dq0-voltage setpoint into an AC voltage by computing a duty cycle m for the PWM modulator.

The DC power control loop

In the first model, the DC side is implemented with a variable power output which is given by a step function passed through the transfer function $H(s) = \frac{1}{0.5s+1}$ and converted back into a current setpoint. In the second model, the DC side is modelled as a constant voltage source. This difference in implementation means changing the DC side power is only possible in the first model.

Conclusions

The differences in implementation suggest that case 1 is a better model as the control architecture is more complex. Moreover, more events can be added, and they can be added externally. This makes case 1 more attractive for generating scenarios and measurements.

Oscillatory modes

The testbed is at its *"stability boundary, i.e. the system is poorly damped, and even small parameter variations can lead to instability"* [48]. This is also seen from the eigenvalue analysis, showing multiple undamped oscillatory modes (see Figure D.18). There are a couple of unstable high-frequency modes and some low-frequency modes. The latter is in the SSO range.

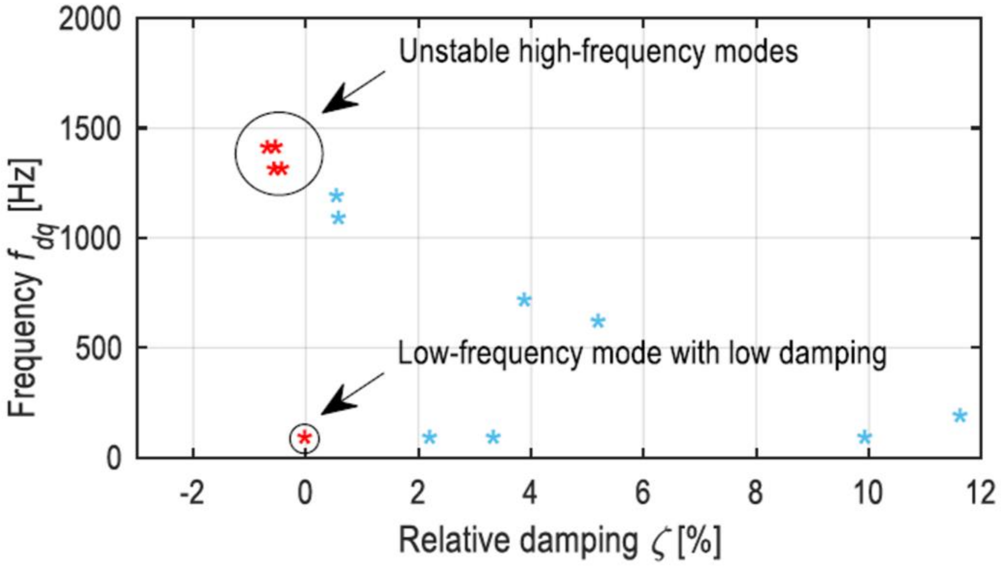


Figure D.18 Eigenvalue analysis for the base case configuration of the CIGRE test bench. Image from [48].

It is possible to mitigate some of these modes by changing system parameters. The current control gain K_{pc} is located inside the PI controller of the current controller of the control system. It multiplies the difference between the reference current from the power control block and the actual output current with this gain to find (part of) the output *dq*-reference voltage. This means that this part ensures that the reference setpoint of the power control loop is satisfied. This parameter has a limited effect on the oscillatory modes.

The voltage feedforward gain K_{ffv} is located inside the current controller of the control system. It multiplies the voltage at the output of the WF (after the transformer) with this gain to find (part of) the output *dq*-reference voltage. This term is used to ensure the desired voltage after the transformer. Decreasing this parameter from 1 to 0.7 p.u. damps the oscillations at 1271 and 1370 Hz (see Figure D.19).

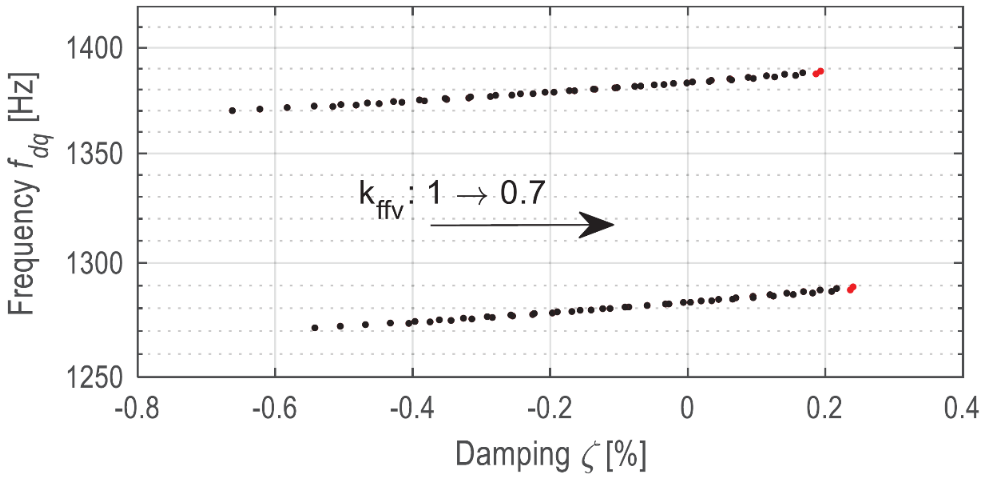


Figure D.19 Parameter sweep of K_{ffv}. Image from [48].

The active damping gain K_{AD} is located inside the current controller of the control system. It multiplies the difference in current over the transformer with this gain to find (part of) the output dq-reference voltage. This term is used to filter the harmonics across the transformer actively. Increasing this parameter stabilises the higher-order oscillatory modes. Increasing this parameter from 0.2 to 0.5 pu successfully damps the oscillatory modes at 1271, 1273, 1370 and 1372 Hz (see Figure D.20).

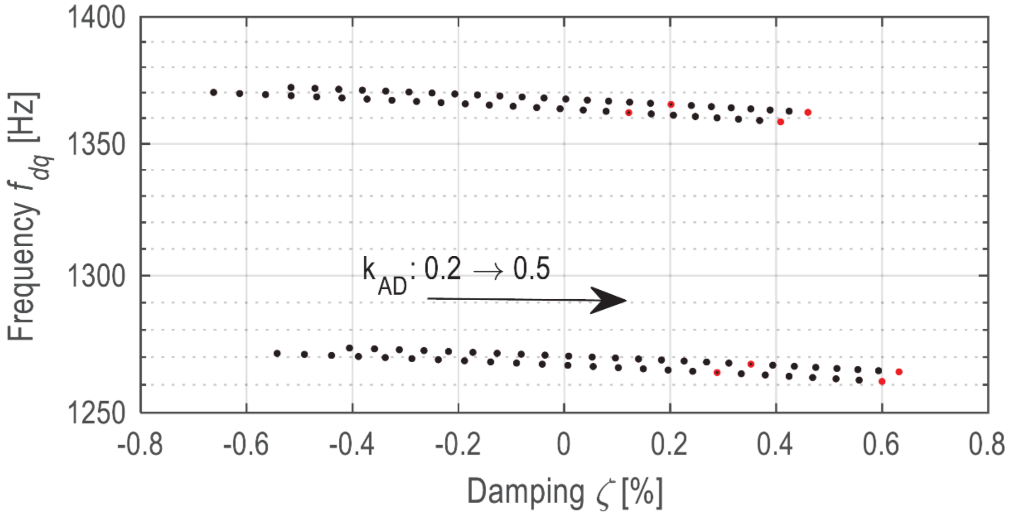


Figure D.20 Parameter sweep of K_{AD}. Image from [48].

The strength of the grid influences the oscillatory modes in the SSO range. A weak grid, e.g., $S_n = 600 \text{ MVA}$, has a Short-Circuit Ratio (SCR) of 1.43. On the other hand, a strong grid with $S_n = 3000 \text{ MVA}$ has an SCR of 7.14.

In Figure D.21, the impact of the SCR on the lower frequency modes is seen. The DC voltage control loop causes the top SSO, while the PLL is responsible for the lower mode [48]. The modes become damped when the SCR increases.

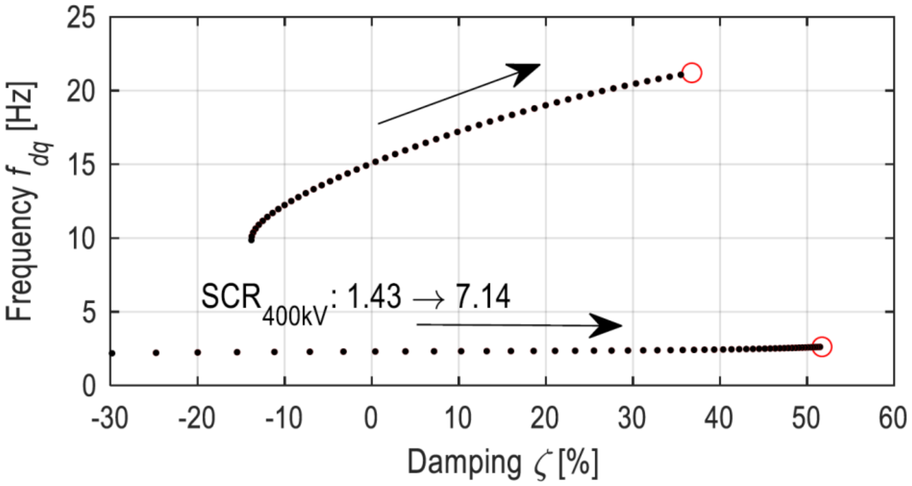


Figure D.21 Parameter sweep of the grid's SCR. Image from [48].

Unfortunately, changing the control parameters did not seem to reduce the oscillations, as suggested in [48]. Possibly, the sweep was done on a different model. Alternatively, the effect of varying the parameters is derived from the system's state-space representation without it being reflected by the model. Consequently, it is impossible to generate only SSO events from this model. A low-pass filter could be applied to isolate the low frequencies from the measurements.

Changing the SCR did show the expected results, showing no oscillations at high SCRs (Figure D.22), while the spectrum is full of them at lower SCRs. This behaviour can generate slightly different scenarios and measurements to construct the automatic SSO detector.

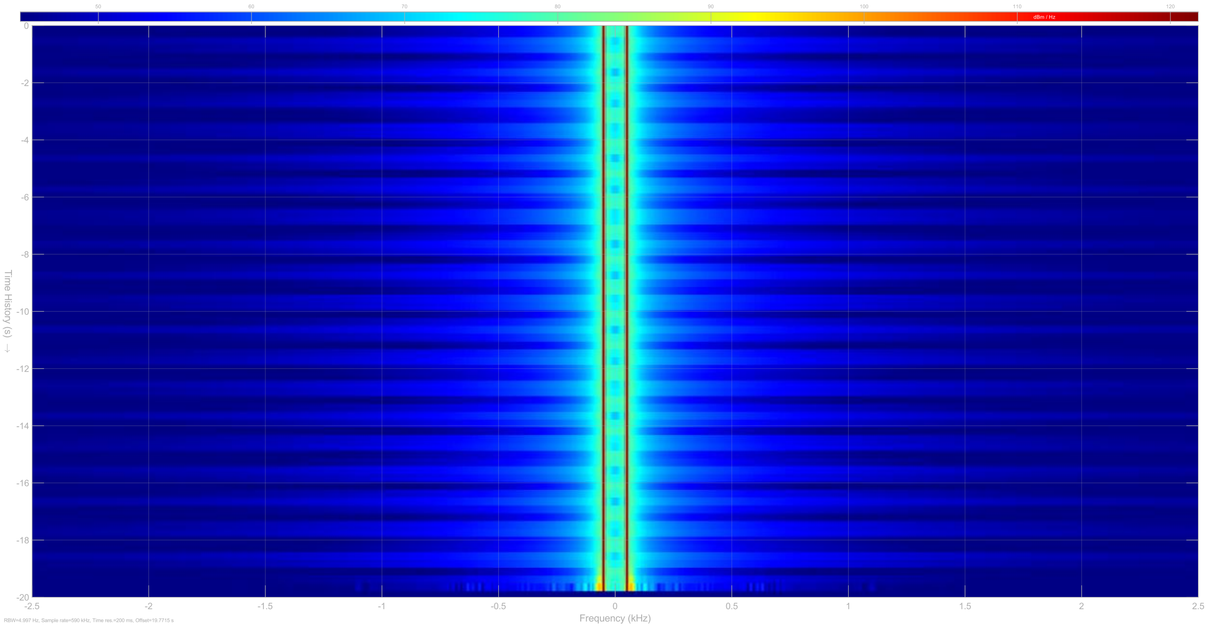


Figure D.22 Spectrogram of the voltage at POC for $S_n = 3000\ \text{MVA}$.

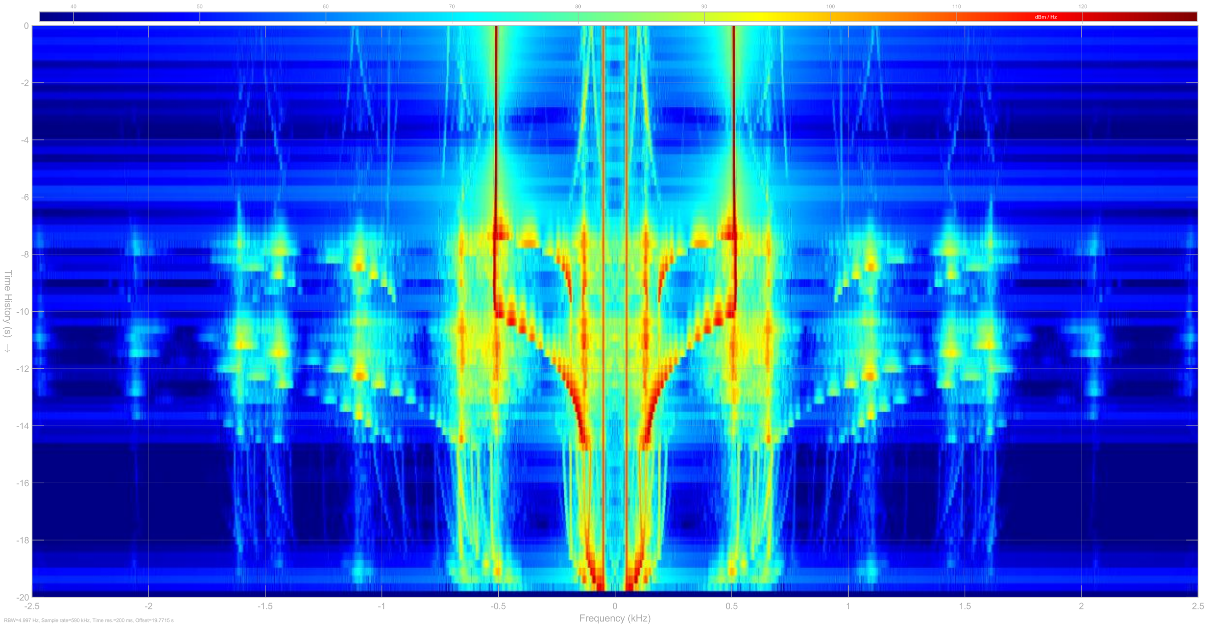


Figure D.23 Spectrogram of the voltage at POC for $S_n = 900\ \text{MVA}$.

Events

There are several events added to the test bench which excite the oscillations. Initially, the system is marginally stable, and events require the network to move towards a different state. Because the system is hardly stable, even small perturbations are sufficient to induce oscillations.

There are four possible events: a step change in the grid voltage, WF power output change, AC voltage reference change and DC voltage reference change. Their meaning is discussed below.

Step in the grid voltage

A step in the grid voltage indicates an event on the grid side. The grid-following converters will try to follow this voltage level.

Step in the WF power

The power output of the WF can change. This change can be natural when the availability of wind causes it. However, the grid operator can also request an amount of active power to be delivered. Some turbines may be turned off, and the setpoint changed for the other turbines to satisfy the operational point. The mechanical power also needs to be adjusted to reflect this new setpoint. Because the pitch control is slow to react, there is a temporary mismatch causing a change in voltage. The converters have to cope with the increased voltage level; if not, oscillations will occur.

In the test bench, it is possible to change the active power output of each WTG string individually.

Step change in AC voltage reference

A step in the AC voltage reference is effectively a step in the reactive power setpoint. A change in this setpoint could be caused by a transformer tap action, as required by a contingency, or to minimise transmission losses [64].

Step in the DC voltage reference

When a fault occurs on the AC side, the power output drops, but DC power persists because of the mechanical input (wind). The DC power is ramped down but slower, meaning that there is much more power produced than there can be delivered to the grid. This energy needs to go somewhere and results in the capacitors being charged, causing the DC voltage to go up. This is simulated by changing the DC voltage reference. This increased level persists until a crowbar is connected to discharge the system.

Default events

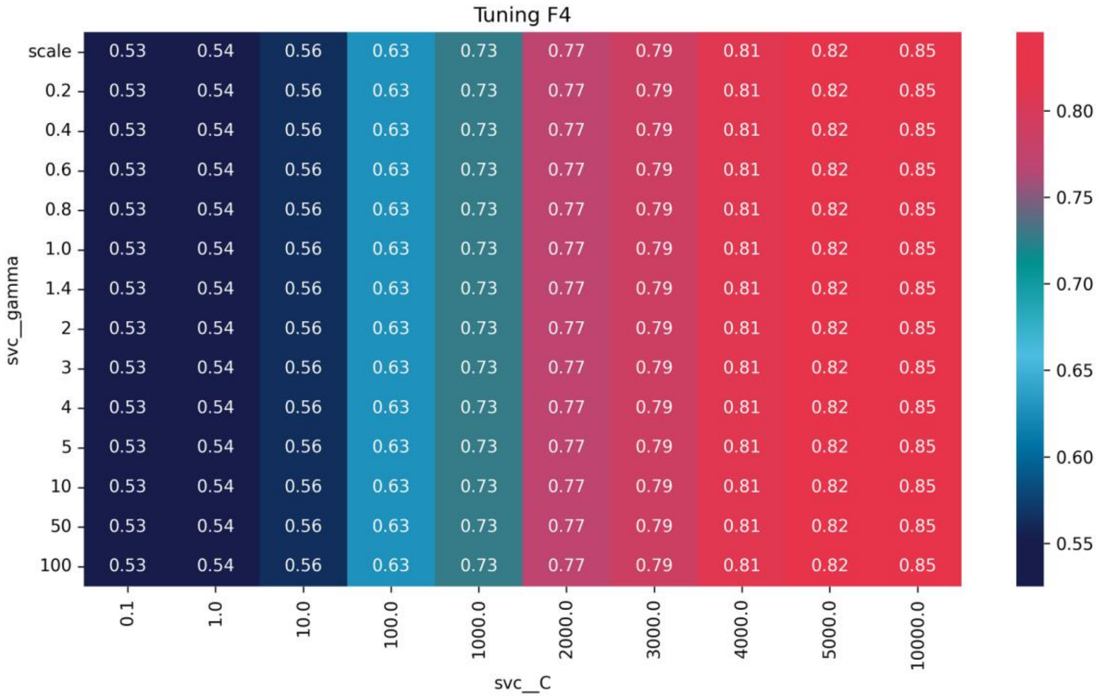
The oscillations in the test bench are excited by chaining all events together.

First, a grid step of 0.02 pu is applied at 1 second. Next, the WF power output is increased from 0 to 1 pu at 5 and 10 seconds, respectively. This corresponds to increasing the power output of each string to its nominal power. After that, the AC voltage reference is decreased from 1.05 to 1.03 pu because some tap changes are needed for power transport. This happens 12 seconds into the simulation. Finally, at 15 seconds, the DC reference voltage reference is reduced from 2 to 1.98 pu. This is because the new active and reactive power setpoints cause a temporary mismatch between the requested and delivered power.

One of the resulting spectrograms is seen in Figure D.23. It is possible to see how the various events excite the modes differently.

Appendix D: Tuning F4 classifier

This appendix contains various supporting figures for the tuning of the feature 4 classifier.



svc_gamma	0.1	1.0	10.0	100.0	1000.0	2000.0	3000.0	4000.0	5000.0	10000.0
scale	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
0.2	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
0.4	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
0.6	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
0.8	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
1.0	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
1.4	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
2	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
3	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
4	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
5	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
10	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
50	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85
100	0.53	0.54	0.56	0.63	0.73	0.77	0.79	0.81	0.82	0.85

Figure A.24 Grid search with CV for SVM with linear kernel as the classifier for feature 4. The values inside the cells represent the balanced accuracy for different parameters.

svc_gamma	0.1	1.0	10.0	100.0	1000.0	2000.0	3000.0	4000.0	5000.0	10000.0
scale	0.56	0.61	0.71	0.84	0.87	0.88	0.89	0.89	0.9	0.91
0.2	0.5	0.51	0.55	0.6	0.7	0.72	0.74	0.75	0.77	0.81
0.4	0.52	0.53	0.56	0.62	0.72	0.75	0.78	0.79	0.81	0.83
0.6	0.52	0.54	0.56	0.63	0.74	0.78	0.8	0.82	0.83	0.85
0.8	0.51	0.54	0.57	0.64	0.75	0.79	0.82	0.83	0.83	0.85
1.0	0.51	0.55	0.57	0.65	0.77	0.81	0.83	0.83	0.84	0.86
1.4	0.51	0.55	0.59	0.66	0.78	0.83	0.83	0.85	0.85	0.86
2	0.51	0.55	0.6	0.7	0.81	0.83	0.85	0.85	0.86	0.86
3	0.55	0.56	0.62	0.72	0.83	0.85	0.86	0.86	0.86	0.87
4	0.61	0.56	0.62	0.72	0.84	0.85	0.86	0.86	0.86	0.88
5	0.57	0.56	0.63	0.73	0.85	0.86	0.86	0.86	0.86	0.88
10	0.56	0.58	0.65	0.77	0.86	0.86	0.87	0.88	0.88	0.89
50	0.57	0.63	0.73	0.85	0.88	0.89	0.9	0.9	0.91	0.92
100	0.58	0.65	0.77	0.86	0.89	0.9	0.91	0.92	0.92	0.91

Figure A.25 Grid search with CV for SVM with RBF kernel as the classifier for feature 4. The values inside the cells represent the balanced accuracy for different parameters.

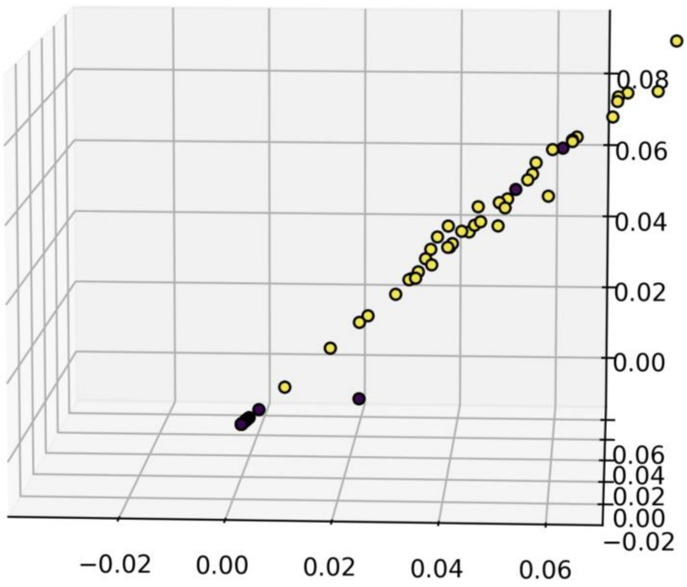


Figure A.26 Values of feature 4 for the different phases. Different colours correspond to different classes.

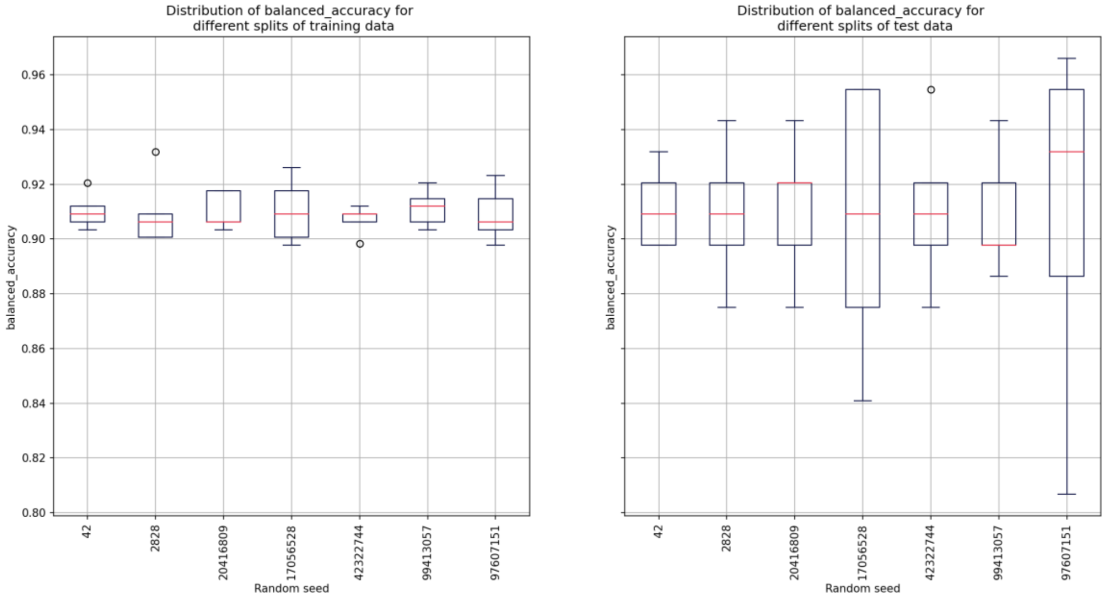


Figure A.27 Performance evaluation for SVM with RBF kernel as the classifier for feature 4 under different random seeds.

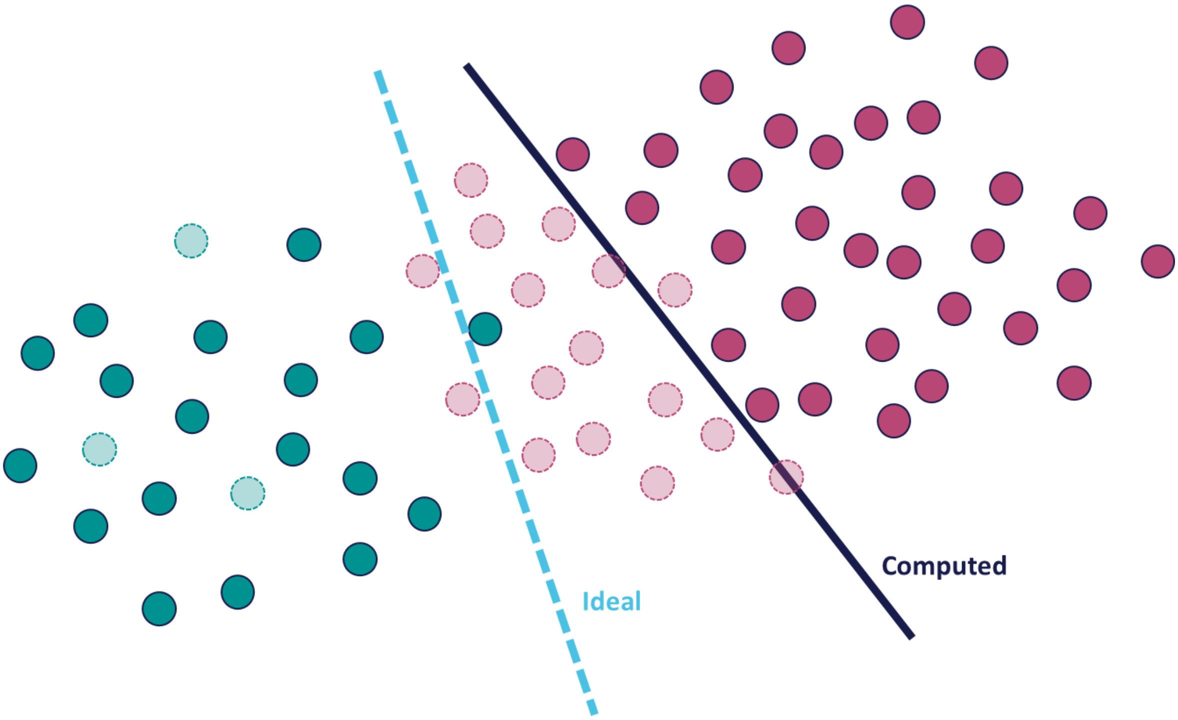
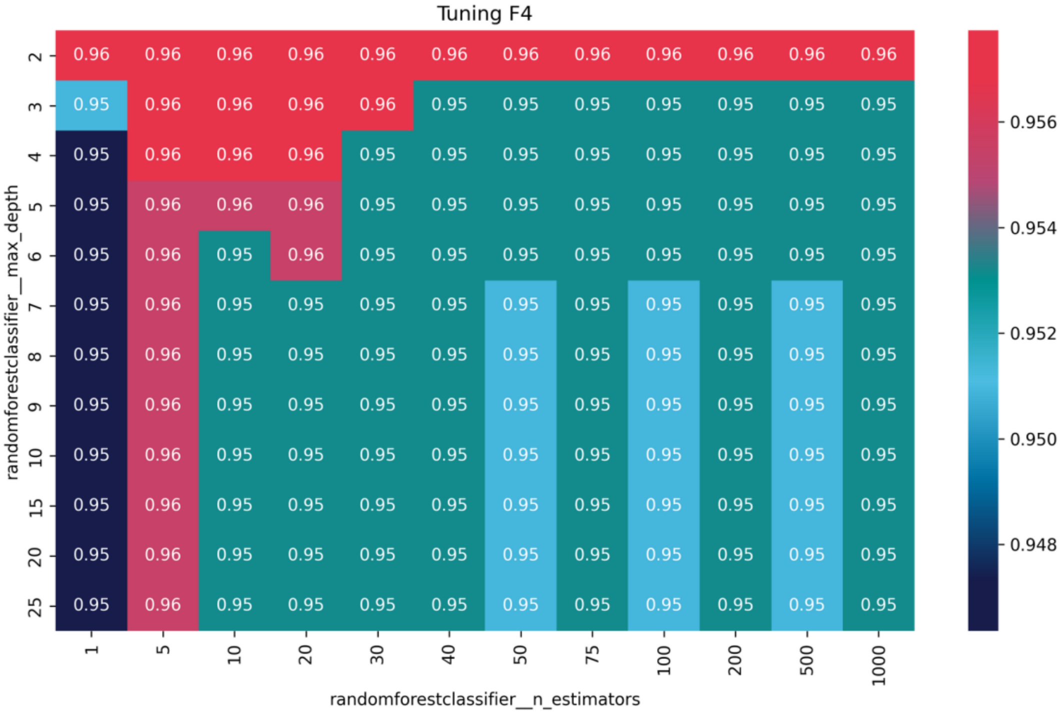


Figure A.28 Problem with SVM as the classifier for feature 4 where the ideal boundary is more to the left of the fitted boundary.



	1	5	10	20	30	40	50	75	100	200	500	1000
2	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96
3	0.95	0.96	0.96	0.96	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95
4	0.95	0.96	0.96	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
5	0.95	0.96	0.96	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
6	0.95	0.96	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
7	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
8	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
9	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
10	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
15	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
20	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95
25	0.95	0.96	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95	0.95

Figure A.29 Grid search with CV for RF as the classifier for feature 4. The values inside the cells represent the balanced accuracy for different parameters.

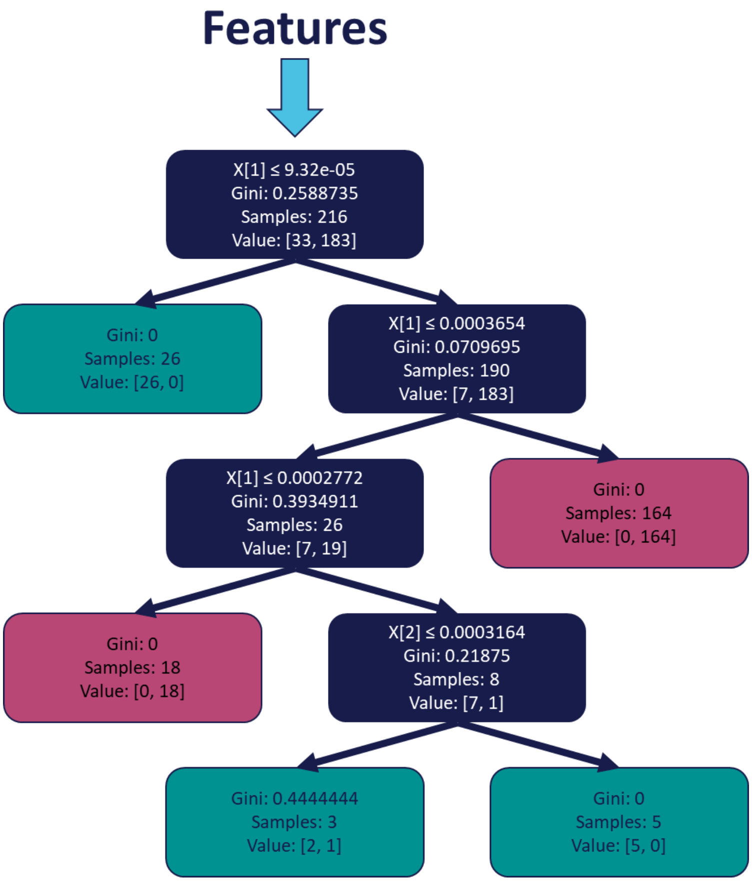


Figure A.30 Example of a fitted classifier by the RF.