NESO
National Energy
System Operator

Delivered in partnership with

tnei

WP3 Report

SSO Identification Tool

Document Reference: 15167-003-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

Document Control

Revi	Status	Prepared by	Checked	Approved	Date
D0	DRAFT	DC			11/02/2024

TNEI Services Ltd

Company Registration Number: 03891836 VAT Registration Number: 239 0146 20

Registered Address		
Bainbridge House	7th Floor West One	7th Floor
86-90 London Road	Forth Banks	80 St. Vincent Street
Manchester	Newcastle upon Tyne	Glasgow
M1 2PW	NE1 3PA	G2 5UB
Tel: +44 (0)161 233 4800	Tel: +44 (0)191 211 1400	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

Document Control 3
Contents 5
Introduction 9
1.1 Quick links 10
2 SSO Tool Architecture 11
2.1 Overview of the process 11
2.2 PSCAD Automation 15
2.2.1 User variables 15
2.2.2 Module structure 22
2.2.3 Custom PSCAD blocks 23
2.3 Frequency domain analysis 26
2.3.1 User variables 26
2.3.2 Module structure 29
2.4 Machine learning model 30
2.4.1 User variables 31
2.4.2 Module structure 32
2.5 User Interface 33
2.5.2 Select Model 34
2.5.3 Global settings 35
2.5.4 Select a simulation type 35
2.5.5 Case generator 37
2.5.6 View and run processes created by case generator 38
2.5.7 Progress update 39
2.5.8 Results view 40
3 SSO Tool Setup Guide 41
3.1 Python packages 41
3.2 PSCAD automation packages 41
3.3 Frontend packages 42
3.4 Network model setup 42
3.5 Running a sample study 44
4 Case Study 47
4.1 SSO Case – Cigre Model 47
4.1.1 Frequency scan 47

4.1.2 Frequency domain analysis ... 49
4.1.3 ML model oscillation detector ... 54
4.2 SE Coast study ... 57
3.2.1. Frequency Scan results ... 58
5 References ... 60

APPENDICES

Appendix B: Review of impedance scan methods ... 64

TABLES

Table 1 Summary of the important steps in the SSO identification framework ... 14

Table 2 List of user parameters in the impedance scanner ... 25

FIGURES

Figure 1 Overall process flow diagram of the developed framework. ... 12

Figure 2 PSCAD automation tool structure and module hierarchy ... 22

Figure 3 Wind farm control custom block to link with proprietary model MHI. ... 23

Figure 4 Wind farm custom dialog window showing available options ... 24

Figure 5 Impedance scanner block developed in the project, (a) view of the block in PSCAD, and (b) options available to the user for scanning frequency of apparatus. ... 25

Figure 6 Frequency domain analysis tool structure and module hierarchy ... 29

Figure 7 Example of a current measurement in pu showing an SSO event. Created with data from IEEE DataPort [12] ... 30

Figure 8 Machine learning based oscillation detector tool structure and module hierarchy ... 32

Figure 9: The home page or landing page for the tool interface ... 33

Figure 10: Help button for more information about how to interact with the interface ... 33

Figure 11: An automated walkthrough of the entire process ... 34

Figure 12: Select a PSCAD model file (.pscx or .pswx) ... 34

Figure 13: Different settings users can modify for a selected PSCAD file ... 35

Figure 14: Click on 'Select Simulations' in the side panel to navigate to the new case generator page 35
Figure 15: Select a simulation type from the dropdown menu 36
Figure 16: Drag individual devices to the left side to add to a new simulation case 37
Figure 17: Change device parameters as necessary to create a new case 37
Figure 18: Create a new scenario by adding a new simulation case 38
Figure 19: Click on the 'Simulations' tab in the side panel to view all cases 38
Figure 20: Run all simulation cases (Process 1, Process 2 etc) 39
Figure 21: Output window showing the progress update along with a flowchart highlighting the completed steps 39
Figure 22: Explore results within the UI in table format and plotly figures 40
Figure 23: Plotly graph showing an example plot. It is possible to add more than one plot to the a single graph for comparison 40
Figure 24 Scanner parameters for the Cigre benchmark model 43
Figure 25 External grid fault level parameters for a strong system 43
Figure 26 Enable scanner to start injecting voltage from 2 seconds 43
Figure 27 Toggle operation of PSCAD widget to change the status of WF filters, (a) switched on, (b) switched off. 44
Figure 28 Screenshot of the frequency injection measurement results from 1 Hz to 40 Hz for different scan types and different scan locations 45
Figure 29 Screenshot of the meta table from the frequency_results database showing the calculated impedance 46
Figure 30 Screenshot of the meta table from the frequency_results database showing the results from the frequency domain analysis 46
Figure 31 SSO benchmark model network diagram. Figure adapted from [13] 47
Figure 32 Apparatus scan from WF1 48
Figure 33 Grid scan from WF1 for 1200MVA external grid fault level 48
Figure 34 Whole system scan from WF1 for 1200MVA external grid fault level 49
Figure 35:Whole system scan from WF2 for 1200MVA external grid fault level 49

Figure 36 Reactance crossover calculations for (a) 1200MVA grid, measured at WF1, (b) 750MVA grid, measured at WF1, (c) 1200MVA grid, measured at WF2, and (d) 750MVA grid, measured at WF2. 51

Figure 37 Grey-box analysis calculations showing (a) whole system modes for 1200MVA grid, (b) whole system modes for 750MVA grid, (c) participation of WFs in 12.4Hz mode for 1200MVA grid, and (d) participation of WFs in 9.5Hz mode for 750MVA grid. 52

Figure 38 Impedance intersection calculations for; (a)&(c) 1200MVA grid, measured at WF1, (b)&(d) for 750MVA grid, measured at WF1. 53

Figure 39 EMT simulation results for a 60ms fault at WF1 export cable for (a) 1200MVA grid measured at WF1, (b) 750MVA grid measured at WF1, (c) 1200MVA grid measured at WF2, and (d) 750MVA grid measured at WF2. 54

Figure 40 Online prediction by the ensemble model on three PSCAD cases. The ensemble model correctly predicts in all cases. 57

Figure 41: Simplified Single-line representation of the Southeast coast GB network. 58

Figure 42: Full matrix impedance intersection results for the STATCOM and SEC network. Red trace in Zpp is the STATCOM impedance while the blue trace corresponds to the network. 59

Figure 43: Reactance crossover results obtained from the wholesystem scan . Blue trace in Reactance Zpp show the reactance of STATCOM while the red trace corresponds to network. 59

Introduction

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. In recent years, the offshore wind energy sector has witnessed a remarkable growth, with the scale of wind farms expanding exponentially. As the share of traditional generation technology has steadily declined and volume of Inverter Based Resources (IBR) have increased, including HVDC interconnectors, it has been observed that the new technologies (both control and converter design) could introduce Subsynchronous Oscillations (SSO) in the network under certain disturbance conditions which do not fall under the traditional definition of SSR.

Several events have been reported in literature since the first ever recorded event in 1970. The technical report 'Wind Energy Systems Subsynchronous Oscillations: Events and Modeling' by the IEEE Wind SSO Task Force [1] provides a succinct summary of real world SSO events and their root cause analysis. Some of the events have recorded more than one oscillation frequency and more than one phenomenon as the cause of the oscillation such as the events in Guyuan, China where between December 2012 - 2013, 58 SSO events were detected. From the reported events, it is clear that 2009 onwards, SSCI has been the predominant cause for SSO and there is a clear increase in the frequency of these events as well. The most recent reported event is in Great Britain (GB) in 2019. A technical report from National Grid ESO provides further information about the incident [2].

Understandably, SSO analysis has received a lot of attention from System Operators (SO) in recent years to understand the phenomena better and develop analysis methods and tools to prevent spurious oscillations that could potentially threaten the security of the system. This project has developed an SSO analysis framework that can be used by the SO to frequently evaluate a large volume of scenarios for potential SSO events. The framework provides a clear pathway from setting up a network model for different scenarios to isolating cases having SSO concerns.

This report is the final deliverable of WP3. The theory behind the algorithms used in the tool are not explained in detail in this report. Please refer to the WP1 and WP2 reports for an in-depth discussion on the fundamental concepts. The purpose of this report is to document the tool architecture and provide a step-by-step guide to the users of the tool. The SSO benchmark model developed by Cigre WG C4.49 for multi-frequency stability of converter-based systems is used as the test network to explain the functionality of the tool. This report serves as a user manual for the final version of the tool handed over to the ESO at the end of WP3.

1.1 Quick links

Overview of the complete tool

User variables available for PSCAD automation module

PSCAD automation module hierarchy

Frequency domain analysis module hierarchy

Machine learning model module hierarchy

SSO tool setup guide

SSO tool case study

Relevant publications

2 SSO Tool Architecture

2.1 Overview of the process

The developed framework is an end-to-end automated process, implemented as a tool with a user interface. The overall process is shown in Figure 1. It includes several important steps to process operational scenarios with the objective of eliminating scenarios that do not poses any threat of damping issues in the subsynchronous frequency range.

The first step in the process is to read the base case EMT model and provide users an interface to define several operational scenarios. These scenarios can be both topological changes in the network (such as planned outage or shunt compensation switching) and operational setpoint adjustments (such as MW and MVAr export of IBRs). Once a list of scenarios is defined, these are applied to the EMT model to create several variations of the base case model.

The next step is to create a frequency domain representation of the EMT models by injecting voltages (limited to less than 1% peak to peak for transmission connections) at certain frequencies such as 1Hz to 40Hz. Several frequency injection techniques are discussed in literature [3], [4], [5], [6] each having their strengths and weaknesses. Some techniques are based on the injection of a wide-band frequency such as white noise to capture the non-linear characteristics of IBRs while others are based on injection in bands with quadratic phase displacement. However, the most important consideration is the use of a Multi Input Multi Output (MIMO) transfer function model over a Single Input Single Output (SISO) model. If the positive and negative sequence impedances are decoupled, then the standard sequence domain impedance, calculated as per equation (1), can be used. This SISO transfer function model is valid as long as there are no active non-linear devices in the system.

$$Y_p = \frac{I_p}{V_p}, Y_n = \frac{I_n}{V_n} \qquad (1)$$

Commented [SM(1]: If we consider injections in %, then 1 kV peak line to neutral for a 400 kV L-L connection voltage would be about 0.3 %. Do we need to consider 1kV always?

Commented [DC2R1]: Good point, I have updated it to %

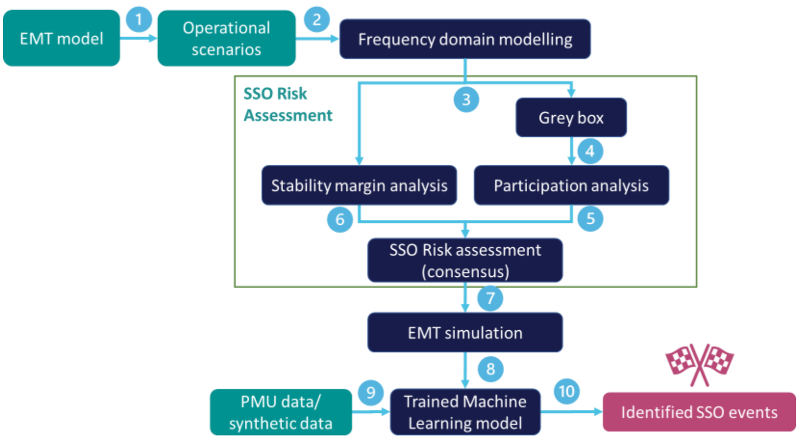


Figure 1 Overall process flow diagram of the developed framework.

The presence of elements such as PLL, current controllers with unequal structure and/or parameters in the *d*- and *q*- axes, dc-link voltage control system, active and reactive power controllers and salient pole synchronous machines introduces Mirror Frequency Coupling (MFC) [3]. This means that systems that are linear time invariant (LTI) in the *dq* domain may not be LTI in the phase domain due to coupling between sequences. It has been reported in [3] and [4] that MFC is dominant at low frequencies i.e., in the subsynchronous range and the correct calculation of the impedance transfer function has an impact on the stability conclusions. To address this limitation, a modified sequence domain impedance calculation is introduced in [3] that ensures that any LTI system in the *dq* domain is also LTI in the sequence domain. This scanning technique involves injecting a positive $(s + jw)$ and a negative $(s - jw)$ sequence perturbation to calculate the full impedance matrix $\mathbf{Y_{pn}}(\mathbf{s})$, given by equation (2).

$$\begin{bmatrix} I_p(s+jw) \\ I_n(s-jw) \end{bmatrix} = \begin{bmatrix} Y_{pp}(s) & Y_{pn}(s) \\ Y_{np}(s) & Y_{nn}(s) \end{bmatrix} \begin{bmatrix} V_p(s+jw) \\ V_n(s-jw) \end{bmatrix} = \mathbf{Y_{pn}}(\mathbf{s}) \begin{bmatrix} V_p(s+jw) \\ V_n(s-jw) \end{bmatrix} \quad (2)$$

The elements of the matrix $\mathbf{Y_{pn}}(\mathbf{s})$ can be defined as; $Y_{pp}(s)$ is calculated by using the positive sequence current $I_p(s + jw)$ induced by a positive sequence voltage $V_p(s + jw)$ at the perturbation frequency $(s + jw)$, $Y_{pn}(s)$ is calculated by the positive sequence current $I_p(s + jw)$ at the coupling frequency induced by a negative sequence voltage $V_n(s - jw)$,

$Y_{np}(s)$ is calculated by the negative sequence current $I_n(s-jw)$ at the coupling frequency induced by a positive sequence voltage $V_p(s+jw)$, $Y_{nn}(s)$ is calculated by the negative sequence current $I_n(s-jw)$ induced by a negative sequence voltage $V_n(s-jw)$ at the perturbation frequency $(s-jw)$. The modified sequence domain method is implemented in the developed framework to measure impedance of VSC converters accurately irrespective of the technology type i.e., WTG, STATCOM, HVDC. The current implementation is not tested for LCC convertors. This will be explored in future work.

Once the linear model of the system is obtained through impedance scan, various frequency domain techniques can be applied for stability analysis such as impedance intersection method [7], transfer function-based approach [8], reactance scan crossover technique [5], the Nyquist criterion for multiloop systems [9], Grey-box approach [10] etc. Each method has its strengths and weaknesses, and estimating the stability outcome of a non-linear system based on its linear equivalent is always challenging. Therefore, we have implemented three frequency domain methods; impedance intersection, reactance crossover and Grey-box and adopted a consensus mechanism (step 7 in Figure 1) to improve the robustness of the screening process. Typical threshold values have been defined for each frequency domain method to implement the consensus approach. For grey box method, a 5% damping factors is set as the threshold to identify poorly damped modes. For impedance intersection, a 90° phase margin is set as the stability threshold and for the reactance crossover a 20Ω resistance is considered as minimum damping requirement in the system. These thresholds are user defined and the values can be changed as appropriate for a specific case study. Through extensive testing under different operational scenarios and different proprietary model, practical threshold values will be identified in future.

The above screening method tags every scenario as either SSO risk, no SSO risk or Inconclusive. All scenarios other than no SSO risk are passed on to the EMT solver for further analysis. As an example, if we started with 50 scenarios, the screening process might find only $\frac{1}{5}$ of those require detailed analysis. This will still result in a large volume of time domain information that need to be analysed manually. The machine learning (ML) model is particularly useful in this case to automatically detect SSO events from simulation results (step 8 in Figure 1). The ML model is based on ensemble methods and utilises measurement data from different PMUs for training and validation. Currently, the model is tuned to predict oscillations in the subsynchronous range only, but it can be extended to harmonic range as well. The details of the ML model are available in the WP1 report.

The final output of the tool is labelled scenarios based on conclusive EMT simulations. The tool calculates participation factors based on Grey-box approach that can be extended in future for root cause analysis. The ten steps marked in Figure 1 are summarised in Table 1.

Table 1 Summary of the important steps in the SSO identification framework

Step	Exchanged data
1	EMT model file read by the SSO tool UI.
2	EMT model file updated by the tool based on user selected operational scenarios and global settings.
3	Frequency vs complex Impedance values in modified sequence domain. Results saved in frequency_results.db database file.
4	Identified modes based on system identification. Results saved in frequency_results.db database file.
5	Participation of different apparatus in the identified modes. Results saved in frequency_results.db database file.
6	Damping and phase margin calculation based on reactance crossover and impedance intersection methods. Results saved in frequency_results.db database file.
7	Consensus based on SSO/non-SSO outcome from each of the three frequency domain methods. Every scenario is tagged as either SSO risk, No SSO risk or Inconclusive and saved in tagged_scenarios.db database file.
8	SSO risk scenarios (including inconclusive scenarios) are studied in EMT simulation and instantaneous current and voltage measurements are passed to the ML model.
9	Instantaneous data from different sources used to train an ML model to identify SSO in any time domain signal.
10	Scenarios studied in step 8 are further labelled as SSO or non-SSO based on conclusive EMT simulations.

2.2 PSCAD Automation

The SSO analysis tool has three major functions – (a) PSCAD automation, (b) frequency domain analysis and (c) machine learning model for oscillation detection. The python modules for each function are developed such that they can be used independently, adhering to the computer science design practice of 'separation of concerns'.

Section 2.2.1 provides a list of all user variables available in the tool to adjust the performance of the automation module. Section 2.2.2 presents the entire module structure for PSCAD automation function for easier understanding of the structure of the code.

2.2.1 User variables

Variable name	Python file	Default value (type)	Purpose
model_name	main	Cigre_Benchmark_ACinj_updated: str	Name of the PSCAD project file
canvas_name	main	WindFarms: str	Name of the canvas where the component parameters are to be varied
model_folder	main	pscadmodel: str	Folder name containing the PSCAD model
pscad_version	main	PSCAD 5.0.1: str	PSCAD version number
fortran_version	main	'Intel 19.2.3372': str	Fortran version number to

Commented [SM(3]: The one available in the sharepoint is named 'Base_15167_NGESO_Acinj.pscx'. Is it the same as this one?

Commented [DC4R3]: It is the same model with some changes to the network parameters. I will upload the latest model before I share the final code.

Variable name	Python file	Default value (type)	Purpose
			use for the simulations
minimize	main	True: Boolean	Minimise the PSCAD software while running the tool
max_concurrent_sim	main	8: int	Maximum concurrent simulation sets allowed by PSCAD license
LCP_max_concurrent_exec	main	8: int	Maximum concurrent simulations allowed by PSCAD license
component_categories	main	["xfmr-3p2w", "resistor", "inductor", "capacitor", "source_3", "Bus", "filter", "multimeter", "const", "GraphFrame"]: list[str]	Name of the component categories used in the PSCAD project file
time_duration	main	4.0: float	Total time duration of the simulation
time_step	main	25.0: float	Integration time step of the simulation

Variable name	Python file	Default value (type)	Purpose
plot_type	main	1: int	To save channels to disk
export_to_csv	main	False: Boolean	Export time domain results into csv files
keep_previous_results	main	False: Boolean	Keep previous csv result files
inj_mag	main	0.5: float	Magnitude of injection by the Harmonic Current injection block from the PSCAD master library
ramp_time	main	0.1: float	Ramp time of the injection by the Harmonic Current injection block from the PSCAD master library
freq_incr	main	1: int	Frequency increment of the injected voltage by

Commented [SM(5]: Is this used for DC injections? I am not sure if the current injection block is configured to be used. Please confirm.

Commented [DC6R5]: Yes, this is for DC injections i.e., the Harmonic injection module from PSCAD library. It used to work fine but I have not tested it recently. I will test it before I share the final code.

Variable name	Python file	Default value (type)	Purpose
			the Harmonic Current injection block from the PSCAD master library
inj_phase	main	0: int	Phase of injection of the injected voltage by the Harmonic Current injection block from the PSCAD master library
injection_sequence	main	pos: str	Sequence of the injected voltage by the Harmonic Current injection block from the PSCAD master library
scan_type	main	wholesystem: str	Type of scan performed

Variable name	Python file	Default value (type)	Purpose
			by the TNEI scanner
scan_location	main	Gen: str	Location in the network where the scanner is connected
enable_scanner	main	True: Boolean	Enable or disable the TNEI scanner
freq_band	main	1: int	Resolution of frequency injection by the sequential injection module
f_min	main	1: int	Starting frequency of sequential injection module
f_max	main	40: int	Ending frequency of sequential injection module
inj_type	main	AC: str	AC (sequential) or DC (injection in band)

Variable name	Python file	Default value (type)	Purpose
FREQ_CALC_TYPE	module_selector	MFC: str	Mirror Frequency coupled (MFC) or Mirror Frequency Decoupled (MFD) to calculate either SISO or MIMO form
IMP_SCAN_MODULE_DEF_NAME	automation_pscad	IMP_SCAN_1: str	Definition name of the TNEI impedance scanner
CURRENT_INJ_MODULE_NAME	automation_pscad	Cinj: str	Name of Harmonic Current injection block from the PSCAD master library
FREQ_CONST_NAME	automation_pscad	FreqHZ: str	Name of the frequency block for sequential injection
VOLTAGE_PEAK_CONST_NAME	automation_pscad	LG_PeakV: str	Name of the voltage block for sequential injection

Variable name	Python file	Default value (type)	Purpose
VOLTAGE_INJ_A	automation_pscad	Vinj_A: str	Source name for the sequential voltage injection
VOLTAGE_INJ_B	automation_pscad	Vinj_B: str	Source name for the sequential voltage injection
VOLTAGE_INJ_C	automation_pscad	Vinj_C: str	Source name for the sequential voltage injection
PSCAD_CASE_CHARACTER_LEN	automation_pscad	30: int	Character length limit for PSCAD project names

2.2.2 Module structure

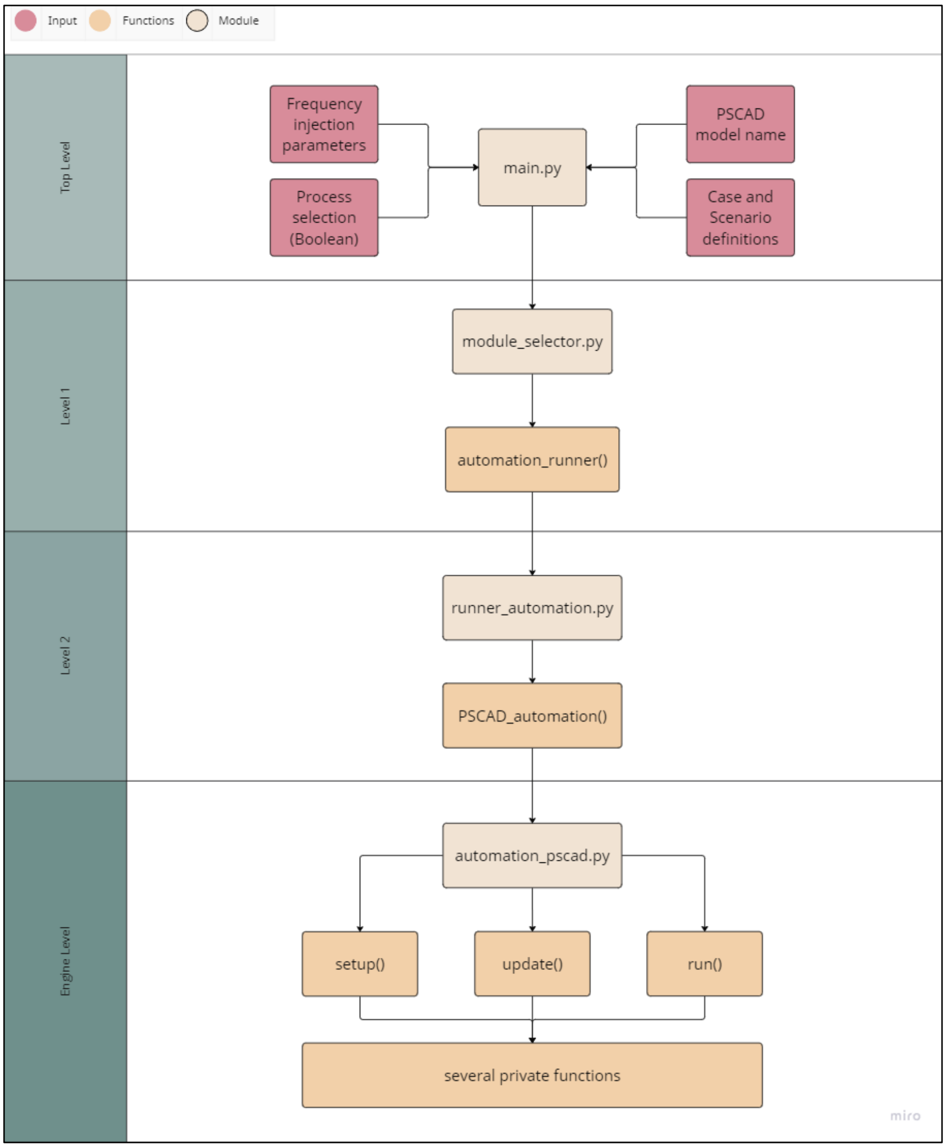


Figure 2 PSCAD automation tool structure and module hierarchy

2.2.3 Custom PSCAD blocks

Two different custom PSCAD blocks are developed to interact with PSCAD project files. These blocks are created to ensure a consistent interface between the project file and the python code. The custom blocks allow certain parameters to be modified by the user either from PSCAD model directly or through the SSO tool user interface.

2.2.3.1 Wind farm user dialog box

The wind farm dialog box is necessary to specify parameter values MW and Mvar setpoints in a proprietary wind farm model. The proprietary models usually have their own dialog box interface and there is no standardization or industry best practice guideline to align the input names from different vendors. Therefore, it is difficult to write a generic code that will be able to interface with any proprietary models.

To get around this problem, a custom dialog box is created, as shown in Figure 3, that includes the variables most useful for the purpose of the SSO analysis. The SSO tool code interfaces with this dialog box while the dialog box interfaces with proprietary models to pass on the relevant information.

Users will have to manually link the custom dialog box with any proprietary model during the base case model set up. Once the link is established, the parameters will be modified by the SSO tool automatically.

The options available to the user to vary from the custom dialog box are shown in Figure 4. These are generic parameters available in every proprietary model.

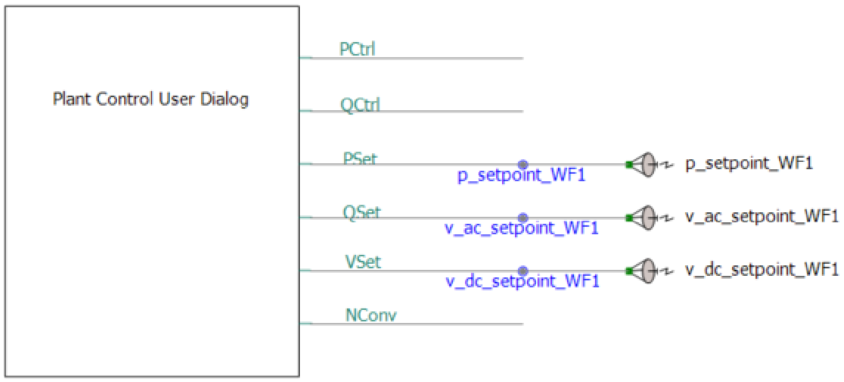


Figure 3 Wind farm control custom block to link with proprietary model MHI.

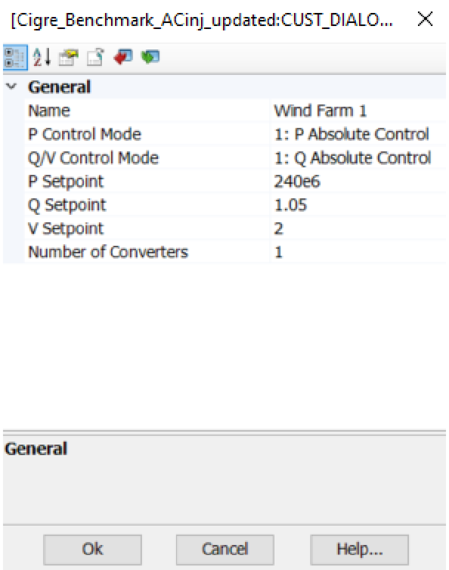
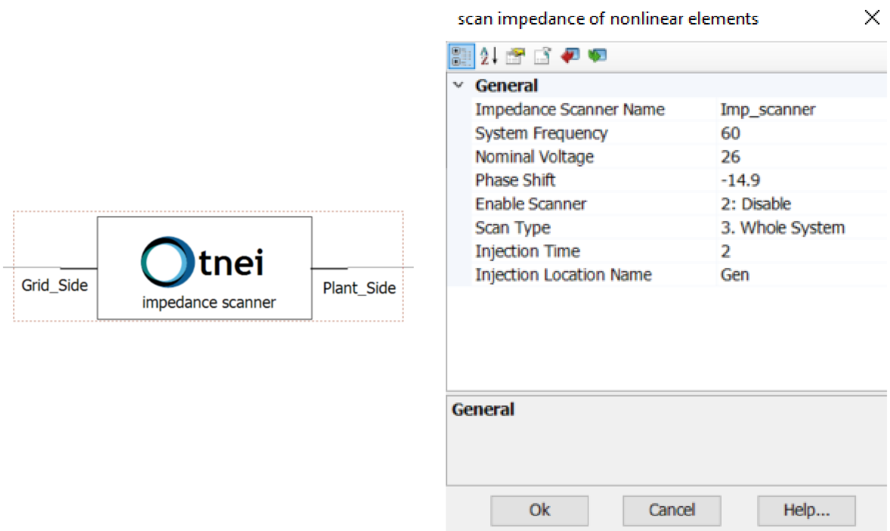


Figure 4 Wind farm custom dialog window showing available options

2.2.3.2 Impedance scanner

The TNEI impedance scanner is designed to allow voltage injections into the network in two different ways – (a) DC injection where positive sequence voltage is injected in a frequency band with a quadratic phase angle displacement, and (b) AC injection where both positive and negative sequence voltages are injected, one frequency at a time.

Figure 5(a) shows the Grid-side and Plant-side view of the impedance scanner. This block can be placed anywhere in the network along a line to measure the impedance of apparatus, grid and the whole system. Figure 5(b) shows the dialog window with user parameters to modify the characteristic of the scanner.



(a) (b)

Figure 5 Impedance scanner block developed in the project, (a) view of the block in PSCAD, and (b) options available to the user for scanning frequency of apparatus.

A short description of the parameters is available within PSCAD when the curser is pointed to the respective locations. These parameters are also explained in Table 2.

Table 2 List of user parameters in the impedance scanner

Name	**Type**	**Purpose**
Impedance scanner name	string	Name of the scanner – Optional
System frequency	integer	Nominal frequency of the system
Nominal voltage	integer	Nominal voltage of the busbar/line where the scanner is connected to the network.

Phase shift	float	Phase angle of the scanner's voltage source to control the exchange of power flow between the scanner and the apparatus.
Enable scanner	string	Enable or disable the scanner
Scan type	string	Apparatus side, grid side or whole system scan
Injection time	integer	Simulation time when the injection to the network is initiated
Injection location name	string	Name of the busbar/area where the scanner is connected

2.3 Frequency domain analysis

The frequency domain analysis function implements three frequency domain methods (a) impedance intersection, (b) reactance crossover and (c) grey-box method. The fundamentals of these methods have been discussed in detail in the WP2 report.

Section 2.3.1 provides a list of all user variables available in the tool to adjust the performance of the frequency domain analysis module. Section 2.3.2 presents the entire module structure for the frequency domain function for easier understanding of the structure of the code.

2.3.1 User variables

Variable name	Python file	Default value (type)	Purpose
grey_box	main	0.05: float	Damping factor threshold to characterise a system mode as critical
impedance_intersection	main	90: int	Phase margin threshold to identify

Variable name	Python file	Default value (type)	Purpose
			potential damping issues.
reactance_crossover	main	20: int	System resistance threshold to identify potential damping issues.
FFT_WINDOW_START_TIME	runner_frequency_domain	2: float	Starting time of the FFT window applied to the time domain simulation results.
META_TABLE_NAME	runner_frequency_domain	main_impedance_table: str	Name of the frequency scan meta table in the frequency_results.db file
META_TABLE_NAME	grey_box_method	main_impedance_table: str	Name of the frequency scan meta table in the frequency_results.db file
META_TABLE_RESULTS	grey_box_method	main_results_table: str	Name of the frequency domain analysis results meta table in the frequency_results.db file
META_TABLE_NAME	impedance_based_stability_ assessment	main_impedance_table: str	Name of the frequency scan meta table in

Commented [SM(7]: In WP2 report, page 41 of 48 Appendix B, section 5, it is written " "The condition for SSO risk for a given system condition (dispatch, network outages, etc) can be defined as follows "If there are reactance crossover on the system and those frequencies have negative resistance on the WTG scan, then there is a risk of SSO" (see Figure B. 4). In our reactance crossover plots why are we comparing system reactance curve with system resistance and not with WTG resistance obtained from WTG side scan?

Commented [DC8R7]: Good point Shiv. We can analyse it both ways. Either we can monitor the turbine scan for -ve resistance or we can check the whole system resistance for a low value.

Variable name	Python file	Default value (type)	Purpose
			the frequency_results.db file
META_TABLE_RESULTS	impedance_based_stability_ assessment	main_results_table: str	Name of the frequency domain analysis results meta table in the frequency_results.db file

2.3.2 Module structure

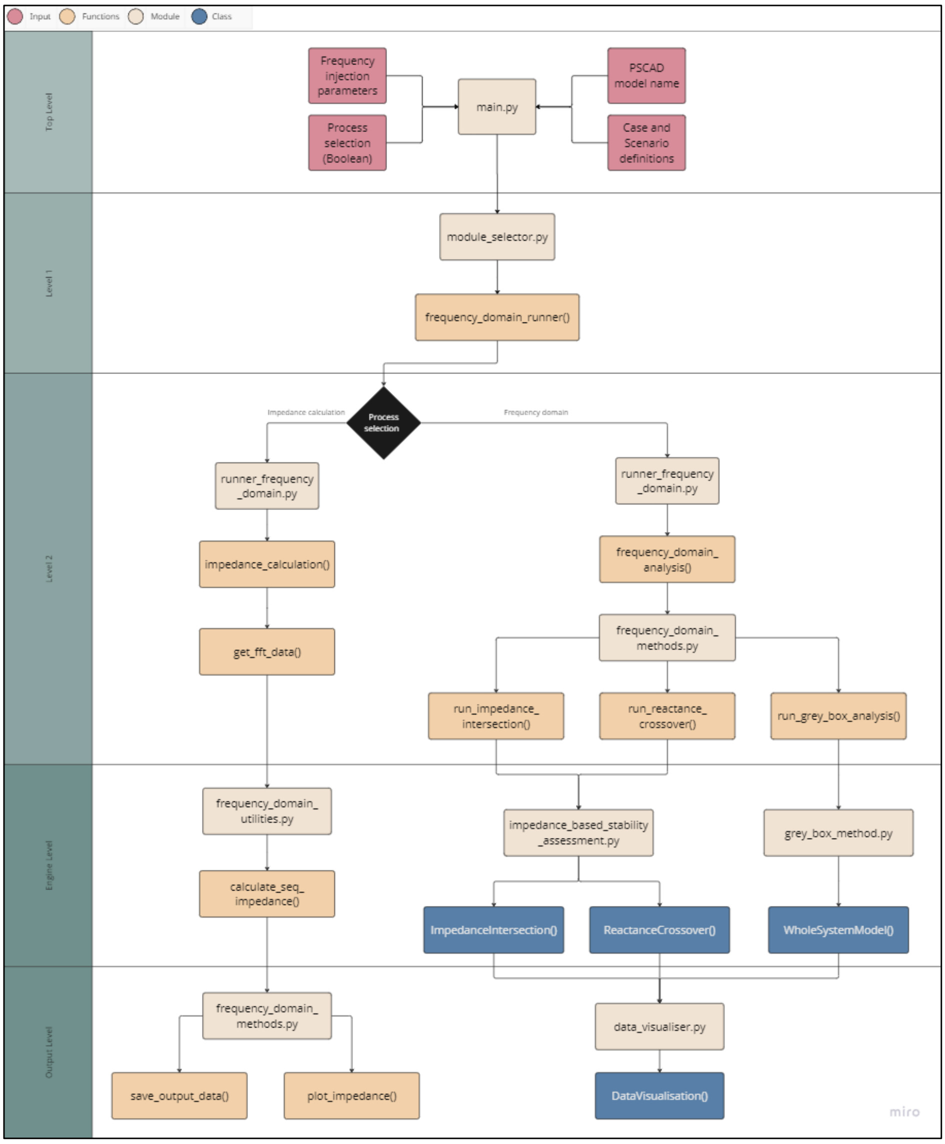


Figure 6 Frequency domain analysis tool structure and module hierarchy

2.4 Machine learning model

The ML model is based on ensemble methods and utilises measurement data from different PMUs for training and validation. Currently, the model is tuned to predict oscillations in the subsynchronous range only, however, this work can be extended to detect oscillations in the harmonic range without any fundamental changes to the training process.

Various methods exist in literature to detect oscillations from time domain signals. A common conclusion from these works is that identifying SSO from raw measurements is not straightforward. As shown in Figure 7, a change in amplitude is expected during an SSO event. However, this change is not unique to SSOs, and there could be other causes, such as switching operations, setpoint adjustments, planned outages, power quality disturbances, etc. In addition, there will be background harmonics and measurement errors. These works confirm, the multitude of causes (and non-mono causality) render the reliable detection of SSOs challenging. Previous literature approached the detection with techniques from signal processing or data-driven (ML) methods. Signal processing techniques offer higher accuracy, but they are computationally expensive [11]. Data-driven methods are fast once the model is trained but training the ML models requires a significant number of (labelled) data samples which may not always exist (e.g. simulations that contain SSOs and non SSOs).

This work develops a hybrid approach that combines the advantages of the signal processing and data-driven methods. The key requirements for an SSO detector using ML are that the training process is fast and does not require many labelled samples, that the detector is easy to understand, can be extended to consider other features, and enables fast detection suitable for near real-time applications.

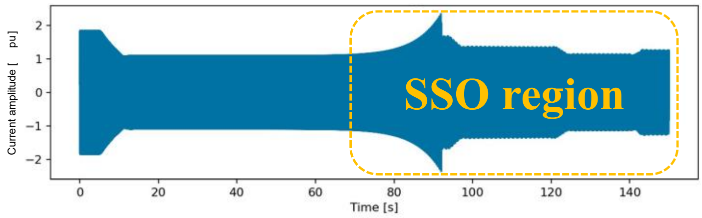


Figure 7 Example of a current measurement in pu showing an SSO event. Created with data from IEEE DataPort [12].

2.4.1 User variables

Variable name	Python file	Default value (type)	Purpose
save_classifier	main	True: Boolean	Save the trained classifier model for future use
calibrate_scalers	main	False: Boolean	Refit the scalers to the feature data range of the current measurement and updates the saved model. Only calibrate when using a different data source than the one used to fit the classifiers

2.4.2 Module structure

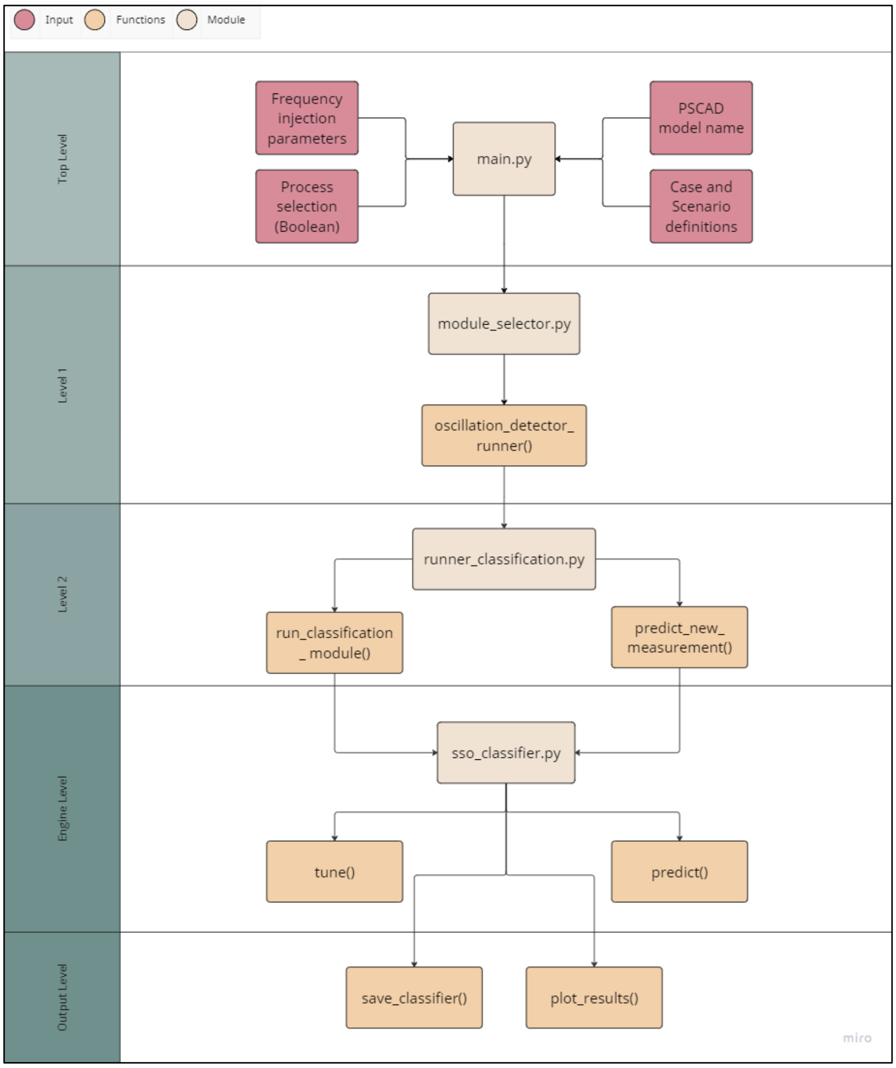


Figure 8 Machine learning based oscillation detector tool structure and module hierarchy

2.5 User Interface

The user interface is a browser-based application developed in React framework. The interface offers full control over the simulation case creation, settings adjustment, and results visualisation. It comes with an in-built walkthrough function in the help section to help users familiarise themselves with the interface. The following subsections are meant to guide users through the different functions step by step based on screen grabs of the interface for a PSCAD test network model.

2.5.1 Home page and help section

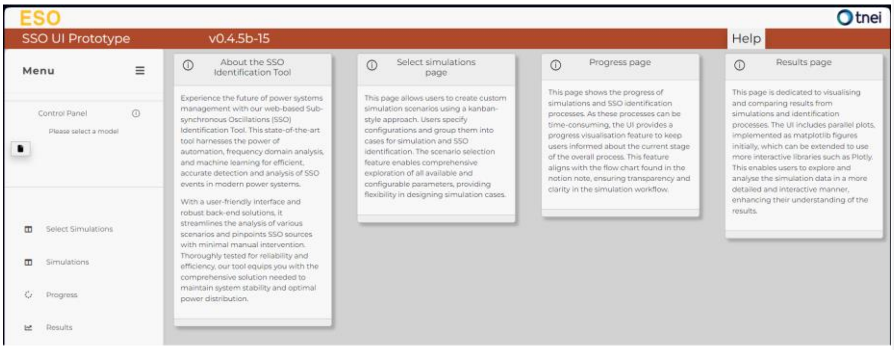


Figure 9: The home page or landing page for the tool interface

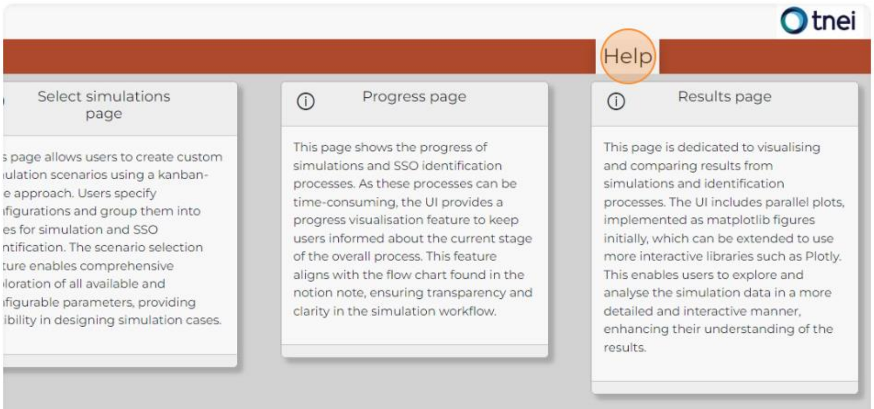


Figure 10: Help button for more information about how to interact with the interface

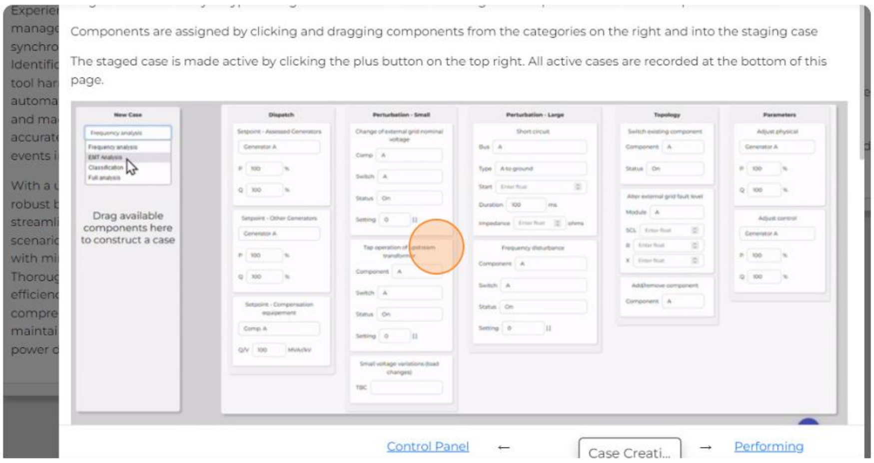


Figure 11: An automated walkthrough of the entire process

2.5.2 Select Model

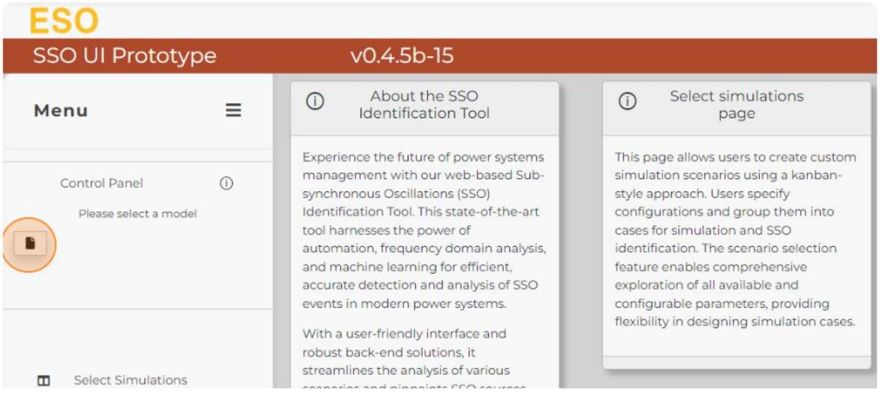


Figure 12: Select a PSCAD model file (.pscx or .pswx)

2.5.3 Global settings

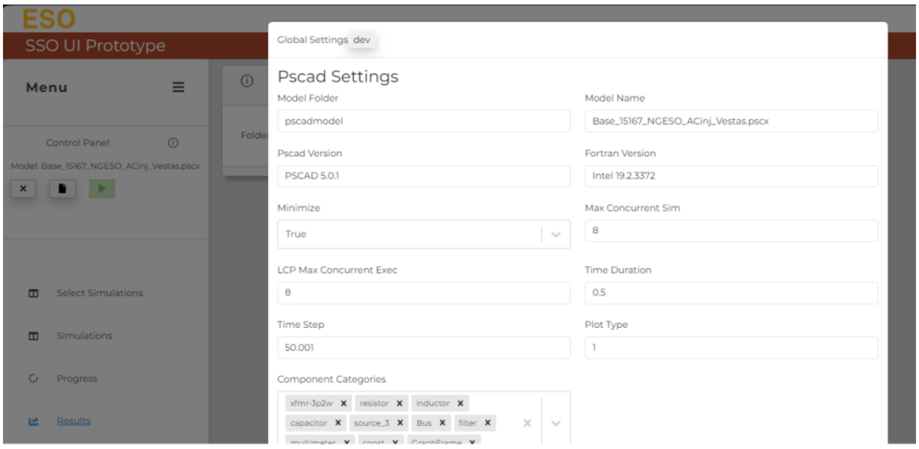


Figure 13: Different settings users can modify for a selected PSCAD file

2.5.4 Select a simulation type

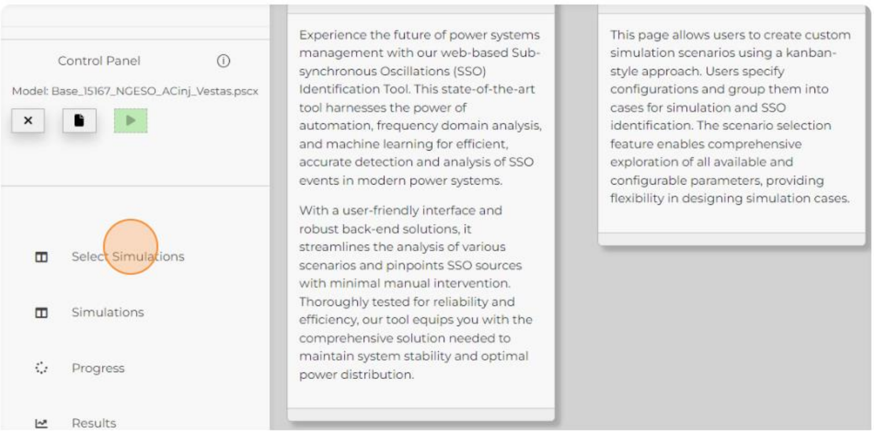


Figure 14: Click on 'Select Simulations' in the side panel to navigate to the new case generator page

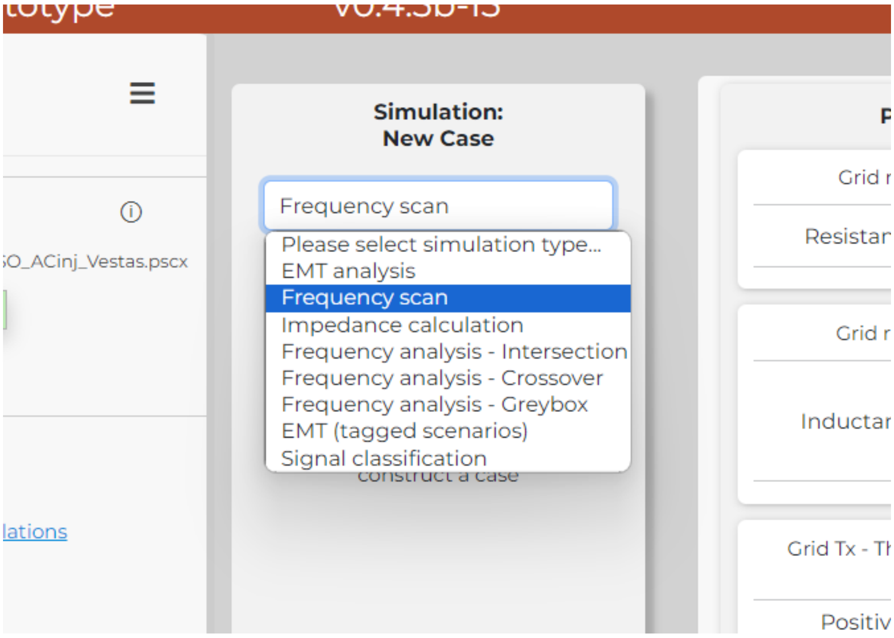


Figure 15: Select a simulation type from the dropdown menu

2.5.5 Case generator

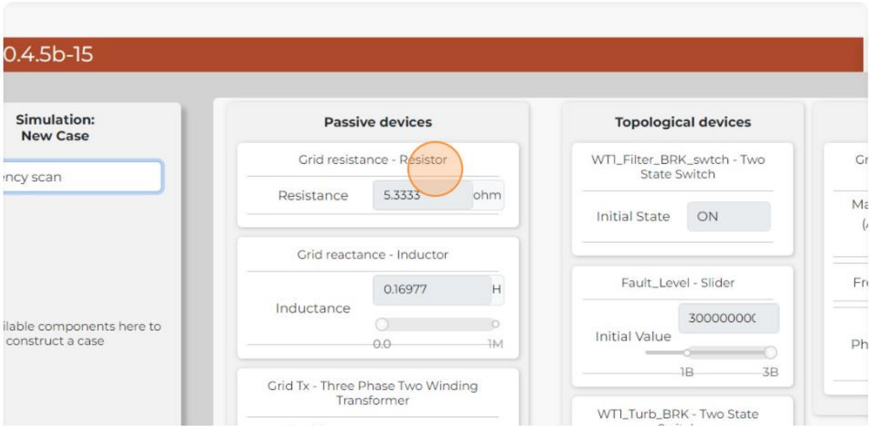


Figure 16: Drag individual devices to the left side to add to a new simulation case

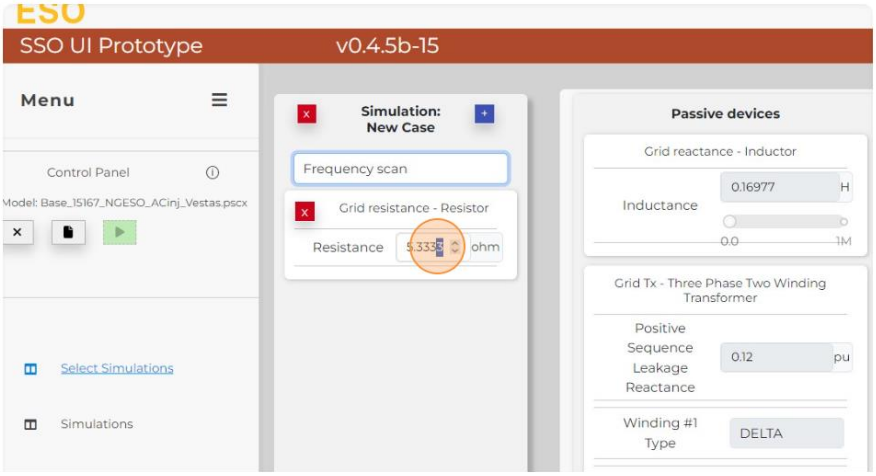


Figure 17: Change device parameters as necessary to create a new case

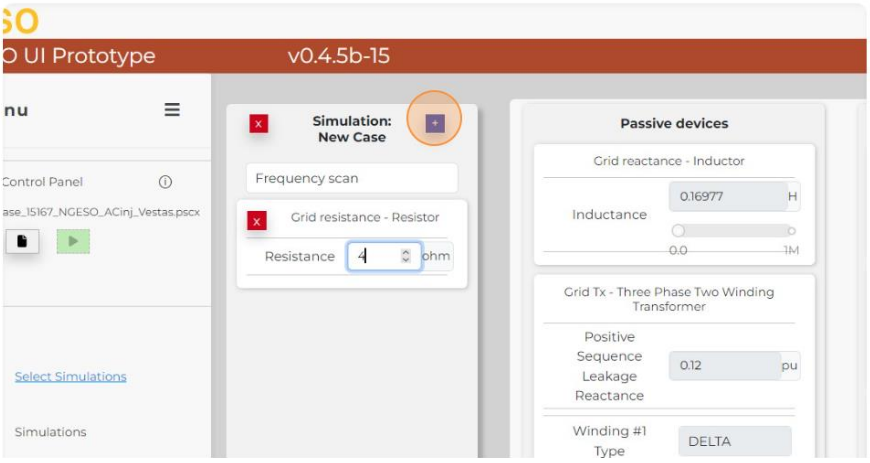


Figure 18: Create a new scenario by adding a new simulation case

2.5.6 View and run processes created by case generator

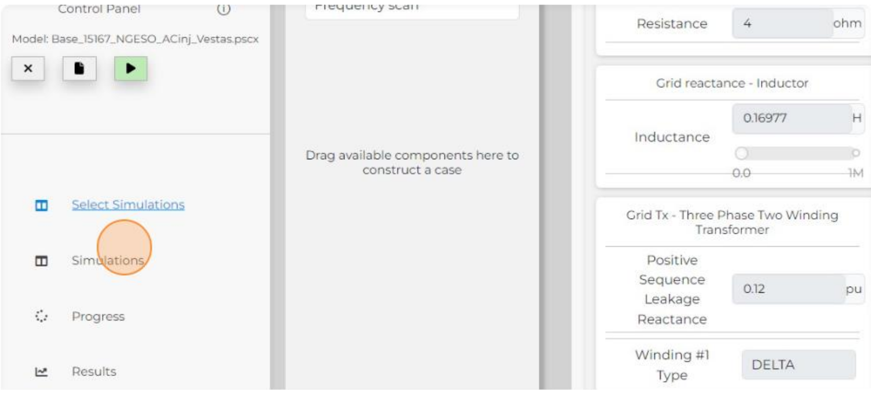


Figure 19: Click on the 'Simulations' tab in the side panel to view all cases

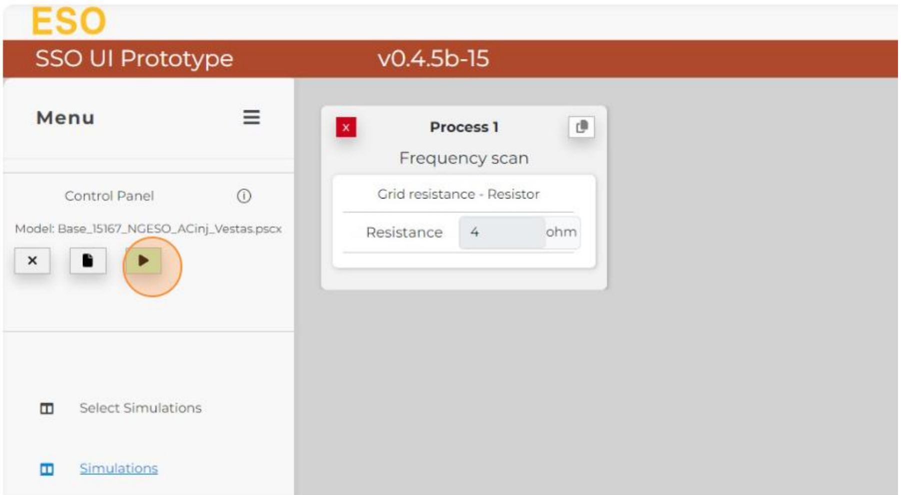


Figure 20: Run all simulation cases (Process 1, Process 2 etc)

2.5.7 Progress update

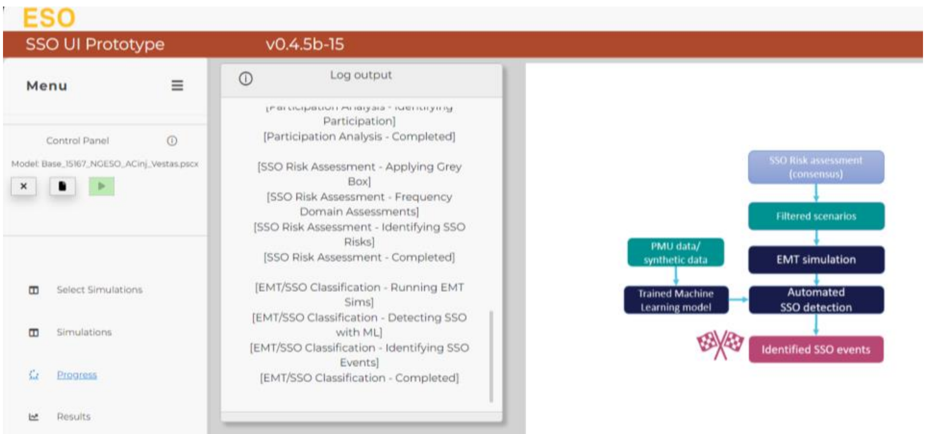


Figure 21: Output window showing the progress update along with a flowchart highlighting the completed steps

2.5.8 Results view

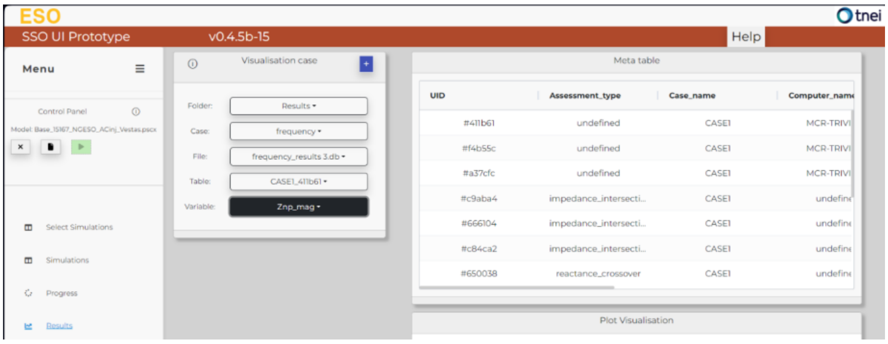


Figure 22: Explore results within the UI in table format and plotly figures

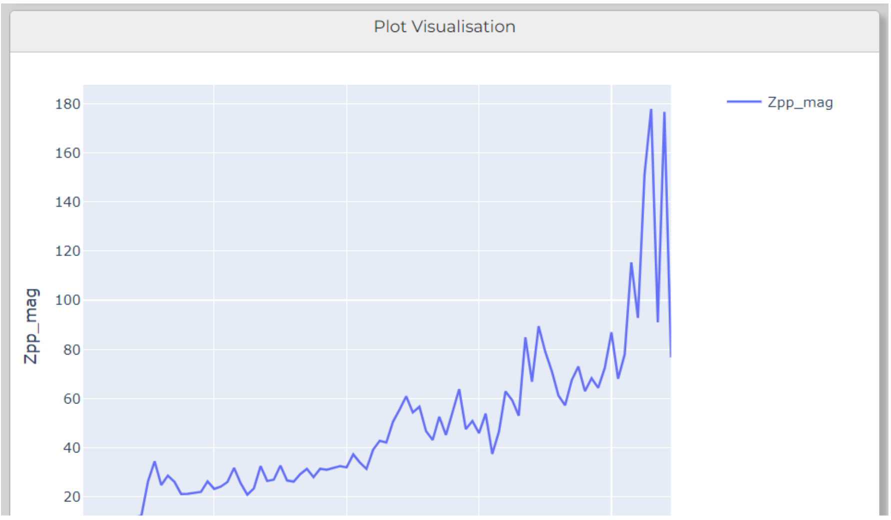


Figure 23: Plotly graph showing an example plot. It is possible to add more than one plot to the a single graph for comparison

3 SSO Tool Setup Guide

The SSO tool core modules are developed in Python while the frontend is developed in JavaScript. To run the tool successfully, users would need to install all required Python and JavaScript packages in addition to the PSCAD software. The following sections list all the setup requirements to enable a user to interact with the web-based user interface of the SSO tool.

3.1 Python packages

Packages	Link to package details
pandas>=1.4.4	https://pandas.pydata.org/
scipy>=1.9.1	https://scipy.org/
numpy>=1.23.3	https://numpy.org/
matplotlib>=3.6.2	https://matplotlib.org/
mat73>=0.59	https://pypi.org/project/mat73/
alive-progress	https://pypi.org/project/alive-progress/
seaborn>=0.12.2	https://seaborn.pydata.org/
openpyxl	https://openpyxl.readthedocs.io/en/stable/
scikit-learn>=1.1.2	https://scikit-learn.org/stable/
attrs>=22.2.0	https://www.attrs.org/en/stable/
setuptools>=57.0.0	https://pypi.org/project/setuptools/
SQLAlchemy>=2.0.5.post1	https://docs.sqlalchemy.org/en/20/
WMI>=1.5.1	https://pypi.org/project/WMI/
psutil>=5.9.4	https://pypi.org/project/psutil/
plotly>=5.14.0	https://plotly.com/
kaleido == 0.1.0.post1	https://pypi.org/project/kaleido/
pytest>=7.3.1	https://docs.pytest.org/en/8.0.x/
scikit-rf	https://scikit-rf.readthedocs.io/en/latest/

3.2 PSCAD automation packages

The PSCAD desktop application and the python module mhi-pscad=2.8.4 can be installed from the website of the developers of the software Manitoba Hydro International (MHI). PSCAD is a licensed commercial software that can be purchased from MHI. There are different categories of licenses that decide the number of allowed parallel simulations in PSCAD. Based on the license type owned by the user, the SSO tool should be configured to ensure that number of parallel simulations launched by the tool are within the license conditions.

3.3 Frontend packages

Packages	Link to package details
wheel	https://pypi.org/project/wheel/
requests	https://pypi.org/project/requests/
flask_login	https://flask-login.readthedocs.io/en/latest/
werkzeug	https://werkzeug.palletsprojects.com/en/3.0.x/
flask	https://flask.palletsprojects.com/en/3.0.x/
flask_assets	https://flask-assets.readthedocs.io/en/latest/
gunicorn	https://gunicorn.org/
dash	https://pypi.org/project/dash/
python-dotenv	https://pypi.org/project/python-dotenv/
dash_bootstrap_components	https://dash-bootstrap-components.opensource.faculty.ai/
dash_html_components	https://dash.plotly.com/dash-html-components
dash_core_components	https://dash.plotly.com/dash-core-components
flask_sqlalchemy	https://flask-sqlalchemy.palletsprojects.com/en/3.1.x/
flask_cors	https://flask-cors.readthedocs.io/en/latest/

3.4 Network model setup

The network model setup guide is based on the Cigre benchmark system shown in Figure 31. The model has two windfarms connected to a substation. The windfarms are of different size and therefore require separate user dialog boxes (Figure 3) to control the export level. The steps necessary to setup the benchmark model, including the windfarms are listed below –

1. Connect the custom plant control user dialog box to the windfarms using wireless radio links. Only the P setpoint, Q setpoint and the V setpoint need to be linked. If the links are already established in the base case model, then you can ignore this step.
2. Connect the TNEI impedance scanner (Figure 5) either at WF1 or WF2 after the 220/66 kV transformer. Only one scanner can be used in the model at a time. This limitation is due to the challenge of identifying the location of the measured signal in PSCAD even though the SSO can work with more than one scanner.

3. Set the scanner parameters to the values in Figure 24.

scan impedance of nonlinear elements

General	
Impedance Scanner Name	Imp_scanner
System Frequency	**50**
Nominal Voltage	**66**
Phase Shift	**0**
Enable Scanner	2: Disable
Scan Type	3. Whole System
Injection Time	2
Injection Location Name	**WF1**

Figure 24 Scanner parameters for the Cigre benchmark model

4. Set the external grid fault level to the values in Figure 25.

[Cigre_Benchmark_ACinj_updated:CalcGrid] id=...

General	
Name	CalcGrid
Nominal voltage	400e3
Nominal frequency	50
Short circuit power	**3000e6**
X/R ratio	10

Figure 25 External grid fault level parameters for a strong system

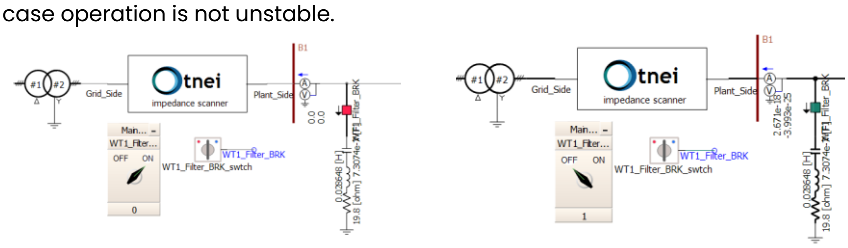
5. Run a simulation for 10 seconds to ensure that WF1 is exporting 240MW and WF2 is exporting 180MW and the system is stable.
6. Now enable the scanner as shown in Figure 26 to inject a random frequency from 2 seconds.

scan impedance of nonlinear elements

General	
Impedance Scanner Name	Imp_scanner
System Frequency	**50**
Nominal Voltage	**66**
Phase Shift	**0**
Enable Scanner	**1: Enable**
Scan Type	3. Whole System
Injection Time	2
Injection Location Name	**WF1**

Figure 26 Enable scanner to start injecting voltage from 2 seconds

7. Run a simulation for 10 seconds to ensure the WFs are operating at their previous export values and the system operating point has not changed due to injections.
8. If the user wants to study topological changes such as filter operation of the WFs then repeat steps 5 to 7 by switching off the filter (Figure 27) to ensure the base case operation is not unstable.



3. (a)
4. (b)

Figure 27 Toggle operation of PSCAD widget to change the status of WF filters, (a) switched on, (b) switched off.

9. Once the above steps are completed, the model should be saved in the 'sso_idenitfication_tool ---> data ---> pscadmodel' folder.
10. The user can launch the SSO tool user interface now and select the saved PSCAD file by clicking on the 'Select model' option. Alternatively, user can run the tool from the main.py file by changing the 'model_name' variable to the respective name of the PSCAD model.

3.5 Running a sample study

The Cigre benchmark model can be used to run EMT studies or impedance scans for different fault level scenarios, generator dispatch values, component parameters or topological variations. In this section, only fault level scenarios are considered as an example for a sample study. The steps to run impedance scans from 1 Hz to 40 Hz for the apparatus, grid and the whole system are explained below. These steps are for running the SSO tool from the main.py file –

1. Set the time duration for study to 4 seconds and time step to 25 microseconds.
2. Set the scan type to 'wholesystem', the scan location to the location of the scanner in the PSCAD file and enable scanner to True. Please note that the order of the scan type selection does not matter.

Commented [SM(9]: Is it necessary to start with wholesystem first? Does the order matters?

Commented [DC10R9]: The order does not matter. I have clarified this in the sentence.

3. Set the frequency band to 1 Hz, the starting frequency to 1 Hz, the ending frequency to 40 Hz and the injection type to 'AC'.
4. Set the Boolean run_frequency_scan = True and leave all other Boolean to False.
5. Run the main.py from the Python IDE.
6. After the completion of the scan from 1 Hz to 40 Hz, set the scan type to 'apparatus' and repeat step 5.
7. After the completion of the scan from 1 Hz to 40 Hz, set the scan type to 'grid' and repeat step 5.
8. The time domain results of the scan are saved in database files (one database file per case study) in 'sso_idenitfication_tool ---> data ---> pscaddata ---> results' folder. The results of 'grid', 'apparatus' and 'wholesystem' are saved in a single database file for a particular case study.

Table: main | Filter in any column

	case_name	scenario_name	pscad_component	pscad_scenario	harm_inj_module_params	scan_location	simulation_type	imulations_timestamp	computer_name	UID
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 51.0, "FMax": 51.0, "FInc": 0.5, "FDQ": 1.0, ...	WTG1	Frequency Scan	20231007_173316	GLS-CHAKRAD	#d0fd07
2	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 49.0, "FMax": 49.0, "FInc": 0.5, "FDQ": 1.0, ...	WTG1	Frequency Scan	20231007_173317	GLS-CHAKRAD	#5816a7
3	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 51.5, "FMax": 51.5, "FInc": 0.5, "FDQ": 1.5, ...	WTG1	Frequency Scan	20231007_173326	GLS-CHAKRAD	#067966
4	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 48.5, "FMax": 48.5, "FInc": 0.5, "FDQ": 1.5, ...	WTG1	Frequency Scan	20231007_173357	GLS-CHAKRAD	#53a104
5	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 52.0, "FMax": 52.0, "FInc": 0.5, "FDQ": 2.0, ...	WTG1	Frequency Scan	20231007_173358	GLS-CHAKRAD	#f73a7f
6	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 48.0, "FMax": 48.0, "FInc": 0.5, "FDQ": 2.0, ...	WTG1	Frequency Scan	20231007_173409	GLS-CHAKRAD	#13b87d
7	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 52.5, "FMax": 52.5, "FInc": 0.5, "FDQ": 2.5, ...	WTG1	Frequency Scan	20231007_173438	GLS-CHAKRAD	#8c6984
8	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 47.5, "FMax": 47.5, "FInc": 0.5, "FDQ": 2.5, ...	WTG1	Frequency Scan	20231007_173440	GLS-CHAKRAD	#3c2c88
9	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 53.0, "FMax": 53.0, "FInc": 0.5, "FDQ": 3.0, ...	WTG1	Frequency Scan	20231007_174219	GLS-CHAKRAD	#7be8fc
10	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 47.0, "FMax": 47.0, "FInc": 0.5, "FDQ": 3.0, ...	WTG1	Frequency Scan	20231007_174221	GLS-CHAKRAD	#a32539
11	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 53.5, "FMax": 53.5, "FInc": 0.5, "FDQ": 3.5, ...	WTG1	Frequency Scan	20231007_174222	GLS-CHAKRAD	#182616
12	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenari...	{"CMag": 0.5, "FMin": 46.5, "FMax": 46.5, "FInc": 0.5, "FDQ": 3.5, ...	WTG1	Frequency Scan	20231007_174224	GLS-CHAKRAD	#b94834

Figure 28 Screenshot of the frequency injection measurement results from 1 Hz to 40 Hz for different scan types and different scan locations

9. To calculate the impedance from the frequency scan results, set the Boolean run_impedance_calculation=True.
10. In addition, to run impedance domain analysis using the calculated impedance, set the Booleans run_impedance_intersection=True, run_reactance_crossover=True, run_grey_box=True and run_consensus=True.
11. Set Booleans run_EMT and run_frequency_scan to False.
12. Repeat step 5.
13. The impedance scan plots, and the frequency domain analysis graphs will be displayed in the default browser of the machine.

14. The results used to generate the plots are saved in 'sso_idenitfication_tool ---> data ---> pscaddata ---> results' folder in the database file 'frequency_results.db'. A screenshot of the main_impedance_table is shown in Figure 29 and the main_results_table is shown in Figure 30.

Table: main_impedance_table

	id	case_name	scenario_name	pscad_component	pscad_scenario	scan_type	scan_location	computer_name	UID
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	CASE1	Scenario1	{"CASE1_Scenario1_92a220": {"896997969": ...	{"CASE1_Scenario1_92a220": ...	wholesystem	WTG2	GLS-CHAKRAD	#53dd4e
2	2	CASE1	Scenario1	{"CASE1_Scenario1_92a220": {"896997969": ...	{"CASE1_Scenario1_92a220": ...	wholesystem	WTG1	GLS-CHAKRAD	#4b9d38
3	3	CASE2	Scenario1	{"CASE1_Scenario1_d0fd07": {"896997969": ...	{"CASE1_Scenario1_d0fd07": ...	apparatus	WTG1	GLS-CHAKRAD	#0e5437
4	4	CASE2	Scenario1	{"CASE1_Scenario1_d0fd07": {"896997969": ...	{"CASE1_Scenario1_d0fd07": ...	grid	WTG1	GLS-CHAKRAD	#d962d6
5	5	CASE2	Scenario1	{"CASE1_Scenario1_71c153": {"896997969": ...	{"CASE1_Scenario1_71c153": ...	wholesystem	WTG1	GLS-CHAKRAD	#8a21b5

Figure 29 Screenshot of the meta table from the frequency_results database showing the calculated impedance

Table: main_results_table

	id	case_name	scenario_name	pscad_component	pscad_scenario	scan_location	assessment_type	figure	UID
	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter	Filter
1	1	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenario...	na	grey_box	{"data": ...	#ac635d
2	4	CASE2	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenario...	WTG1	impedance_inters...	{"data": [{"mode"...	#79c1a4
3	1	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenario...	WTG2	reactance_crossover	{"data": [{"mode"...	#a140d2
4	2	CASE1	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenario...	WTG1	reactance_crossover	{"data": [{"mode"...	#ea2270
5	5	CASE2	Scenario1	{"CASE1_Scenario...	{"CASE1_Scenario...	WTG1	reactance_crossover	{"data": [{"mode"...	#7fc7d5

Figure 30 Screenshot of the meta table from the frequency_results database showing the results from the frequency domain analysis

15. The results of the consensus mechanism are saved 'sso_idenitfication_tool ---> data ---> pscaddata ---> results' folder in the database file 'tagged_scenarios.db'. This database provides the final list of cases that are tagged as having potential SSO risks based on frequency domain analysis.

4 Case Study

4.1 SSO Case – Cigre Model

The original benchmark model for multi-frequency analysis is developed in MATLAB Simulink. The authors have migrated the model to PSCAD and used it as the base model for the tool development. The PSCAD model has been validated with the original Simulink model. The validation results are not included here as these have been discussed in detail in the WP2 report. Figure 31 shows the network diagram of the benchmark system. It has two windfarms (WF1 240 MW and WF2 180 MW capacity) connected to a 220kV offshore substation with an AC export cable connecting to the onshore 400kV substation. The onshore network is represented by a voltage source behind a variable fault impedance to reproduce different grid conditions. Both windfarms have grid following turbines.

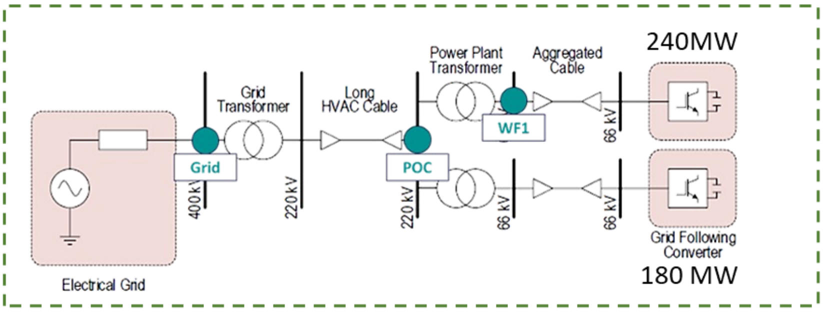


Figure 31 SSO benchmark model network diagram. Figure adapted from [13]

4.1.1 Frequency scan

Three different frequency scans are performed on the Cigre network. Apparatus side scan (Figure 32) i.e., scanning the impedance of the wind farm after the onshore export transformer. The scans are performed using the modified sequence domain method that captures the coupling between positive and negative sequence networks. The apparatus scan is performed with an infinite voltage source and before the scan is performed it is ensured that the wind farm is exporting at a stable operating point.

Grid side scan (Figure 33) calculates the impedance of the remaining grid i.e., grid in parallel with WF2. Similar to apparatus scan, the grid scan uses the modified sequence domain method of impedance calculation. In grid side scan, the fault level of the external grid is considered.

The third type of scan is called Whole System. In this case, the entire grid is scanned to provide the modified sequence domain impedance of both wind farms connected in parallel to the external grid. These scans are performed from both wind farm locations. The results from WF1 are shown in Figure 34 and from WF2 are shown in Figure 35.

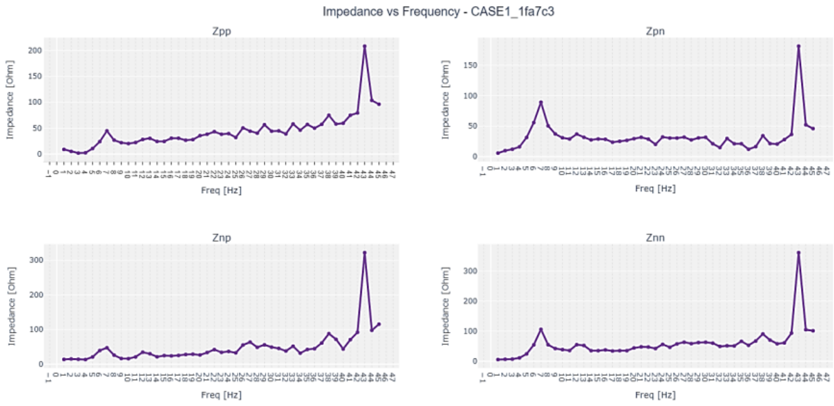


Figure 32 Apparatus scan from WF1

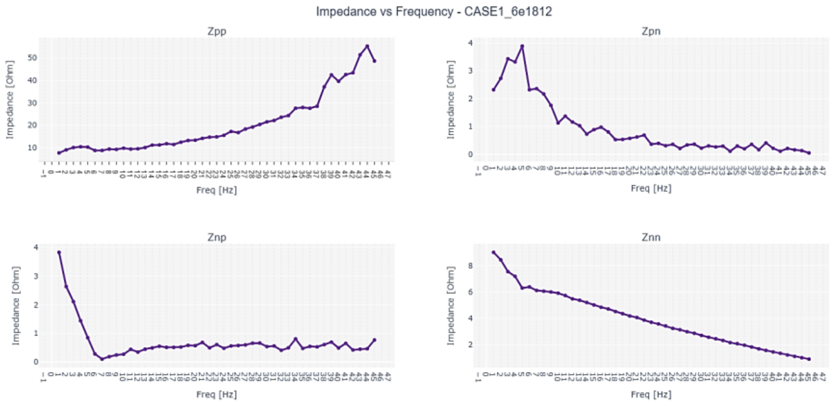


Figure 33 Grid scan from WF1 for 1200MVA external grid fault level

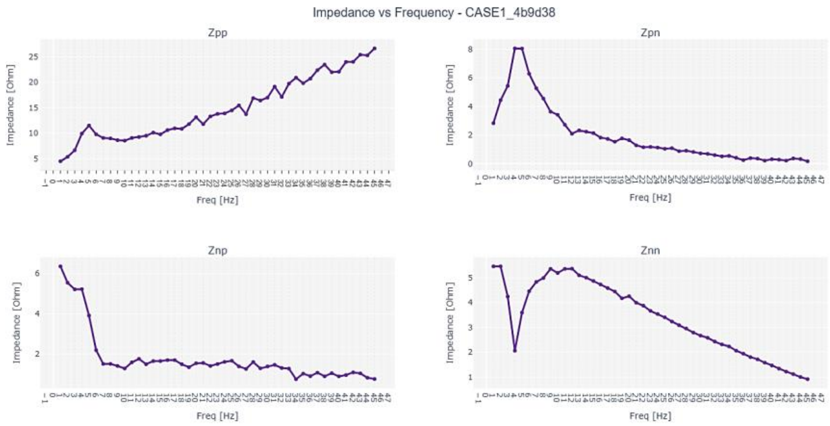


Figure 34 Whole system scan from WF1 for 1200MVA external grid fault level

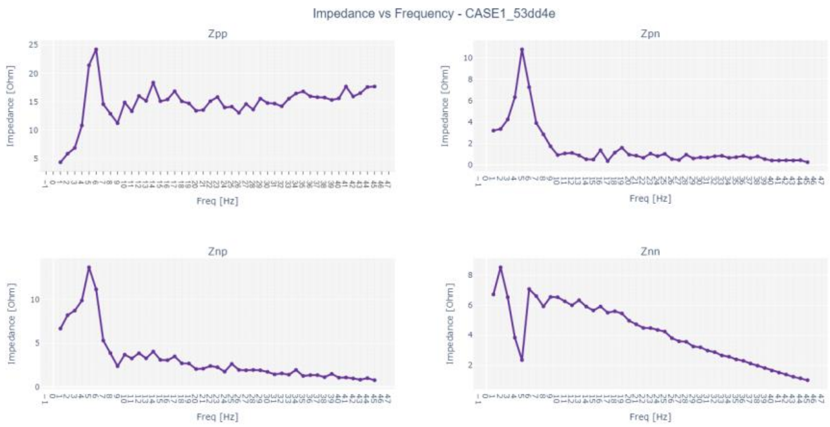


Figure 35:Whole system scan from WF2 for 1200MVA external grid fault level

4.1.2 Frequency domain analysis

The case study shows the performance of the frequency domain screening technique for relatively weak (1200MVA, 2.85 SCR) and very week (750MVA, 1.78 SCR) system conditions of the onshore grid. Figure 36 shows the reactance crossover calculations based on the whole system scans performed at WF1 and WF2 for the two fault levels. No intersection of the overall reactance of the system is observed at WF1 for a fault level of 1200MVA (Figure 36 (a)). For the same fault level, measurement at WF2 shows an intersection close to the fundamental frequency (Figure 36 (c)). As the fault level reduces, the intersection frequency reduces as well at both locations (Figure 36 (b) & (d)). However, no conclusions can be drawn as the system damping is positive at intersection points.

The results from the Grey-box calculation are shown in Figure 37. The estimated modes of the system show a clear movement towards the imaginary axis as the fault level of the system reduces from 1200MVA (Figure 37 (a)) to 750MVA (Figure 37 (b)). The damping factor of the 9.5Hz mode is 2% which falls within the assumed 5% threshold to consider as a critical scenario. Additionally, a participation analysis of two poorly damped modes, 12.4Hz at 1200MVA and 9.5Hz at 750MVA, reveal an interesting outcome. For the higher fault level scenario, WF2 has higher participation in the 12.4Hz mode (Figure 37 (c)) but the relative participation swaps places as the fault level reduces. This is a useful information for SOs to perform root cause analysis and to take effective mitigation actions by contacting the plants interacting negatively with the poorly damped modes of the system.

Figure 38 presents the results of impedance intersection method measured at WF1. For the higher fault level scenario, only one intersection is observed below 50Hz between the wind farm scan (red line) and the grid scan (blue line) in Figure 38 (a). Although the intersection frequency is in the subsynchronous range (4.26Hz), the corresponding phase margin is significant (Figure 38 (c)). The phase margin is calculated as per equation (3).As the fault level reduces, one more intersection appears at 9.64Hz having a phase margin of around 85°. This margin is within the assumed threshold of 90°, and the frequency coincides with the outcome of the Grey-box method.

$$\begin{gathered}\theta_{margin} = (\theta_{grid} - \theta_{apparatus}) + 180 \\ if\ \theta_{margin} > 180,\ \theta_{margin} = \theta_{margin} - 360\end{gathered} \tag{3}$$

Commented [SM(11]: If you could elaborate how phase margin is calculated (the mathematical relation) using the phase angle information, it would be helpful.

Commented [DC12R11]: Sure, I have added the equation

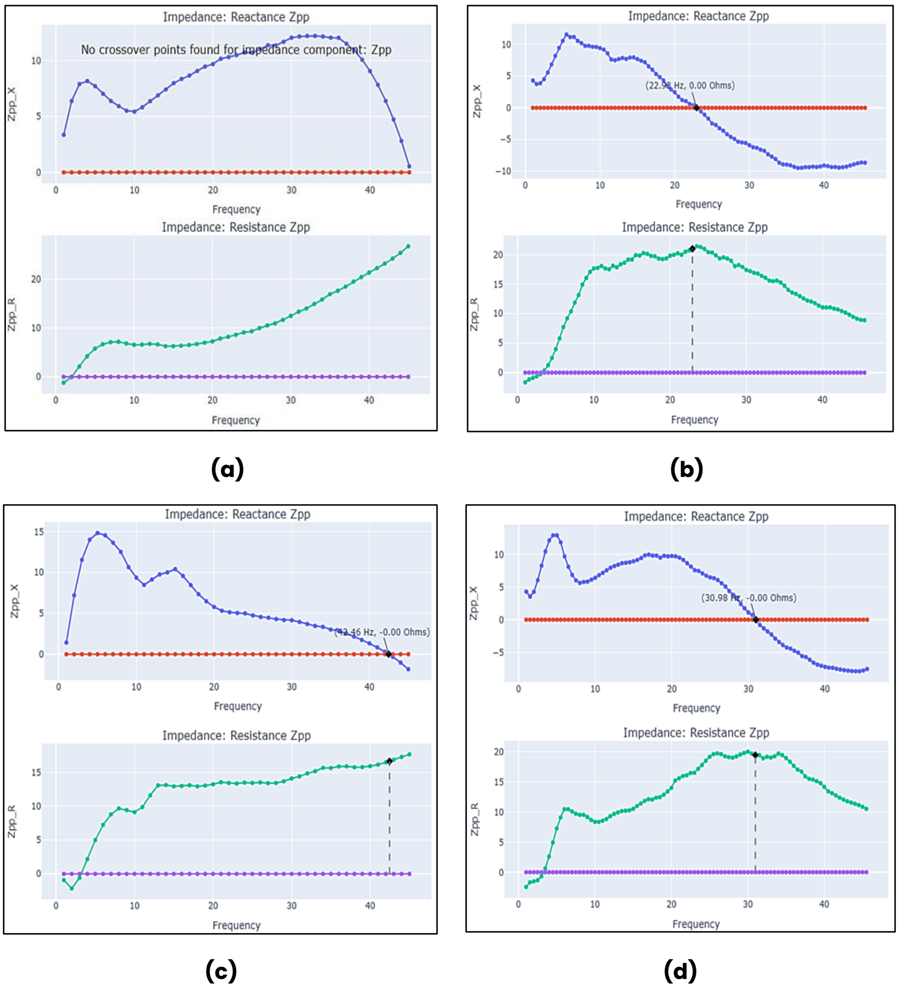
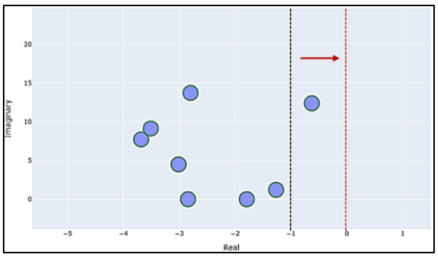
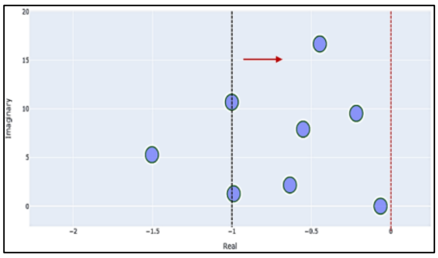


Figure 36 Reactance crossover calculations for (a) 1200MVA grid, measured at WF1, (b) 750MVA grid, measured at WF1, (c) 1200MVA grid, measured at WF2, and (d) 750MVA grid, measured at WF2.

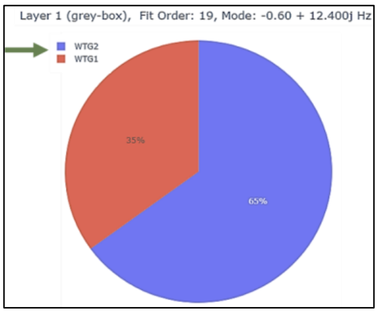
Commented [SM(13]: Figures c and d show system resistance less than 20 ohm. Even though the damping is positive, is this not a flag as far as threshold for reactance is concerned?

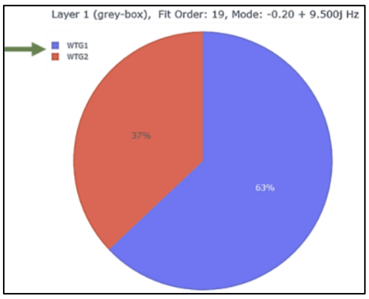
Commented [DC14R13]: This is flagged as SSO risk as this is less than our defined threshold.





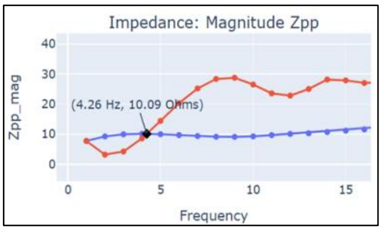
(a) **(b)**

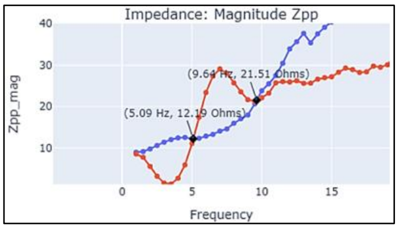




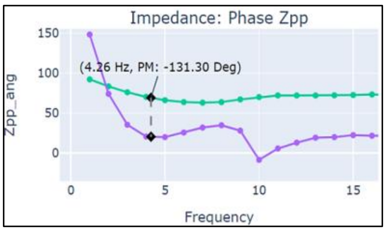
(c) **(d)**

Figure 37 Grey-box analysis calculations showing (a) whole system modes for 1200MVA grid, (b) whole system modes for 750MVA grid, (c) participation of WFs in 12.4Hz mode for 1200MVA grid, and (d) participation of WFs in 9.5Hz mode for 750MVA grid.

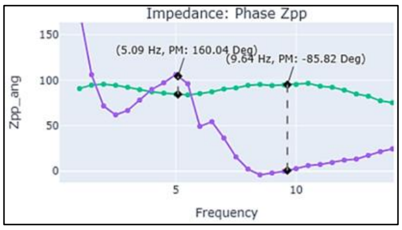




(a) **(b)**



(c)



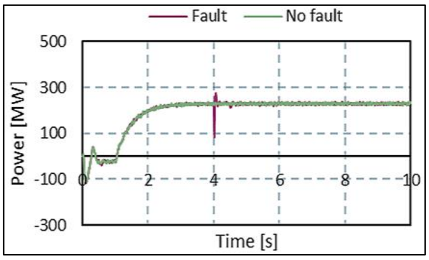
(d)

Figure 38 Impedance intersection calculations for; (a)&(c) 1200MVA grid, measured at WF1, (b)&(d) for 750MVA grid, measured at WF1.

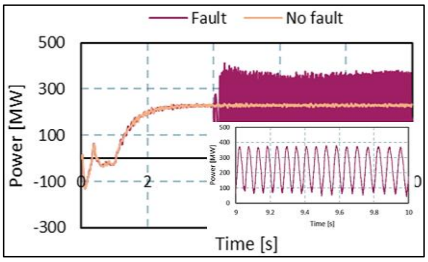
Commented [SM(15]: Apparatus side Impedance curve (red line) looks different here. For example at 15 Hz (fig a) the impedance is around 28 Hz. If we see impedance curve from apparatus scan of WF1 at 1200 MVA fault level shown in Fig 17, it is around 10 Hz. Why are they different? Am I missing something?

Commented [DC16R15]: Good point, let me check and get back to you.

From the frequency domain analysis, we have consensus between two methods out of three. Therefore, the 750MVA fault level scenario will be tagged as SSO risk. To verify the outcome of the screening method, EMT analysis is carried out for the two scenarios. A 60ms fault is applied to the WF1 export cable to excite the poorly damped mode. The results of the EMT analysis are presented in Figure 39. When no fault is applied, the system has a stable response, measured at both WFs, for 1200MVA and 750MVA grid fault level scenarios. Even after a fault is applied, the windfarms remain stable for the 1200MVA scenario (Figure 39 (a)&(c)). However, as the grid fault level reduces, the 60ms fault triggers an oscillation in the WF power outputs (Figure 39 (b)&(d)). In both cases, the system remains intact after the fault is cleared. Due to insufficient damping, even after the system goes back to its original configuration, the oscillations are sustained. This proves that the frequency domain methods are performing as expected and similar SSO risk scenarios will be identified by the tool for further EMT analysis.



(a)



(b)

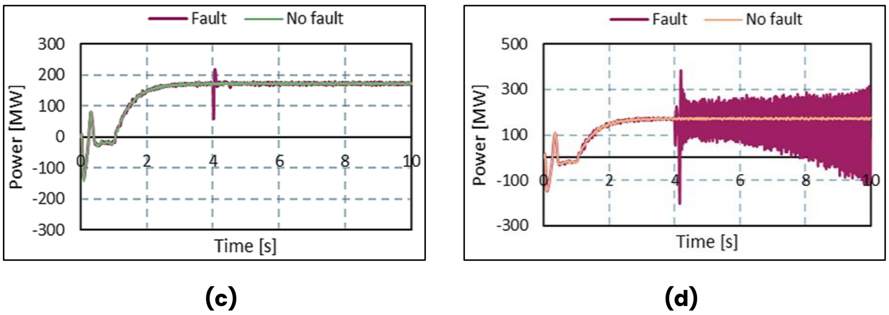


Figure 39 EMT simulation results for a 60ms fault at WF1 export cable for (a) 1200MVA grid measured at WF1, (b) 750MVA grid measured at WF1, (c) 1200MVA grid measured at WF2, and (d) 750MVA grid measured at WF2.

4.1.3 ML model oscillation detector

4.1.3.1 SETTINGS

The CIGRE benchmark model for multi-frequency analysis, developed by WG C4.49 in MATLAB Simulink, is used to generate the training data for the machine learning model. The model (Figure 31) has two windfarms with grid following inverters connecting to the grid by an AC export cable. The grid is modelled as a voltage source behind a grid impedance. An SSO event can be triggered in this model by applying a disturbance when the grid fault level is reduced $S_n \leq 1100$ MVA [13]. This means that by varying S_n, different scenarios with oscillations can be created.

The CIGRE benchmark model is simulated for the following grid fault levels: $S_n = \{3000, 1500, 1300, 1100, 1000, 900, 800, 750, 700\}$ MVA. The following events are applied; an increase of 0.02 pu in the grid voltage at 1s, the power outputs of the two wind farms are increased from 0 to 1 pu at 5 and 10s, the AC voltage reference of the inverter is decreased from 1.05 to 1.03 at 12s, and the DC reference voltage is reduced from 2 to 1.98 pu at 15s. Measurements are recorded with a sampling frequency of 10 kHz at three points (see Figure 31), WTG string (WF1), the point of coupling of the strings (POC), and the grid side (Grid). Both current and voltage are measured, resulting in six measurements being recorded per S_n value. A total of 54 measurement samples are obtained for the nine S_n values. The features are computed using a segment duration of 1 second for F1-F3 and a minimum frequency resolution of 1 Hz for F4. The thresholds are set to $\varepsilon_{c-sig} = 9, \varepsilon_{c-sim} =$

6 and $\varepsilon_{LSS} = 0.002$. The parameter m used in the F1 (Trend) calculation is iterated between 2 (repetitive pairs) and 5 (repetitive quintuplets). Labels are manually assigned by inspecting the spectrogram produced by each measurement sample to identify when the SSO event occurs.

4.1.3.2 OFFLINE TRAINING OF THE ENSEMBLE MODEL

This case study investigates the performance of the proposed training approach of the ML model. The ensemble model is trained offline using the 54 measurement samples generated from the CIGRE benchmark. The ML models are implemented in scikit-learn (Python). This dataset has imbalance of classes e.g. the samples for most fault levels S_n have more SSO events than non-SSO events. A stratified k-fold cross-validation strategy verified the model's robustness under different training and test data split percentages while accounting for the class imbalance. The cross-validation is repeated seven times, each with a uniquely shuffled version of the data, resulting in more data splits for testing the model's robustness. The following values are selected as candidates for $C = \{0.1, 1, 10, 100, 1000, 2000, 3000, 4000, 5000, 10000\}$ and $\gamma = \{\text{scale}, 0.2, 0.4, 0.6, 0.8, 1.0, 1.4, 2, 3, 4, 5, 10, 50, 100\}$; $\gamma = \text{scale}$ is the default value in scikit-learn. The BA is the selected performance metric as this metric considers class imbalances.

Table 1: Performance of the classifiers on predicting individual segments under different train-test splits.

n-splits (% train/test)	classifier features 1 – 3 (SVM) train BA	test BA	classifier feature 4 (3DT) train BA	test BA	Training time [s]
2 (50/50)	0.998 (± 0.002)	0.981 (± 0.009)	0.97 (± 0.021)	0.962 (± 0.026)	0.709
3 (67/33)	0.995 (± 0.002)	0.984 (± 0.009)	0.97 (± 0.015)	0.96 (± 0.037)	1.101
4 (75/25)	0.995 (± 0.002)	0.985 (± 0.009)	0.97 (± 0.009)	0.959 (± 0.039)	1.411
5 (80/20)	0.993 (± 0.002)	0.988 (± 0.009)	0.97 (± 0.009)	0.958 (± 0.047)	2.143

Table 1 shows the BA for different train-test splits. The better the BA on individual segments, the higher the chance to correctly predict the whole measurement sample. The SVM classifier has a test BA above 0.98 and shows little deviation across the different splits, indicating a good generalisation of performance. A higher percentage of training data slightly improves the BA. On the other hand, the classifier for F4 has a slightly lower BA, between 0.958 and 0.962, and has a greater (first) standard deviation, ranging from 0.026 to 0.047. A higher number of splits slightly reduces the BA for the test set. The feature extraction takes 68.8 seconds for all 54 of the 20-second measurement samples, while the training time ranges from 0.7 to 1.7 seconds on a Dell XPS 9510 with an Intel i7-11800H CPU, 32 GB RAM and an NVIDIA RTX 3050 Ti 4GB GPU. The following package versions are used: NumPy 1.26.2, pandas 2.1.4, SciPy 1.11.4 and scikit-learn 1.3.2.

4.1.3.3 ONLINE PREDICTION

This case study tests the online prediction with the ensemble model using measurements from a PSCAD model of the CIGRE benchmark system. Although the modelled system is the same, the data source is completely different (different software with modelling differences) having 20 kHz sampling rate that produces a different range for F1 and F3. F1 counts the number of sample points in the upward and downward-trending regions, and F3 uses the relative change between two sample points. Therefore, the scaler must be calibrated to the current feature range, as it would otherwise result in unseen feature values for the ensemble model. Moreover, different $\varepsilon_{c-sig}, \varepsilon_{c-sim}$ and ε_{LSS} thresholds for the features may also produce a clearer distinction between parts with SSO events and those without them. Therefore, we recommend inspecting two measurement samples with and without an SSO event to calibrate the features to each new measurement source by selecting the optimal thresholds.

The PSCAD data has six scenarios; a 60ms fault applied at WF1 for 3000 MVA and 750 MVA grid fault levels, a 60ms fault applied at WF2 for 3000 MVA and 750 MVA grid fault levels, and a 60ms fault applied on the export cable for 3000 MVA and 750 MVA grid fault levels. Calibrating the thresholds gives $\varepsilon_{c-sig} = 250$, $\varepsilon_{c-sim} = 25$ and $\varepsilon_{LSS} = 0.001$. $f_{min-res} = 2$ to reduce the segment duration of feature 4 to two seconds. $\varepsilon_{voting} = 0.3$, meaning that if the ensemble model predicts at least 30% of the individual segments as having an SSO event, the entire measurement is classified as SSO. The calibration is performed from samples at WF1, while the same thresholds and scaler are used for the other two locations. Figure 40 shows three of the six scenarios. With a voting system, the ensemble model can correctly predict the presence of SSOs from a different data source. The computation of the features on the 10-second simulation takes 1.5 seconds on average, while the ensemble model takes less than 1ms to predict the label.

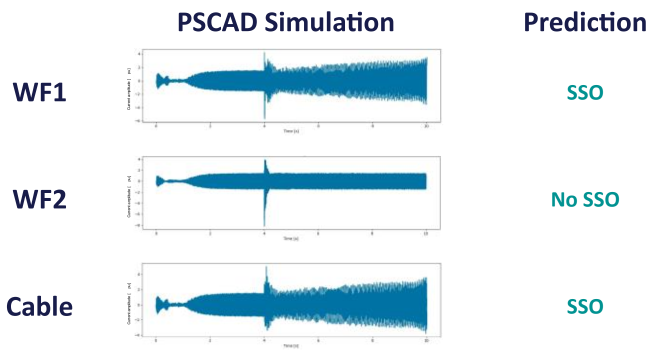


Figure 40 Online prediction by the ensemble model on three PSCAD cases. The ensemble model correctly predicts in all cases.

4.2 SE Coast study

This section presents the results obtained from implementing the SSO tool in the Southeast coast (SEC) EMT model. The SEC EMT network is a generic representation and does not depict any specific scenario. However, the voltage and angle of the boundary buses have been adjusted to reflect a power flow condition that ensures all bus voltages remain within acceptable limits. The purpose of this test is to evaluate the performance of the SSO tool with a User encrypted blackbox EMT model and identify any potential integration issues that may arise during the process.

It is important to highlight that the tool validation process will include subsequent steps, such as creating specific load conditions (winter/summer) and considering variations in the import/export of interconnectors and power output of wind farms. The SSO tool will then be utilized to identify any potential concerns related to Sub-Synchronous Control Interaction (SSCI) that may arise due to the presence of multiple Inverter Based Resources (IBRs) operating in close proximity.

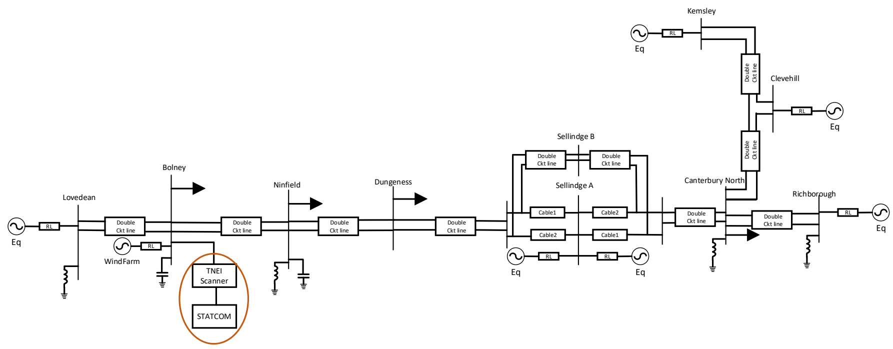


Figure 41: Simplified Single-line representation of the Southeast coast GB network.

3.2.1. Frequency Scan results

In this test, the apparatus under study is a VSC STATCOM of 225 MVAr rating connected at Bolney substation as shown in Figure 41. The STATCOM is operating in voltage control mode regulating the voltage at 400 kV level. The frequency scan is performed with 1% injection magnitude based on peak value of the phase voltage and voltages are injected in positive sequence one frequency at a time from 1 Hz to 45 Hz.

The frequency scan results are presented in Figure 42 and Figure 43. The impedance intersection between STATCOM and network shows no intersection at any frequency within the range. It is expected as the network mostly consists of ideal voltage sources and is relatively a strong network condition. The closed loop output impedance of the STATCOM at the selected operating mode and point is significantly higher than the network which signifies an efficient control design from a stability perspective. Further, the reactance crossover results shown in Figure 43 also shows reactance Zpp_X monotonically increasing in the studied frequency range. This concludes that the selected operating scenario is stable in the sense of sub-synchronous oscillation and does not present any SSO risks.

Figure 42: Full matrix impedance intersection results for the STATCOM and SEC network. Red trace in Zpp is the STATCOM impedance while the blue trace corresponds to the network.

Figure 43: Reactance crossover results obtained from the wholesystem scan . Blue trace in Reactance Zpp show the reactance of STATCOM while the red trace corresponds to network.

5 References

[1] T. F. Report, "Wind Energy Systems Sub-Synchronous Oscillations: Events and Modelir IEEE PES Resource Center, US, July 2020.

[2] ESO, "ESO Technical Report - FINAL2," Available [onlir https://www.nationalgrideso.com/document/152346/download , UK, 2019.

[3] A. Rygg, M. Molinas, C. Zhang and X. Cai, "A Modified Sequence-Domain Impedar Definition and Its Equivalence to the dq-Domain Impedance Definition for the Stabi Analysis of AC Power Electronic Systems," *IEEE Journal of Emerging and Selected Top in Power Electronics,* p. 1383–1396, 2016.

[4] S. Shah, P. Koralewicz, V. Gevorgian and R. Wallen, "Sequence Impedance Measureme of Utility-Scale Wind Turbines and Inverters - Reference Frame, Frequency Coupli and MIMO/SISO Forms," *IEEE Transactions on Energy Conversion,* p. 75–86, 2022.

[5] M. S. D. M. a. B. B. Y. Cheng, "Reactance scan crossover-based approach investigating SSCI concerns for DFIG-based wind turbines," *IEEE Transactions on Pov Delivery,* pp. 742-751, 2013.

[6] W. Ren and E. Larsen, "A Refined Frequency Scan Approach to Sub-Synchronous Cont Interaction (SSCI) Study of Wind Farms," *IEEE Transactions on Power Systems,* p. 390 3912, 2016.

[7] S. Shah and L. Parsa, "Impedance Modeling of Three-Phase Voltage Source Convert in DQ, Sequence, and Phasor Domains," *IEEE Transactions on Energy Conversion,* p. 113 1150, 2017.

[8] J. Sun, "Impedance-based stability criterion for grid-connected inverters," *I Transactions on Power Electronics,* p. 3075–3078, 2011.

[9] F. B. a. P. C. L. X. Wang, "An impedance-based stability analysis method for parallel voltage source converters," in *International Power Electronics Conference (IPE Hiroshima 2014 - ECCE ASIA)*, Hiroshima, Japan, 2014.

[10] Y. Zhu, Y. Gu, Y. Li and T. C. Green, “Participation Analysis in Impedance Models: The Gre
Box Approach for Power System Stability,” *IEEE Transactions on Power Systems,* vol.
no. 1.

[11] R. A. D. C. a. C. H. Wells, “Distributed Real-Time Electric Power Grid Event Detection a
Dynamic Characterization,” in *Grid of the Future Symposium, 2014: CIGRE.* , 2014.

[12] F. Zhang, “Simulated Synchrophasors in SSOs and Phasor Measurement Data Record
During a Subsynchronous Oscillation Incident in an Actual Power System,” IEEE, Chi
2022.

[13] Ł. K. e. al, “Practical Aspects of Small-signal Stability Analysis and Instability Mitigatic
in *21st Wind & Solar Integration Workshop*, The Hagaue, Netherlands, 12-14 October 2(
.

[14] Manitoba Hydro International Ltd., “Sub Synchronous Oscillations SSO Investigation a
Applications,” Manitoba Hydro International Ltd., 2016 November 2016. [Onlir
Available: https://www.youtube.com/watch?v=d6FZxZYTLg8&t=2282s. [Accessed
September 2022].

[15] R. N. Damas, Y. Son, M. Yoon, S.-Y. Kim and S. Choi, “Subsynchronous Oscillation a
Advanced Analysis: A Review,” *IEEE Access,* vol. B, pp. 224020-224032, 2020.

[16] D. Kidd and P. Hassink, “Transmission Operator Perspective of Sub-Synchronc
Interaction,” in *PES T&D*, Orlando, 2012.

[17] A. Ostadi, A. Yazdani and R. Varma, “Modeling and Stability Analysis of a DFIG-Bas
Wind-Power Generator Interfaced With a Series-Compensated Line,” *IEEE Transactic
on Power Delivery,* vol. 24, no. 3, pp. 1504-1514, 2009.

[18] X. Xie, X. Zhang, H. Liu, Y. Li and C. Zhang, “Characteristic Analysis of Subsynchronc
Resonance in Practical Wind Farms Connected to Series-Compensat
Transmissions,” *IEEE Transactions on Energy Conversion,* vol. 32, no. 3, pp. 1117-1126, 2(

[19] M. Beza and M. Bongiorno, “On the Risk for Subsynchronous Control Interaction in Ty
4 Based Wind Farms,” *IEEE Transactions on Sustainable Energy,* vol. 10, no. 3, pp. 14
1418, 2019.

[20] M. Sahni, D. Muthumuni, B. Badrzadeh, A. Gole and A. Kulkarni, "Advanced Screeni Techniques for Sub-Synchronous Interaction in Wind Farms," in *PES T&D*, Orlando, 20

[21] B. Badrzadeh, M. Sahni, D. Muthumuni, Y. Zhou and A. Gole, "Sub-synchronous Interact in Wind Power Plants - Part I: Study Tools and Techniques," in *2012 IEEE Power and Ener Society General Meeting*, San Diego, 2012.

[22] M. Sahni, B. Badrzadeh, D. Muthumuni, Y. Cheng, H. Yin, S.-H. Huang and Y. Zhou, "Su synchronous Interaction in Wind Power Plants - Part II: An ERCOT Case Study," in *2 IEEE Power and Energy Society General Meeting*, San Diego, 2012.

[23] B. Badrzadeh, M. Sahni, Y. Zhou, D. Muthumuni and A. Gole, "General Methodology Analysis of Sub-Synchronous Interaction in Wind Power Plants," *IEEE Transactions Power Systems,* vol. 28, no. 2, pp. 1858-1869, 2013.

[24] X. Jiang and A. Gole, "A Frequency Scanning Method for the Identification of Harmo Instabilities in HVDC Systems," *IEEE Transactions on Power Delivery,* vol. 10, no. 4, pp. 187 1881, 1995.

[25] T. Services, "WP1 Report," TNEI Services, Manchester, 2023.

[26] B. G. a. A. Semlyen, "Rational approximation of frequency domain responses by vec fitting," *IEEE Trans. Power Delivery,* vol. 14, no. 3, July 1999.

[27] A. Arsenovic, "scikit-rf: An Open Source Python Package for Microwave Netw Creation, Analysis, and Calibration [Speaker's Corner],," *IEEE Microwave Magazine,* v 23, no. 1, Jan 2022.

[28] Y. Cheng, M. Sahni, D. Muthumuni and B. Badrzadeh, "Reactance scan crossover-bas approach for investigating SSCI concerns for DFIG-based wind turbines," *I Transactions on Power Delivery,* p. 742–751, 2013.

[29] Y. Zhu, Y. Gu, Y. Li and T. C. Green, "Impedance-Based Root-Cause Analysis: Comparat Study of Impedance Models and Calculation of Eigenvalue Sensitivity," *I Transactions on Power Systems,* vol. 38, no. 2.

[30] B. Guvstavsen, "The Vector Fitting Website," [Online]. Availab www.sintef.no/projectweb/vectorfitting.

[31] H. D. S. B. Salarieh, "Review and comparison of frequency-domain curve-fitti techniques: Vector fitting, frequency-partitioning fitting, matrix pencil method a loewner matrix," *Electric Power Systems Research.*

[32] B.-G. R. B. C. K. J. L. M. T. A. S. Y. W. X. Kocewiak Ł, "Instability mitigation methods in mode converter-based power systems," *20th International Workshop on Large-Sc Integration of Wind Power into Power Systems as well as on Transmission Networks Offshore Wind Power Plants (WIW 2021),* pp. 464-480, 2021.

[33] B.-G. R. B. C. K. J. L. M. S. Y. W. X. Kocewiak L, "Practical Aspects of Small-signal Stabi Analysis and Instability Mitigation," 2022.

[34] IEEE, "First benchmark model for computer simulation of subsynchronous resonanc *IEEE Transactions on Power Apparatus and Systems,* vol. 96, no. 5, pp. 1565-15 September 1977.

Appendix B: Review of impedance scan methods

1. A Modified Sequence-Domain Impedance Definition and Its Equivalence to the dq-Domain Impedance Definition for the Stability Analysis of AC Power Electronic Systems [2]

- The main contribution of the paper is a method to determine the sequence impedance of a Source-Load system, enabling the identification of frequency coupling in the sequence domain.
- Frequency coupling has an impact on the stability performance of the Source-Load system. **A system with frequency coupling will be less stable in comparison with a mirror frequency decoupled (MFD) system**.
- Sources of frequency coupling in IBRs are PLL, asymmetrical implementation of dq axis controllers, DC-bus voltage control, and active and reactive power controllers.
- Sources of frequency coupling on synchronous machines are rotor saliency, poorly tuned PSSs, and frequency and voltage control loops.
- The eigenvalues of the d-q impedance matrix are equal to those obtained by the modified sequence domain impedance (MSDI) matrix.

Key features of the reviewed method

- The method relies upon the definition of the modified sequence domain impedance (MSDI) matrix, which uses the relation between phasors in the dq domain with the corresponding phasors in the sequence domain.
- The off-diagonal components of the MSDI matrix Z_{pn} (below) allows the identification of the influence of a positive sequence current on the negative sequence voltage, and vice versa.
- The eigenvalues of the dq impedance matrix are equal to the MSDI matrix.
- Figure B. 1 shows the circuit equivalents of the impedance matrices in both domains. If a system is mirror frequency decoupled the off-diagonal components of $\boldsymbol{Z_{pn}}$are zeros; the current dependant voltage sources in the sequence domain are not necessary.

- Simulation results showed that systems which have mirror frequency coupling have less stability margin in comparison with those mirror frequency decoupled. Sources of frequency coupling are PLL; converter current controllers with unequal structure of parameter values in the d and q axes; dc-link voltage control systems; active and reactive power controllers; and salient poles synchronous machines.
- A drawback is that the authors did not specify the selection of the perturbation signals. They only mentioned that the two perturbation signals must be linearly independent; for that, the perturbations defined in dq domain must have different frequencies.
- Another drawback is the lack of explanation of the stability analysis process. There is no clear procedure to process the MIMO system (2x2 sequence impedance matrix), to produce the stability analysis (multiple Nyquist plots) and the application of the generalised Nyquist criteria.

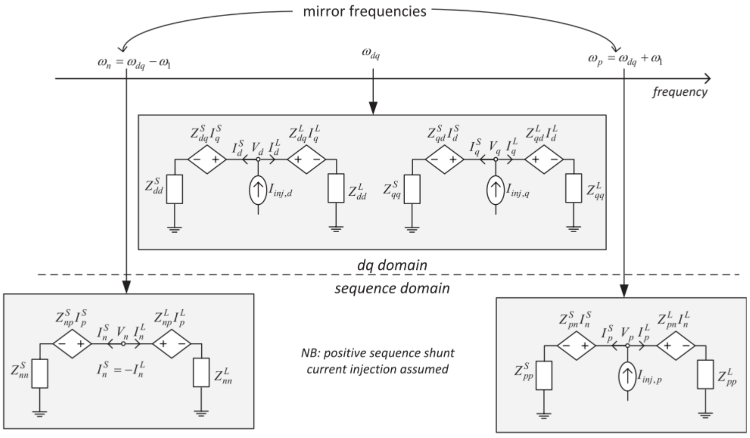


Figure B. 1 Circuit representation of frequency coupling using dq and sequence domain.

Summary of the proposed method

1. The method requires the injection of two test signals in abc domain.
2. Storing the time-series response (voltage and current) of source and load subsystems.
3. Transform the stored signals to dq domain and perform FFT.
4. Extract the magnitude of the components at the frequency of interest.
5. Calculate the dq impedance matrix.

6. Obtain the sequence impedance matrix using *dq* to *abc* transformation.

2. Sequence Impedance Measurement of Utility-Scale Wind Turbines and Inverters - Reference Frame, Frequency Coupling, and MIMO/SISO Forms [3]

- Active devices like IBRs have a frequency-coupled response due to their configuration; in practical terms, an injection of a positive-sequence voltage perturbation produces a negative-sequence current, in addition to the positive-sequence response. Therefore, the off-diagonal elements of a matrix transfer function (impedance or admittance) will likely be not zero for IBRs.
- The frequency coupling is significant at low frequencies, below a couple of hundred Hertz; therefore, the SSO analysis must be considered in the screening method.
- The sequence impedance of active devices has a reference frame. To ensure the phase alignment with the phase-A voltage, a correction of the Fourier components was introduced. The correction aligns the starting point of the data window of the FFT with the fundamental component of phase A voltage.

Key features of the reviewed method

- The method enables the computation of the sequence domain admittance matrix containing frequency coupling components (off-diagonal terms).
- The 2x2 matrix is a MIMO system that can be reduced to a SISO system. If the impedance from the grid side Z_g (see Figure B. 2)does not show frequency coupling, it is possible to reduce the system to use Nyquist and Bode plots to study the stability of the system.
- The method provides a compensation procedure to align the FFT components of the data corresponding to the measurements, with the peak of the fundamental component of the phase A voltage.

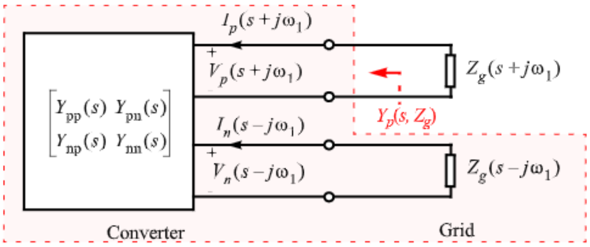


Figure B. 2 SISO representation of the system.

Summary of the proposed method

1. Select a frequency s and inject two test signals in the sequence domain. One at positive sequence frequency $(s + j\omega_1)$ and another in negative sequence frequency $(s - j\omega_1)$.
2. Perform FFT analysis and extract 4 sets of components for both injections.

- For the first injection; positive sequence injection.
- Voltage $V_p(s + j\omega_1)_{(1)}$ at positive sequence perturbation frequency
- Current $I_p(s + j\omega_1)_{(1)}$ at positive sequence perturbation frequency
- Voltage $V_n(s + j\omega_1)_{(1)}$ at negative sequence perturbation frequency
- Current $I_n(s + j\omega_1)_{(1)}$ at negative sequence perturbation frequency
- Repeat for the second injection, negative sequence injection.

3. Estimate the correction factor and apply the compensation to the phases of Fourier components.
4. Obtain the sequence domain admittance matrix Y_{PN}
5. Select another frequency s from the frequency interval of interest.

3. Impedance Modelling of Three-Phase Voltage Source Converters in DQ, Sequence, and Phasor Domains [6]

- The coupling between grid side impedance (AC and DC) and DC side impedance cannot be neglected. The coupling plays an important role in low-frequency resonance, which is not captured by decoupled methods.

- The stability analysis of the frequency response converting sequence domain into dq domain is crucial to produce an accurate assessment.
- Poorly tuned PLL introduce unstable resonance between the control loop and the grid impedance. Those comparisons are possible using the proposed linear transformation between different domains (sequence, dq, and phasor)

4. Impedance-based stability criterion for grid-connected inverters [4]

- The effects of grid impedance on grid-connected inverter stability can be studied by using a simple small-signal model involving the grid impedance and the inverter output impedance.
- Under the assumption that the inverter is controlled as a current source and is stable when operating with an ideal grid, it will remain stable with a nonideal grid if the ratio of the grid impedance to the inverter output impedance satisfies the Nyquist criterion.
- The analysis also indicates that grid-connected inverters should be designed to have as high output impedance as possible to be able to operate with a wide range of grid impedances.

Key features of the reviewed method

- The method provides a stability assessment criterion based on grid and IBR output impedance scan.
- Once the grid side and IBR impedances are obtained from frequency scans, the bode plot of both impedances can be computed. The phase difference needs to be checked at frequencies where the magnitude of the grid side impedance is equal to the IBR impedance. If the phase difference between the grid side and the IBR side is closer to 180 degrees, the system will have lower damping for harmonics at that frequency.

5. Reactance scan crossover-based approach for investigating SSCI concerns for DFIG-based wind turbines [7]

- Decoupled analysis for WTG side and system side. The analysis focuses on the crossover reactance on the network side (driving point reactance scan) to define a potential SSO, checking the resistance of the WTG side.
- The analysis showed potential SSO not only for radial connections but also for meshed (close to radial).
- There is a higher SSO risk for scenarios with faults, in comparison to those only with outages.

Key features of the reviewed method

- The method provides a tool to identify potential SSCI by combining two scans. The driving point scan to obtain the system impedance is performed using PSCAD harmonic injection (Method 1 in Figure B. 3). The method of harmonic injection is suitable for systems with active components like WTG.
- If the system reactance has crossover frequencies, which means that the reactance crosses from positive to negative, and vice-versa, the system may be at risk of SSO.
- The system reactance scan is combined with the WTG scan. In this case, the WTG is disconnected from the system and connected to an ideal source.
- The condition for SSO risk for a given system condition (dispatch, network outages, etc) can be defined as follows "If there are reactance crossover on the system and those frequencies have negative resistance on the WTG scan, then there is a risk of SSO" (see Figure B. 4).
- The method is simple and easy to implement and interpret.
- The method requires performing two sets of harmonic injections; one with the system with the WTG, and another with the WTG connected to an ideal voltage source. The drawback is the amount of computation required to perform the scans.

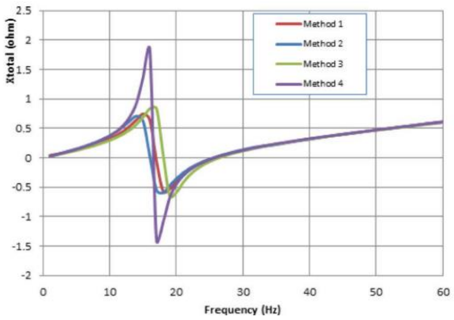


Figure B. 3 Total reactance of the network

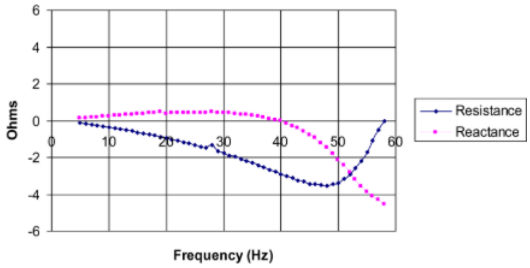


Figure B. 4 Resistance and reactance of the WTG.

Summary of the proposed method

1. Define system condition. Introduce changes on WTG dispatch or the status of branch elements in the network.
2. Perform a harmonic impedance scan and obtain the system reactance as a function of frequency.
3. Perform a harmonic impedance scan of the turbine connected to an ideal voltage source and obtain the WTG resistance as a function of frequency.
4. Verify the if the frequency of the system reactance crossover also has a negative resistance on the WTG impedance. If so, the system has a potential risk of SSO under the given condition.

6. A Refined Frequency Scan Approach to Sub-Synchronous Control Interaction (SSCI) Study of Wind Farms [5]

- A detailed mathematical explanation of frequency coupling due to d-q saliency. In the analysed DFIG WTG the effect of different gain values Kd and Kq results in the creation of new harmonics components (frequency coupling).
- The derivation of impedance considering frequency coupling improves the accuracy of the detection of potential SSO risks.

Key features of the reviewed method

- The method can identify when there is a source of mirror frequency coupling on the WTG side. The inclusion of a term accounting for frequency coupling makes the method more accurate in comparison with standard methods that assume that the frequencies are decoupled.
- The method is easy to implement, but it is computationally intensive; it requires two perturbations for each frequency under study. However, it does not need to scan the grid and WTG independently.
- The new method (considering frequency coupling) also uses the reactance crossover frequency based on the total impedance to identify SSO risk (see Figure B. 5).

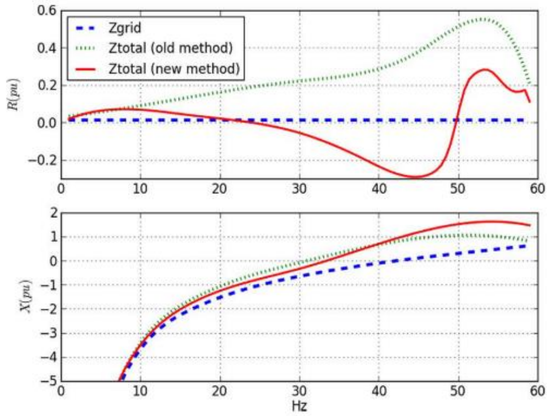


Figure B. 5 Resistance and reactance of the whole system.

Summary of the proposed method

1. Establish a steady state operating condition of the system under test (e.g., power, and reactive power out of the WTG (see Figure B. 6), terminal voltage, frequency). Record the steady state quantities with a known Z^1_{test}.

2. Inject a perturbation signal at $f1$, and measure voltage and current at the same frequency.
3. Inject another perturbation at $f2$, which must be at the coupling frequency of $f1$, and measure the voltage and current at the same frequency.
4. Repeat until $f1$ sweeps the range of frequencies under study.
5. Define another known test impedance Z_{test}^2 and repeat the previous injection steps.
6. Solve the equation for self-impedances.
7. Solve the equation for coupling impedances.
8. Finally, obtain the total system impedance.

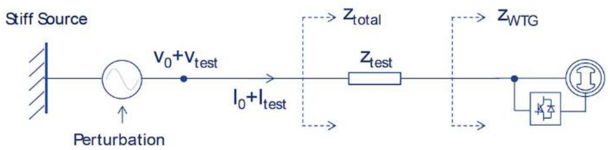


Figure B. 6 Setting for WTG scan.