

ALMA MATER STUDIORUM - UNIVERSITY OF BOLOGNA

SCHOOL OF ENGINEERING

DEPARTMENT of
ELECTRICAL AND INFORMATION ENGINEERING
“Guglielmo Marconi”
DEI

MASTER’S DEGREE
IN ELECTRICAL ENERGY ENGINEERING

CURRICULUM “ELECTRICAL ENGINEERING”

MASTER THESIS
in
APPLIED MEASUREMENTS FOR POWER SYSTEMS

**Real-time PMU monitoring of an IEEE 5-Bus Network
implemented on OPAL system**

CANDIDATE
Federica Costa

SUPERVISOR
Prof. Lorenzo Peretto

CO-SUPERVISORS
Prof. Roberto Tinarelli
Ing. Alessandro Mingotti

Academic Year
2019/2020

Session II

*“When you can measure what you are speaking about,
and express it in numbers, you know something about it”
- Lord Kelvin*

Acknowledgements

I would like to thank my family for the constant encouragement and support given to me during these years at the University and my supervisors for the precious advices.

Table of Contents

Introduction.....	6
Chapter 1: Phasor Measurement Units	8
1.1 Overview of PMUs	8
1.1.1 Working principle of PMUs.....	8
1.1.2 Role of PMUs in the network	10
1.1.3 Synchrophasors, frequency and ROCOF	11
1.2 Evaluation of Measurements.....	13
1.2.1 TVE, FE and RFE	13
1.2.2 Measurement responses and reporting rates.....	14
1.3 Compliance Verifications	15
1.3.1 Steady-state requirements	15
1.3.2 Measurement latency compliance	17
1.4 Synchrophasor Networks	18
1.4.1 Measurement Systems.....	18
1.4.2 Phasor Data Concentrator	19
1.4.3 Communication System (IEC 61850)	19
1.4.4 Cyber Security issues	22
Chapter 2: Power Quality Measurements	24
2.1 Power Quality objectives	24
2.2 PMUs for Power Quality.....	25
Chapter 3: Real-Time Simulations.....	28
3.1 Real Time Implementations	28
3.1.1 What are Real Time Simulations	28
3.1.2 Application categories.....	31
3.1.3 Importance of RT tests and validation	33
3.2 HIL Synchrophasor testing	34
3.2.1 PMU testing architectures	34
3.2.2 PMU compliance procedures	36

3.2.3 Other HIL tests for PMUs	38
Chapter 4: Practical Implementation	39
4.1 Topology observability rules	39
4.2 Optimization of PMU placement and algorithms	40
4.3 Implementation of the case study.....	41
4.3.1 The IEEE 5-bus system and load flow solution	41
4.3.2 Optimization of PMU placement	43
4.4 Simulink code and practical tests.....	44
4.4.1 Implementation of a PMU in Simulink.....	44
4.4.2 Analysis of the PMU block.....	48
4.4.3 Preliminary test results.....	55
4.4.4 Computation of the accuracy indices	62
4.5 OPAL Real-Time simulator.....	71
4.5.1 Basic structure.....	71
4.5.2 Re-arrangement of the Simulink code.....	74
4.5.3 Analysis of the real-time data.....	83
4.5.4 Final results and considerations	90
Chapter 5: Future perspective on PMUs and DT	96
5.1 Digital Twin concept.....	96
5.2 Future trends for PMUs and DT implementation	98
5.3 Expected outcomes	99
Conclusions.....	102
List of Abbreviations and Acronyms	104
List of Figures.....	107
List of Tables	111
Bibliography	112

Introduction

In the recent years, Phasor Measurement Units (PMUs) have become more and more employed in the network as they can tackle most of the problems originated by the wide deployment of Distributed Generation (DG).

Given their importance for the future of electric power systems, this thesis proposes a first step study towards the real-time validation of PMUs performances in the grid as well as the authentication of their results.

At the beginning, the state-of-the-art and the working principle of PMUs are studied as well as its metrological requirements stated in the IEEE C37.118.1 and C37.118.2 Standards for guaranteeing correct measurement performances. Communication systems among PMUs and their possible applicability in the field of power quality (PQ) assessment are also investigated.

This preliminary review is followed by an analysis of the working principle of real-time (RT) simulators, especially in the validation of models as their goal is to combine flexibility and accuracy of digital simulation techniques with the real time response [1].

The importance of hardware-in-the-loop (HIL) implementation is then considered, specifically the examination of possible case studies specific for PMUs, including compliance tests which are one of the most important parts.

The core of the thesis is then focused on the implementation of a PMU model in the IEEE 5-bus network in Simulink and in the validation of the results using OPAL RT-4510 as a real-time simulator.

An initial check allows one to get an idea about the goodness of the results in Simulink, comparing the PMU data with respect to the load-flow steady-state information. In this part, the most frequently used accuracy indices are also calculated for both voltage and current synchrophasors.

The following part is the most relevant one, as it consists in the re-arrangement of the code and in its implementation in OPAL-RT 4510 simulator, after which an initial analysis is carried out in a qualitative way in order to get a sense of the goodness of the outcomes.

Finally, the confirmation of the results is based on a numerical examination of the attained voltage and current synchrophasors, and accuracy indices coming from Simulink models and from OPAL system, using a Matlab script.

In this last part, the validation of the findings is also proven by checking that no overruns occurs during the simulation and the real step-size used is the correct one.

Lastly, this work also intends to propose suggestions for an upcoming operation of PMUs in a more complex system as the Digital Twin (DT) in order to improve the performances of the already-

existing protection devices of the distribution system operator (DSO) for a future enhancement of power systems reliability.

Chapter 1 – Phasor Measurement Units

1.1 Overview of PMUs

1.1.1 Working principle of PMUs

A Phasor Measurement Unit (PMU) is a digital device which provides synchronized voltage and current phasor measurements, referred to as “synchrophasors” [2].

In figure 1, the basic structure of a PMU is represented.

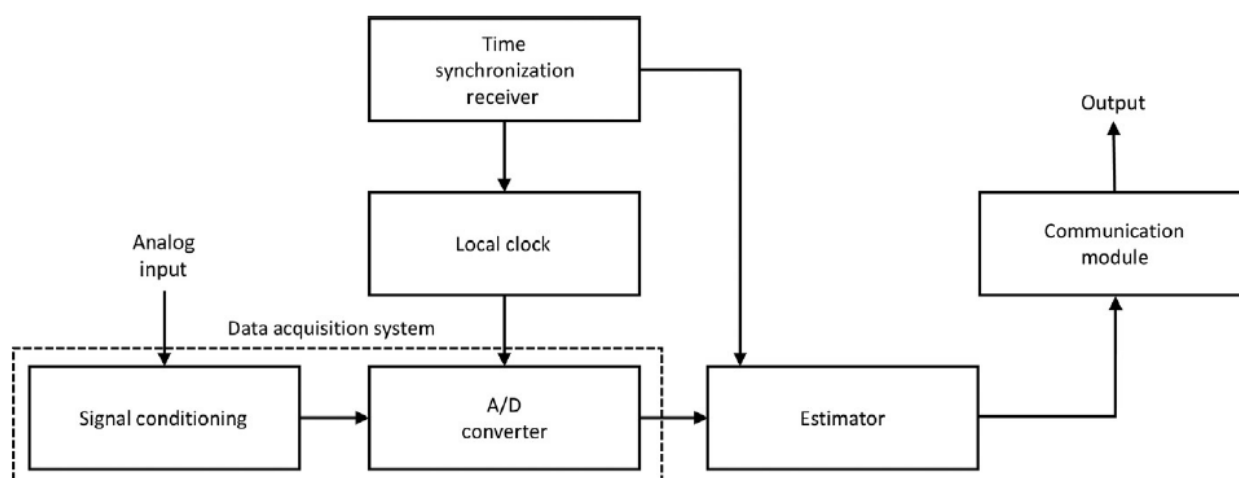


Figure 1: Basic PMU structure. Adapted from [3]

At the installation bus, instrument transformers (ITs) such as current and voltage transformers (CTs and VTs) are needed in order to measure the three-phase quantities. These analog signals are converted into digital by means of an analog to digital converter (ADC) with a sampling rate usually varying from 12 to 128 samples per cycle of the nominal power frequency [2].

The sampling clock is phase-locked with the Global Positioning System (GPS) clock pulse which provides the UTC time reference used to time-tag the outputs. The time of the measurement consists of three numbers: a second-of-century (SOC), a fraction-of-second (FRACSEC) count and a message time quality flag [4]. This time-tag is sent out with the phasors, thus if a phasor information packet arrives at the Phasor Data Concentrator (PDC) out of order, the phasor time response can still be assembled correctly. Whereas, if the GPS pulse is not received for a while, then the time-tagging error may result in a significant phase error [6].

Phasors of phase voltages and currents are computed from the sampled data by the PMU microprocessor according to a signal processing technique which is described as follows [2].

To estimate a phasor, continuous sampling at a minimum rate of 600 to 6.4 kHz, for a nominal frequency of the system of 50 Hz, is performed. To reconstruct the signal, the easiest method

consists in using the Discrete Fourier Transform (DFT) allowing to get the magnitude and phase of the signal for each phase of the three-phase quantities.

The signal phasor is computed in a continuous process from successive samples in a moving data window of one or several fundamental cycles, in order to reduce the noise.

When the frequency varies by a small amount around its nominal value, the leakage error introduced in phasor estimates can be compensated with high accuracy by a post-processing filtering.

The calculated phasors are combined in order to form the positive, negative and zero sequences: the latter are necessary in case of unbalanced conditions. Other relevant quantities which are measured are the frequency and the Rate Of Change Of Frequency (ROCOF).

Computed phasor measurements are transmitted through a digital communication network to higher level applications at a rate of 10 up to 60 frames per second. This phasor data from several PMUs is then collected by a special-purpose computer called Phasor Data Concentrator (PDC) which correlates the phasor data by time stamp to create a system-wide measurement set.

The IEEE Standards C37.242 [5] for synchronization, calibration, testing and installation of PMUs and C37.244 [6] for PDC requirements have been recently published in order to assist users to specify the performance and the functional requirements of typical PMUs and PDCs. PDCs may also interface directly to upper level PDCs, Energy Management Systems (EMS) or a Supervisory Control And Data Acquisition (SCADA) system [2], [7].

PMU devices may also be synchronized with each other using a Precision Time Protocol (PTP), defined in the IEEE 1588 Standard, particularly in substations where the series of IEC 61850 Standards have been introduced [7], [8].

The performances of PMUs in terms of accuracy and processing time are dictated by its components: a major role is played by the instrumentation channel, the ADC and the parameters of the phasor estimation algorithm.

Theoretically, PMU data are time-tagged with a precision better than 1 microsecond and a magnitude accuracy better than 0.1%, but in practice this is rarely achieved due to the uncertainties coming from the instrumentation channel.

The most recent IEEE C37.118-2014 standard defines the synchrophasor convention and the time-tagging process as well as it provides a definition of an accuracy measure and the requirements for measurement performance under steady-state conditions [9].

According to the previous stated Standard, PMUs can be classified into two classes of performances: P Type (Protection applications requiring fast response) and M Type (Measurement

applications requiring high precision). IEEE C37-118.1 specifies both steady-state and the dynamic performance compliance criteria for each class of PMUs [10].

Sinusoidal waveform distortions are the major difficulty in the phasor estimation: a wavelet analysis could first discriminate the discontinuities, followed by an adaptive window algorithm to estimate the phasor quantities. The accuracy of phasor measurement under dynamic conditions was improved by dynamic phasor estimates based on the least-square method, which is one of the many algorithms available in literature [7].

Harmonic distortion as well as the impact of switching load on the frequency estimation have led to the extension of synchrophasor algorithms to provide PMUs with the capability of estimate harmonic phasors especially in active distribution networks [7].

It has to be highlighted that there is no standard phasor algorithm implemented in PMUs, so there exists alternatives to the DFT which have been proposed in literature, such as Kalman filtering and neural networks. Moreover, the window length, the sampling rate, the phasor estimates reporting rate, the communication protocol as well as the measurement accuracy are all distinctive to each PMU device [7].

In the next section, particular relevance will be given to the evaluation of measurements performed by PMUs and the compliance limits provided by IEEE Standards.

1.1.2 Role of PMUs in the network

The ongoing increase of DG, Plug-in Electric Vehicles (PEVs), microgrids and power electronic components in network is kicking off new challenges to be tackled. Among which, DG causes troubles regarding both the planning and the operation of the network, as stability issues; uncontrolled PEV charging can lead to a violation of the lower limit for the voltage and the overloading of the distribution system, whereas harmonic injection from power electronic components such as electric drives, Distributed Energy Resources (DER) and PEV inverters can increase the Total Harmonic Distortion (THD), modifying conventional voltage and current level patterns which must stay within certain fixed limits [11].

This is why PMUs are regarded as a valuable resource in the framework of Wide Area Monitoring, Protection, Automation and Control (WAMPAC) for the management of distribution grids.

Among the several applications of PMUs, the most relevant ones for the correct behaviour of the distribution network would concern the real-time monitoring and measuring of relevant quantities.

Firstly, dynamic monitoring and protection relates to the detection of islanding operation of DG in order to avoid issues such as the reclosing out of synchronism or temporary overvoltages (TOV):

this would be carried out by two PMUs providing their measurements to a voltage angle difference algorithm. Moreover, PMUs would also be efficient in the monitoring of electromechanical transients as synchronous generators which are used in both conventional centralized and distributed generation [11].

As far as fault location and detection is concerned, an important role can be played by PMUs since they can overcome the issue of overcurrent protection devices which struggle in detecting high impedance faults. PMUs can compare pre- and post-fault data in order to detect the moment in which the transient condition has started, taking advantage of the sampling windows [12].

Finally, PMUs can be helpful in the estimation of harmonic levels. Both DG and PEV inverters must comply with certain limits, but their large-scale presence in the network may cause impacts on the distribution system. In order for the Distribution System Operator (DSO) to keep the THD under control, PMUs can be optimally placed using optimization algorithms to minimize costs and ensure a complete observability of the network at the same time. This will also prevent overloads in communications and data storage facilities [11].

1.1.3 Synchrophasors, frequency and ROCOF

The phasor representation of sinusoidal signals is commonly used in AC power systems. The sinusoidal waveform is defined in equation (1) as:

$$x(t) = X_m \cos(\omega t + \phi) \quad (1)$$

Its corresponding phasor representation is represented in equation (2):

$$\mathbf{X} = \frac{X_m}{\sqrt{2}} e^{j\phi} = \frac{X_m}{\sqrt{2}} (\cos\phi + j\sin\phi) = X_r + jX_i \quad (2)$$

Where the magnitude is the root-mean-square (RMS) value ($X_m/\sqrt{2}$) of the waveform and the value of ϕ depends on the time scale, particularly when $t = 0$. It's also important to notice that this phasor is defined for the angular frequency ω [9].

According to IEEE Standard C37.118.1 [9], the synchrophasor representation of the signal $x(t)$ in equation (1) is the value \mathbf{X} in equation (2) where ϕ is the instantaneous phase angle relative to a cosine function at the nominal system frequency synchronized to the Coordinated Universal Time (UTC). Therefore, under this definition, ϕ is the offset from a cosine function at the nominal system frequency synchronized to UTC.

The convention for synchrophasor representation is illustrated in figure 2:

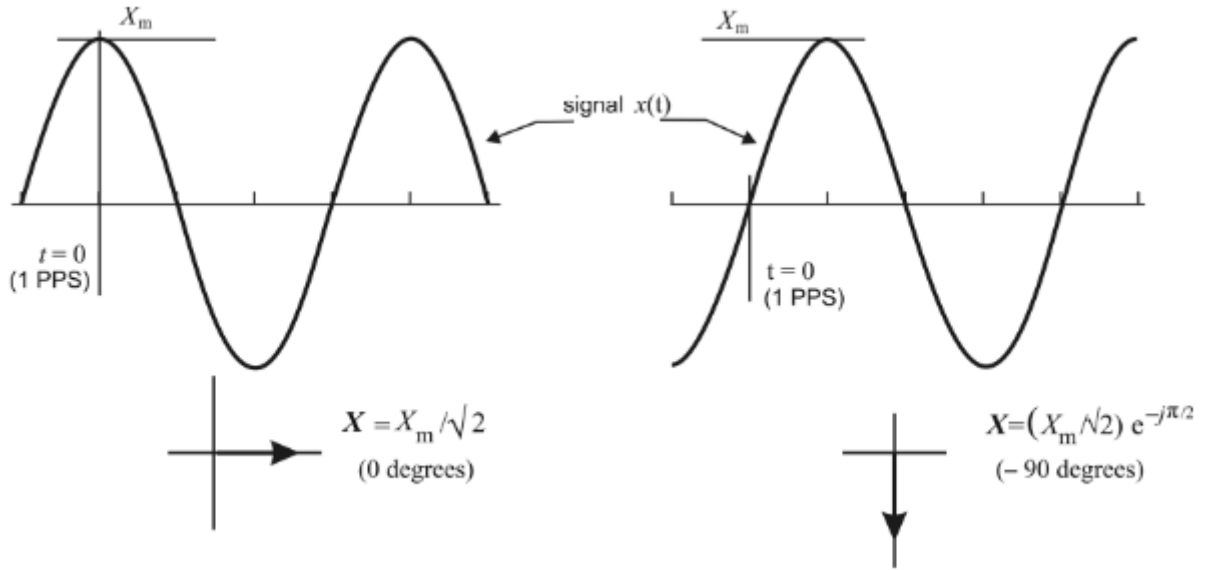


Figure 2: Convention for synchrophasor representation. Adapted from [9]

A PMU shall calculate and be capable of reporting frequency and ROCOF. For this measurement, the following standard definitions shall be used. Given a sinusoidal signal, as shown in equation (3):

$$x(t) = X_m \cos [\Psi(t)] \quad (3)$$

Frequency is defined as shown in equation (4):

$$f(t) = \frac{1}{2\pi} \frac{d\Psi(t)}{dt} \quad (4)$$

Whereas the ROCOF is defined as shown in equation (5):

$$\text{ROCOF}(t) = \frac{df(t)}{dt} \quad (5)$$

Synchrophasors are always computed in relation to the system nominal frequency (f_0). If the cosine argument is represented as in equation (6):

$$\Psi(t) = \omega_0 t + \varphi(t) = 2\pi f_0 t + \varphi(t) = 2\pi \left[f_0 t + \frac{\varphi(t)}{2\pi} \right] \quad (6)$$

Therefore, the formula for the frequency becomes as shown in equation (7):

$$f(t) = f_0 + \frac{d\left[\frac{\varphi(t)}{2\pi}\right]}{dt} = f_0 + \Delta f(t) \quad (7)$$

where $\Delta f(t)$ is the deviation of the frequency from the nominal one. The ROCOF simplifies as:

$$\text{ROCOF}(t) = \frac{d^2\left[\frac{\varphi(t)}{2\pi}\right]}{dt^2} = \frac{d[\Delta f(t)]}{dt} \quad (8)$$

The frequency in phasor measurements may be reported either as the actual frequency $f(t)$ or as the deviation of the frequency from the nominal one $\Delta f(t)$, which is equal to a scalar number in steady-state conditions [9].

1.2 Evaluation of Measurements

1.2.1 TVE, FE and RFE

According to IEEE Standard C37.118.1 [9], the theoretical values of a synchrophasor representation of a sinusoidal and the values obtained from a PMU may include differences in both amplitude and phase. Theoretically they could be separately specified but, in this Standard, they are considered together in a quantity called Total Vector Error (TVE).

The TVE is an expression of the difference between a “perfect” sample of a theoretical synchrophasor and the estimate given by the unit under test at the same instant of time. The value is normalized and expressed as per unit of the theoretical phasor. TVE is defined in equation (9):

$$\text{TVE}(n) = \sqrt{\frac{[\widehat{X}_r(n) - X_r(n)]^2 + [\widehat{X}_i(n) - X_i(n)]^2}{[X_r(n)]^2 + [X_i(n)]^2}} \quad (9)$$

Where $\widehat{X}_r(n)$ and $\widehat{X}_i(n)$ are the sequences of estimates given by the unit under test of the real and imaginary parts, respectively; and $X_r(n)$, $X_i(n)$ are the sequences of the theoretical values of the input signal at the instants of time (n) assigned by the unit to those values.

These theoretical values can be determined in certain well-defined situations, such as constant frequency [9].

The frequency and ROCOF measurements shall also be evaluated. With the criteria defined in [9], frequency and ROCOF errors are the absolute value of the difference between the theoretical values and the estimated ones given in [Hz] and [Hz/s] respectively.

The frequency measurement error (FE) is defined as in equation (10):

$$\text{FE} = |f_{\text{true}} - f_{\text{measured}}| \quad (10)$$

The ROCOF measurement error (RFE) is defined as:

$$\text{RFE} = \left| \left(\frac{df}{dt} \right)_{\text{true}} - \left(\frac{df}{dt} \right)_{\text{measured}} \right| \quad (11)$$

It is important to underline that the measured and the true value are for the same instant of time, which will be given by the time tag of the estimated values [9].

Synchrophasor measurements shall be synchronized to UTC time with accuracy sufficient to meet the requirement of IEEE Standard C37.118.1 [9]. In particular, a phase error of 0.01 radian, corresponding to 0.57 degrees, in the synchrophasor measurement will cause 1% TVE which is the maximum steady-state error allowed in [9]. A 0.01 radian phase error corresponds to a time error of $\pm 26 \mu\text{s}$ for a 60 Hz system and $\pm 31 \mu\text{s}$ for a 50 Hz system [4].

Measurement compliances and verifications, according to the performance classes, are analysed in the following paragraph.

1.2.2 Measurement responses and reporting rates

In order to be compliant with [9], a PMU shall provide synchrophasor, frequency and ROCOF measurements that meet the requirements specified in this and in the following paragraph.

The synchrophasor requirements do not only involve certain values of TVE, FE and RFE as previously stated, but an important attention must be also given to the time needed to obtain the results and the required reporting rates for PMUs depending on the power system's rated frequency.

Measurement response time is the time to transition between two steady-state measurements before and after a step change is applied to the input. This shall be measured by applying a positive or negative step change in phase or magnitude to the PMU input signal, which is held at steady-state before and after the step change [9].

Measurement delay time is defined as the time interval between the instant that a step change is applied to the input of a PMU and a measurement time during which the stepped parameter achieves a value that is halfway between the initial and final steady-state values. Both the step time and the measurement time are measured on the UTC time scale.

The purpose of this evaluation is to verify that the time tagging of the synchrophasor measurement has been properly compensated for the filtering system group delay, so that the delay will be near zero [9].

Measurement reporting latency is the time delay from when an event occurs on the power system to the time that it is reported in data. This latency includes many factors, such as the window over which data is gathered to perform a measurement, the estimation method, measurement filtering, PMU processing time and so forth.

In [9], PMU reporting latency is defined as the maximum time interval between the data report time and the time when data becomes available at the PMU output.

Measurement reporting rate is a constant value, F_s , expressed as an integer number of times per second at which synchrophasors, frequency and ROCOF are provided.

PMUs shall support data reporting at sub-multiples of the nominal power system frequency. Required rate for 50 or 60 Hz systems are expressed in Table 1.

Table 1: Reporting rates. Adapted from [9]

System Frequency	50 Hz			60 Hz					
Reporting rates F_s [frames/second]	10	25	50	10	12	15	20	30	60

The actual rate to be used shall be selectable by the user, indeed higher rates at 100 or 120 Frames/s can be permissible as well as lower rates than 10 Frames/s. For slower rates verifying $F_s < 10$ Frames/s, no dynamic requirements must be verified, and no filtering is required.

1.3 Compliance Verifications

1.3.1 Steady-state requirements

The IEEE Standard C37.118.1 [9] states that compliance with the requirements shall be evaluated by class of performance, either P or M class.

As previously anticipated, P class is optimal for applications requiring fast response (e.g. for protection purposes); whereas M class is ideal for analytic measurements requiring greater precision, but not needing minimum reporting delay. It is up to the user to choose the adequate class for the needed application.

In order to perform compliance tests, a calibration device has to be used: it has to be traceable to national standards and have a test uncertainty ratio of at least 4 compared with the test requirements, e.g. it provides a TVE measurement within 0.25% where TVE is 1% [9].

All compliance tests have to be performed with all parameters set to standard reference conditions, except for those being varied as specified for the test. The reference condition specified for each test is the value of the quantity being tested when not being varied; reference conditions for all tests are as follows:

1. Voltage at nominal
2. Current at nominal
3. Frequency at nominal
4. Voltage, current, phase and frequency constant
5. Signal total harmonic distortion (THD) $< 0.2\%$ of the fundamental

6. All interfering signals < 0.2% of the fundamental

For all devices, both steady-state and dynamic compliance tests must be performed, except for measurements at reporting rates (F_s) lower than 10 Frames/s which shall not be subject to dynamic performance requirements [9].

Synchrophasor, frequency and ROCOF steady-state measurement requirements are illustrated in Table 2.

Table 2: Steady-state measurement requirements. Adapted from [9]

Influence quantity	Reference condition	Minimum range of influence quantity over which PMU shall be within given TVE			
		P Class		M Class	
		Range	Max TVE [%]	Range	Max TVE [%]
Signal frequency range - f_{dev} (Note 1)	$F_{nominal}$ (f_0)	± 2.0 Hz	1	± 2.0 Hz for $F_s < 10$ $\pm F_s/5$ for $10 \leq F_s < 25$ ± 5.0 Hz for $F_s \geq 25$	1
Voltage signal magnitude	100 % rated	80 % to 120 % rated	1	10 % to 120 % rated	1
Current signal magnitude	100 % rated	10 % to 120 % rated	1	10 % to 120 % rated	1
Phase angle with $ f_{in} - f_0 < 0.25$ Hz (Note 2)	Constant or slowly varying angle	$\pm \pi$ radians	1	$\pm \pi$ radians	1
Harmonic distortion	< 0.2 % THD	1 % each harmonic up to 50 th		10 % each harmonic up to 50 th	
		Max FE	Max RFE	Max FE	Max RFE
	$F_s > 20$	0.005 Hz	0.01 Hz/s	0.025 Hz	6 Hz/s
	$F_s \leq 20$	0.005 Hz	0.01 Hz/s	0.025 Hz	2 Hz/s
Out-of-band interference (Note 3)	< 0.2 % of input signal magnitude	No requirements for both max FE and Max RFE (Note 4)		10% of input signal magnitude for $F_s \geq 10$	Max TVE 1.3
				Max FE	Max RFE
				0.01 Hz	0.1 Hz/s

Note 1: The signal frequency range tests are performed applying a deviation frequency f_{dev} to the nominal one ($f_0 \pm f_{dev}$). These tests must be performed over the specified range and they must meet the given requirements at three temperatures: $T = 0; 23$ and 50 °C.

Note 2: The phase angle test can be performed when an input frequency offset f_{in} is present in the nominal frequency f_0 so that $|f_{in} - f_0| < 0.25$ Hz. This condition provides a slowly varying phase angle that simplifies compliance verification without causing significant other effects.

Note 3: In order to test the out-of-band interference, one has to consider the fundamental frequency of the input test signal (f_{in}) which is varied between f_0 (nominal system frequency, e.g. 50 or 60 Hz) and $\pm 10\%$ of the phasor reporting rate frequency (F_s): $f_0 - 0.1 (F_s/2) \leq f_{in} \leq f_0 + 0.1 (F_s/2)$.

This test is performed since a signal whose frequency exceed the reporting rate F_s can alias into the passband, therefore the effectiveness of the PMU anti-aliasing filter must be verified.

This test can be performed using a single frequency sinusoid added the fundamental power single at the required magnitude level and varied over a range from below the passband (10 Hz) and from above the passband up to the 2nd harmonic ($2 f_0$).

Note 4: Frequency and ROCOF are required to comply with the measurement limits only over the same range of frequencies specified for phasors, although most of these measurements will operate over a much wider range.

1.3.2 Measurement latency compliance

The latency in measurement reporting is a critical factor for measurements used in real-time applications. In addition to measurement latency, there are many factors contributing to reporting delay, such as communication coding and transmission distance. The application using the data must take into consideration all delays determining the system performance, not only computational delays. According to [9], PMU real-time output reporting latency shall be determines for each reporting rate F_s using at least 1000 consecutive messages. The reporting latency is the maximum of these values and it shall be determined to an accuracy of at least 0.0001 s, as illustrated in Table 3.

Table 3: Measurement reporting latency. Adapted from [9]

Performance class	Maximum measurement reporting latency [s]
P class	$2/F_s$
M class	$5/F_s$

It is worth emphasizing the fact that compliance tests cannot be performed only on PMUs, but also on the whole measurement chain. The contributions of transducers, as Low Power Instrument Transformers (LPITs), are not negligible but they have to be regarded as the main source of uncertainty, therefore if LPITs are not characterized, their errors cannot be compensated [3]. Several calibration procedures for the whole measurement chain under nominal and off-nominal conditions have been proposed in literature [13], [14].

1.4 Synchrophasor Networks

1.4.1 Measurement System

An example of synchrophasor network can be seen in figure 3 consisting of PMUs and PDCs.

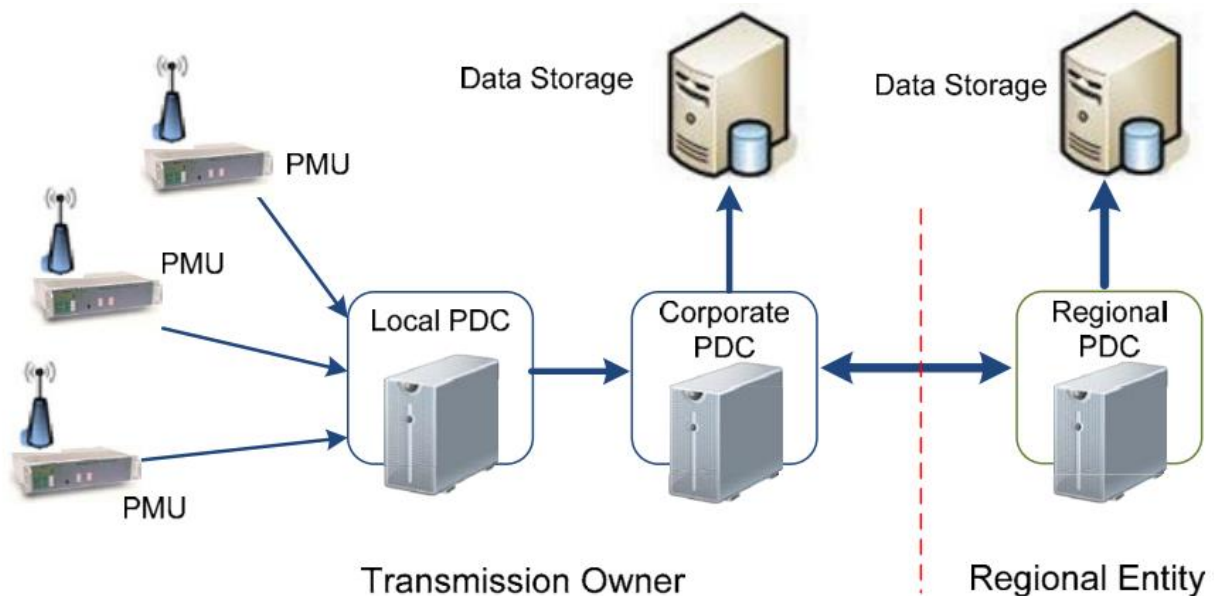


Figure 3: Synchrophasor Network. Adapted from [4]

If multiple Intelligent Electronic Devices (IEDs) in a substation provide synchrophasor measurements, a local PDC may be deployed in the substation. Its role is to collect data coming from various PMUs installed in key substations.

The aggregated data collected by PDCs may be used to support many applications, e.g. visualization of information and alarms or provide control and protection functionalities [4].

Many PDCs belonging to different utilities may be interconnected to a common central PDC to aggregate data across the utilities in order to provide an overview of the grid power measurements useful for the system operators, such as measured frequency, primary voltages, current, real and reactive power flows [4].

1.4.2 Phasor Data Concentrator

A PDC works as a node in a communication network where synchrophasors data from a number of PMUs or PDCs is correlated and fed out as single stream to the higher level PDCs and/or applications. One of the main tasks of a PDC is to correlate synchrophasor data by time-tag to create a system of wide measurement set.

Among additional features, there are:

1. Quality checks on the phasor data
2. Check for disturbance flags
3. Monitoring of the overall measurement system and displaying the results
4. Specialized outputs, directly to a SCADA or EMS system

A PMU or PDC may transmit its data in one or more separate data streams; each of them may have different contents and may be sent at a different rate, or it may be sent to different devices and locations. Therefore, each stream must be individually controllable, have its own ID code and a separation configuration control. This allows to send data to different devices with different purposes and class of service (M or P class) [4].

1.4.3 Communication System (IEC 61850)

The Standard IEC 61850 has been developed in order to standardize the communication system among substations, which previously exploited protocols and standards from other application fields [15].

To ensure interoperability, IEC 61850 serial communication protocol deals with both the standardization of data objects and the mode in which they are accessed. Among the most common services, there are GOOSE and SV. GOOSE stands for “Generic Object Oriented System Event” which is used for the rapid transmission of information that is critical in terms of time, such as changes of status, interlocks or opening commands between IEDs. Sampled Value (SV), instead, is capable of transmitting rapidly a flow of current or voltage samples [15].

One of the most critical situations could be the release of a circuit breaker. To deal with this, one would have to associate SV service to the class with the most stringent time requirements (3 ms, for instance) to avoid delays in the detection of fault conditions.

Security criteria adopted for these messages involve a continuous communication between IEDs via cyclic data transmission and multiple sending of data to ensure it has been received.

In order to properly analyse the sequence of events, they must have a time with 1 ms accuracy. This may be obtained using a Simple Network Time Protocol (SNTP) or even higher order accuracies in the order of 1 μ s can be achieved by means of optical fibre (OF).

As it can be visualized in figure 4, IEC 61850 selects the basic technologies for communication stacks: a stack structure in accordance with ISO/OSI (Open Systems Interconnection model) layers, which include Ethernet (layers 1 and 2), TCP/IP (layers 3 and 4) and the Manufacturing Messaging Specification or MMS (layers 5 to 7). The object-oriented model and relative services are mapped at MMS application layer (layer 7). Only critical services over time, such as SV and GOOSE, are mapped directly at Ethernet layer (layer 2) [15].

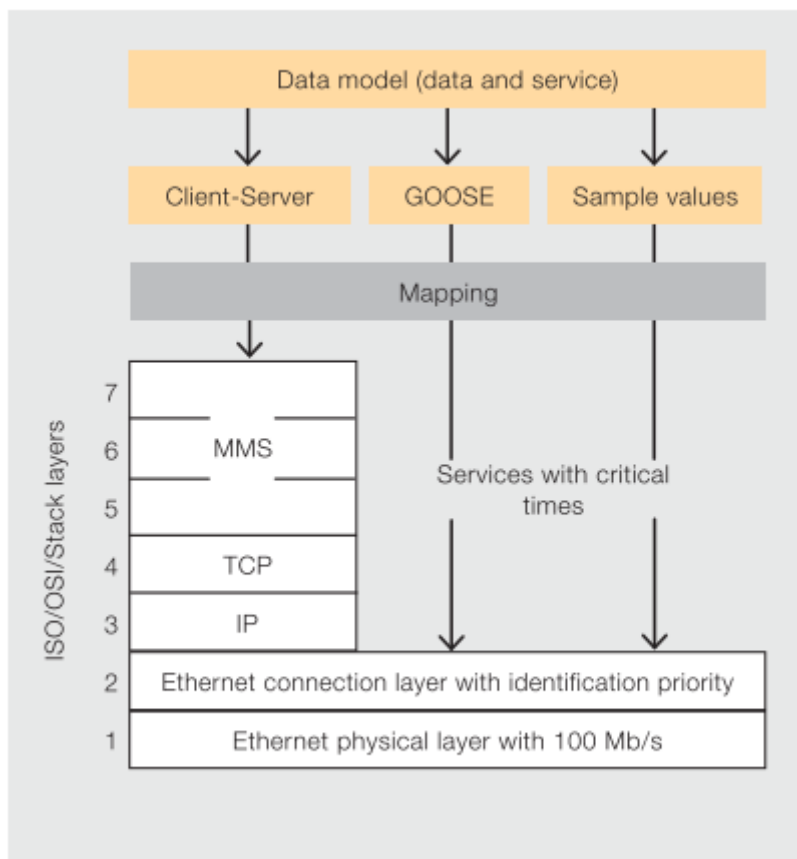


Figure 4: IEC 61850 mapping into ISO/OSI layers. Adapted from [15]

The communication structure is realized as follows and as can be seen in figure 5: the highest level is the station level in which central computers with their relative Human-Machine Interfaces (HMIs) are located.

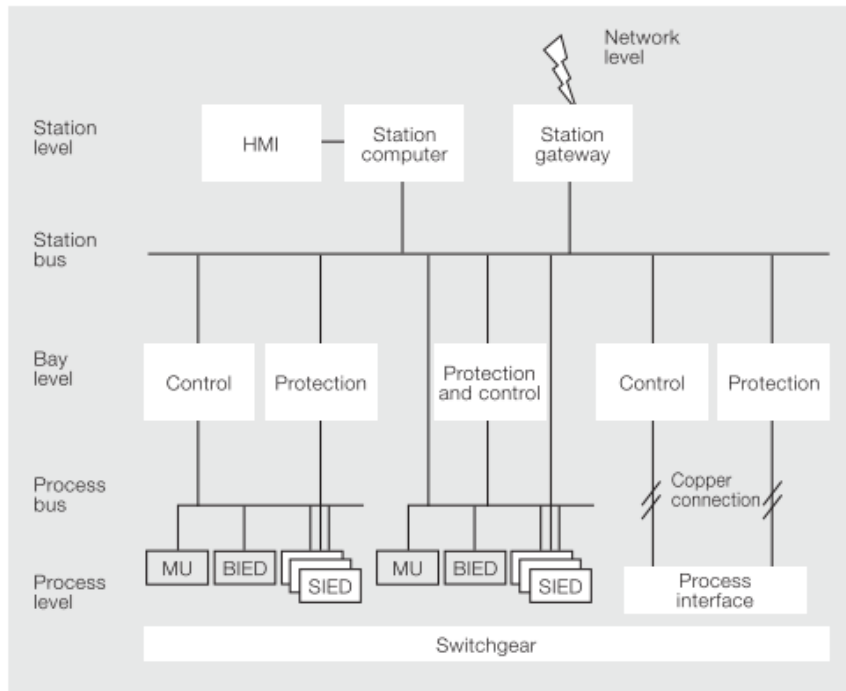


Figure 5: Station and process buses. Adapted from [15]

The station bus connects the protection, control and monitoring IEDs of the bay units to the devices at station level. The transmitted information concerns control, such as measurements and interlocks. The MMS protocol is used for transferring data between the station level and the bay IEDs, while GOOSE is the service used for transferring data from bay to bay.

The process bus connects the bay units to operating devices in the field using services such as SV for transmitting measurement samples for protection purposes.

The bottom of the structure there is the so-called process level in which IEDs of circuit breakers, disconnectors and their relative connections are realized.

In this part, units called Merging Units (MUs) are used to convert analog signals from conventional and non-conventional current or voltage instrument transformers into IEC 61850 SV data frames.

The sampling frequencies and the synchronization accuracy required are specified in the Standard.

In addition to all this, the presence of DG has made crucial the importance of cyber security since the decentralization of distribution grids together with ICT (Information and Communications Technology) has made networks more and more subject to cyber-attacks. The integration of new sensors, such as PMUs, and non-secure IP based communication protocols, like IEC 61850, has significantly increased the vulnerability of power systems. Moreover, the absence of human interaction and the presence of automated monitoring facilitate them. The IEC 61850-9-2 Standard using SV has been helpful in identifying voltage abnormalities at the low voltage level [16] and a new model proposed in [17] based on Digital Twin (DT) technology may help in the detection of

cyber-attacks, based on training exemplified for predefined contingencies, in order to quickly classify the type of grid event.

1.4.4 Cyber Security issues

Cyber security issues have become of prime importance to power system operators, ever since power industries have exploited ICT, because cyber threats, if successful, may affect the reliability of the bulk electric systems [18].

According to IEEE Standard C37.244 [6], cyber security for information technology focuses on three aspects of the electronic information communication systems:

1. *Availability*: it is typically addressed by redundancy and security measures to prevent denial of service (DoS) or Distributed DoS (DDoS) attacks meant to shut down a machine or a network, making it inaccessible to its users.
2. *Integrity and authenticity*: they respectively refer to the validity of data and its correct source. They are accomplished through means such as digital signatures or various types of authentication codes.
3. *Confidentiality*: it is achieved by preventing unintentional disclosure of information and is accomplished through the use of encryption and access control.

In figure 6, it is possible to observe that cyber-attacks may manipulate control signals in any level of the electric facility (process, bay or station) and acronyms C, I and A denote confidentiality, integrity and availability, respectively.

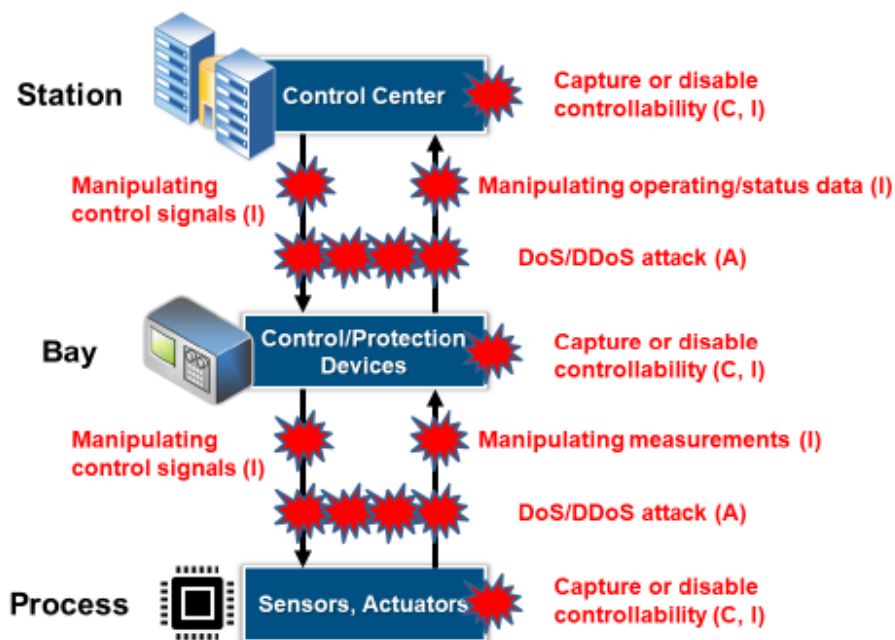


Figure 6: Cyberattack paths and their relationship with information security. Adapted from [18]

Specifically concerning PMUs, precise timestamps are required with an accuracy up to tens of nanoseconds to UTC and their working principle is strictly related to synchronization.

Therefore, any natural or malicious timing anomaly is of concern to operators; threats can include system or satellite malfunctions, Global Navigation Satellite System (GNSS) jamming and spoofing, and even malware attacks. In [19], a particular focus has been given to defence-in-depth schemes and metrics aimed at detecting GNSS attacks and errors.

The application of cyber security to PDCs aims at keeping the reliability of PDCs themselves, securing synchrophasor communications and ensuring correct performances and functionalities.

It is worth noting that the PDC itself may be needed to implement access controls, firewalls and intrusion detection functions. In [6], several standards helping cyber security and programs are proposed; some of them are specifically designed for the automation and control of substations as IEEE C37.240 [20].

Chapter 2 – Power Quality Measurements

2.1 Power Quality objectives

The term *power quality* refers to a wide variety of electromagnetic phenomena that characterize the voltage and current at a given time and at a given location on the power system. It also refers to the concept of powering and grounding electronic equipment in a manner that is suitable to the operation of that equipment and compatible with the premise wiring system and other connected equipment, according to IEEE Standard 1159-2019 [21].

Distortion in voltage and/or current represents the major difficulty in the phasor estimation: a wavelet analysis could first discriminate the discontinuities, followed by an adaptive window algorithm to estimate the phasor quantities [7].

Harmonic distortion as well as the impact of switching loads on the frequency estimation have led to a broadening of synchrophasor algorithms to provide PMUs with the capability of estimating harmonic phasors, especially in active distribution networks [7].

As far as power quality (PQ) is concerned, it must be said that all aspects regarding PQ monitoring are influenced by the objectives that the utility is seeking to address. These aspects include the monitoring technology selected, the number of monitors deployed and their location, the parameters that are measured and how often they are measured [22].

In general, the following six main objectives for power quality monitoring can be distinguished:

- *Compliance verification*: it compares a defined set of PQ parameters with limits given by standards (e.g. EN 50160, IEC 61000-2-2, IEC 61000-4-30), rules or regulatory specifications [23]–[27].
- *Performance analysis or Benchmarking*: it is usually an issue for a network operator and the results are used for internal purposes, as asset management or strategic planning [26].
- *Site characterisation*: it is used to describe PQ at a specific site in a detailed way, answering pre-connection questions of a specific customer regarding PQ.
- *Troubleshooting*: it concerns measurements based on a PQ problem, such as exceeding levels and equipment damage.
- *Advanced applications and studies*: they cover specific measurements and investigations to improve the efficiency of the system operation.
- *Active power quality management*: it includes all applications where any kind of network operation control is derived from the measurement results.

To accomplish these various tasks, different degrees of measurement accuracy are required. Utilities can leverage voltage and current transducers already present on the grid or integrate new sensors and instrument transformers offering multiple choices of accuracy, for instance:

- Power distribution transformers ($\pm 0.6\%$),
- Integral CT and VT (± 1 to 2.5% and phase angle error $\pm 1.5^\circ$),
- Instrument transformers (± 0.15 to 0.3%),
- Sensors (± 0.2 to 1%).

Especially for voltage harmonic measurements in MV, HV and Extra High Voltage (EHV) networks, conventional instrument transformers can have a significant influence on the accuracy of the measurements [26].

2.2 PMUs for Power Quality

As anticipated in the previous paragraph, the usage of PMUs in order to assess PQ in the power system is becoming more and more relevant owing to an increase in the number of installed power electronic devices, such as HV Direct Current (HVDC), Flexible AC Transmission System (FACTS), wind and solar power plants [28].

In [28], several test cases have been developed on three PMUs made by different manufacturers and assessments have been performed based on the criteria defined in the IEEE Standard C37.118.1 [9].

In this paragraph, particular emphasis is given to voltage unbalances and harmonics tests.

A *voltage unbalance* occurs when the phase voltages are not equal and/or they do not have a phase shift of 120° with respect to each other; voltage unbalances are steady-state quantities and they are defined as the maximum deviation from the average of the three-phase voltages or currents [26].

The most common reason for a voltage unbalance occurs when the load currents are in an unbalanced situation: the uneven distribution of single-phase loads can be continuously changing across the three-phase system. This problem is getting worse and worse as the presence of single-phase generators and loads will be noticeably increased in the future smart grids [26].

On the other hand, in order to assess the harmonic influence on frequency estimation of the PMUs, five different test sets consist of a combination of the fundamental harmonic component together with a single higher harmonic (ranging from 2nd to 30th) with the magnitude of 10% of the fundamental harmonic component [28].

There exist several indices to describe the harmonic phenomenon, nevertheless the most common ones are Total Harmonic Distortion (THD) indices for voltages and currents, computed as follows:

$$\text{THD}_V = \frac{\sqrt{\sum_{h=2}^{\infty} V_h^2}}{V_1} \quad (12)$$

$$\text{THD}_I = \frac{\sqrt{\sum_{h=2}^{\infty} I_h^2}}{I_1} \quad (13)$$

Results provided in [28] show that, concerning steady-state voltage unbalance tests, the TVEs for all PMUs are relatively small and below the required criteria of 1%, as it can be seen in figure 7, required in IEEE Standard C37.118.1 [9].

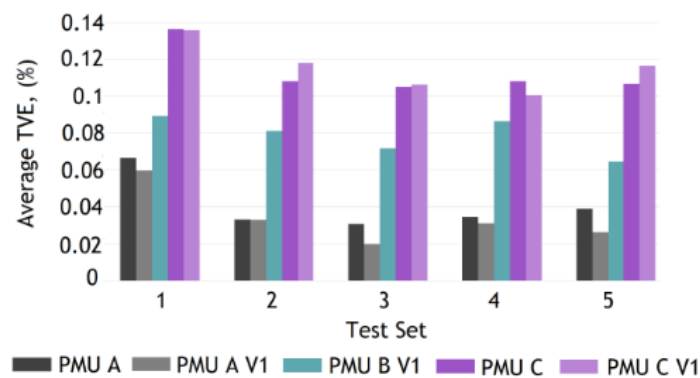


Figure 7: Average TVE for PMU phase phasors and positive sequence (V1) phasors. Adapted from [28]

Whereas, regarding harmonics, only one PMU was capable of performing such measurements. In figure 8, the test results for all the five sequences are illustrated in terms of THD and its error from the reference.

The error for the harmonic estimation is relatively small: about 0.05% or smaller for current harmonics and 0.03% or smaller for voltage ones [28].

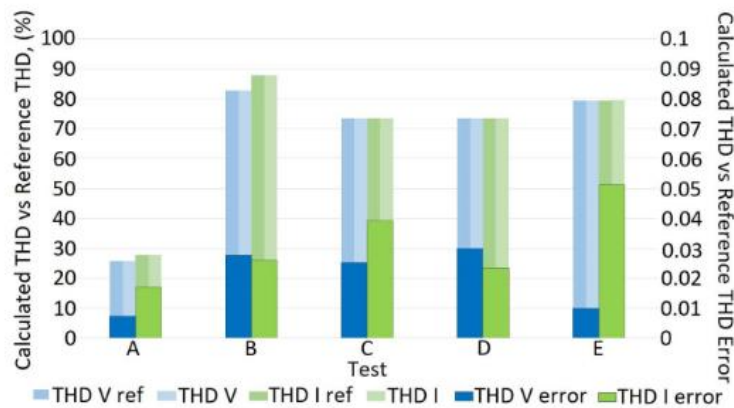


Figure 8: THD of voltage and current harmonics. THD V/I total harmonic distortion for voltage and current signals. THD V/I error is calculated as the difference of the computed and reference THD. Adapted from [28]

In the overall, it can be concluded that it is possible to use PMU data for some indicative steady-state PQ assessment, such as frequency, voltage testing and voltage unbalance estimation. Nonetheless, most of the PMUs are not capable of performing harmonic measurements as special modules and estimation algorithms are required [28].

Chapter 3 – Real-Time Simulations

3.1 Real-Time Implementations

3.1.1 What are Real-Time simulations

Most of Real-Time (RT) simulations are referred to be as *discrete-time simulations*. During discrete-time simulations (or “fixed-time simulations”), the time moves forward in steps of equal duration [29].

In such simulations, the amount of real time required to compute all equations and functions during a given time-step may be shorter or longer than the duration of the simulation time-step itself. Therefore, three different cases may occur and are illustrated in figure 9 [29].

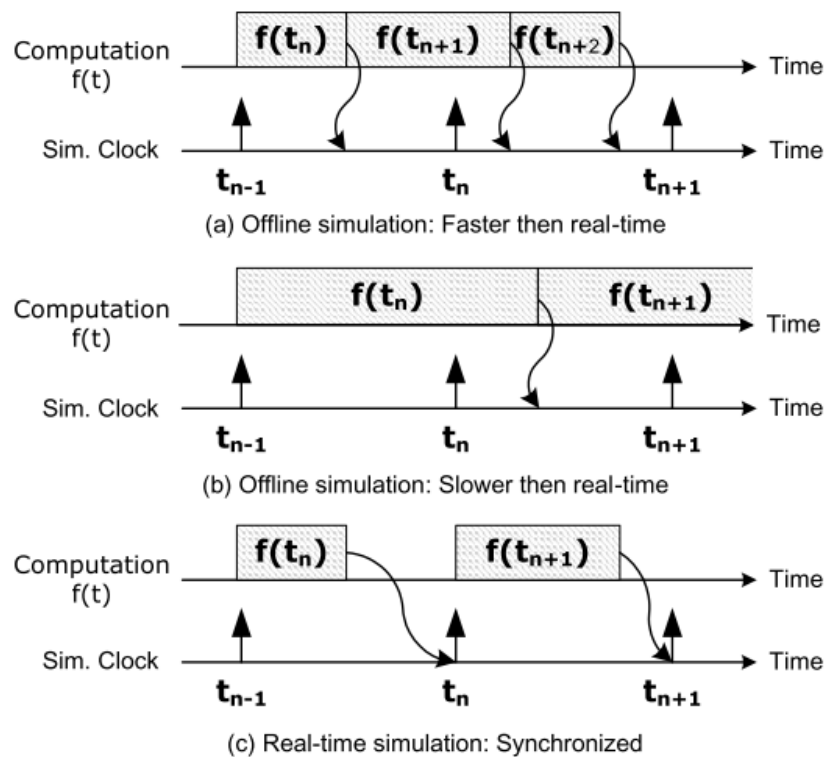


Figure 9: Real-Time simulation techniques. Adapted from [29]

The case a) of figure 9 indicates a case in which the computing time is shorter than a fixed time-step; it is also referred to as “faster than real-time”.

Whereas the case b) of the same figure shows that the computing time takes longer than the time-steps; it is referred to as “slower than real-time” simulation.

Both cases are called “offline simulations” and they are characterized by the fact that the instant at which one receives the result is not relevant. What is important is the obtaining of the results as fast

as possible; this speed is strictly related to the computational power and the system's mathematical model complexity [29].

On the other hand, during a RT simulation, the accuracy of the computations depends both on the precise dynamic representation of a system and on the length of time used to produce results. As it can be seen in case c) of figure 9, the RT simulator must be able to produce the outputs of the simulation within the same length of time that its physical counterpart would.

Indeed, the computational time has to be shorter than the duration of the time-step as, during the latter, the RT simulator must drive inputs and outputs (I/O) to and from the externally connected devices. If this synchronization constraint is not met, the simulation is considered erroneous and it is commonly known as *overrun* [29].

As it can deduced, proper time-step duration must be determined in such a way that the system frequency response is accurately represented, up to the fastest transient of interest.

For each of these time-steps, the simulator must execute the same list of tasks:

1. Read the inputs and generates outputs. As it can be understood, the states of any externally connected device are sampled just once at the beginning of each time step.
2. Solve the model equations
3. Exchange results with other simulation nodes
4. Wait for the start of the next step. Conversely with respect to offline simulations, in real-time ones, any idle time preceding the next time step is lost.

Regarding time-steps, one of the first challenges concerns the choice of correct simulation speed depending on the type of application needed. In figure 10, the typical time-steps and computing power requirements for a variety of applications is illustrated.

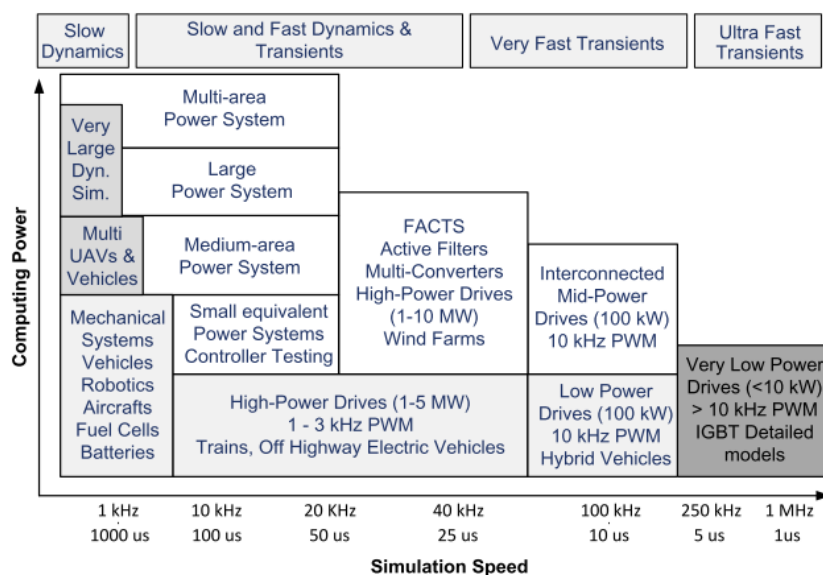


Figure 10: Simulation time-step by application. Adapted from [29]

Typical Electromagnetic Transient (EMT) simulations require time-steps of few tens of microseconds to provide acceptable results for transients up to 2 kHz.

In figure 11, specific requirements for EMT simulations are shown.

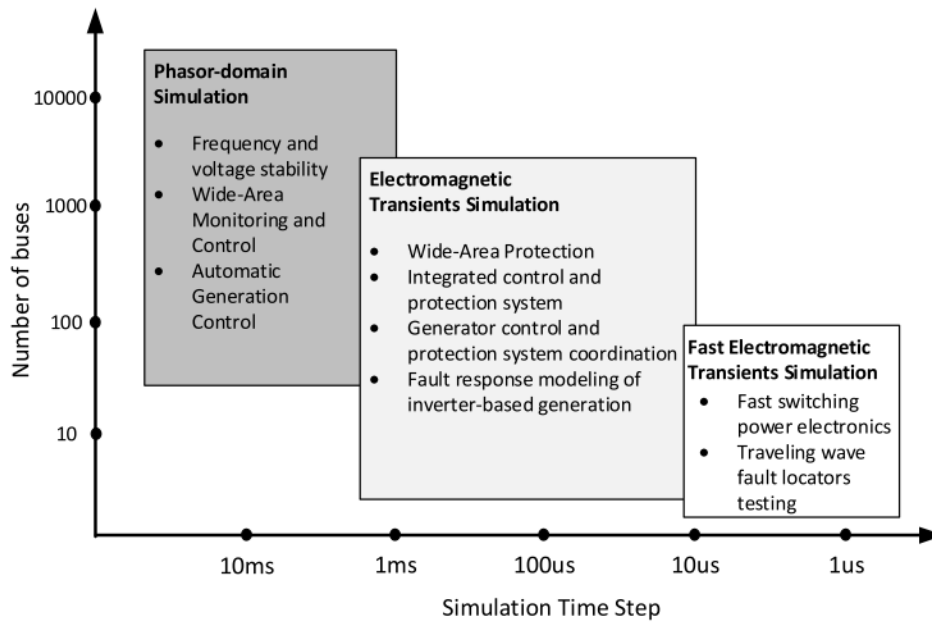


Figure 11: Suggested EMT simulation time-steps. Adapted from [30]

According to [29], as a rule of thumb, a simulation step should be smaller than 5 to 10 % of the smallest time constant of the system. Nevertheless, both numerical solver performances and the bandwidth of interest must be considered when selecting the right time step.

Despite the fact that discrete time is a necessary requirement for today's simulators, it can also be a major limitation for the simulation of non-linear systems, such as HVDC, FACTS, active filters or drivers owing to the numerical instabilities caused by the system's non-linearities [29].

One of the most common effects of a non-linearities in a simulation is the *jitter*. It occurs when an event does not occur synchronously to the simulation fixed-time-step and a jitter can be visualized as sub-synchronous or uncharacteristic harmonics with amplitude variations appear visible in the resulting waveforms [29].

There exist a number of proposed solutions to contain this problem:

1. Discrete-time compensation techniques involving interpolation algorithms
2. Usage of advanced I/O cards running a sampling rates much faster than the fixed-time simulation [31], [32].
3. Multi-rate simulations in which different parts of a model are simulated at different rates. In these simulations, though, there exists issues concerning the results accuracy and simulation stability [33].

3.1.2 Application categories

As depicted in figure 12, there are three main different kinds of simulations to be performed with a RT simulator.

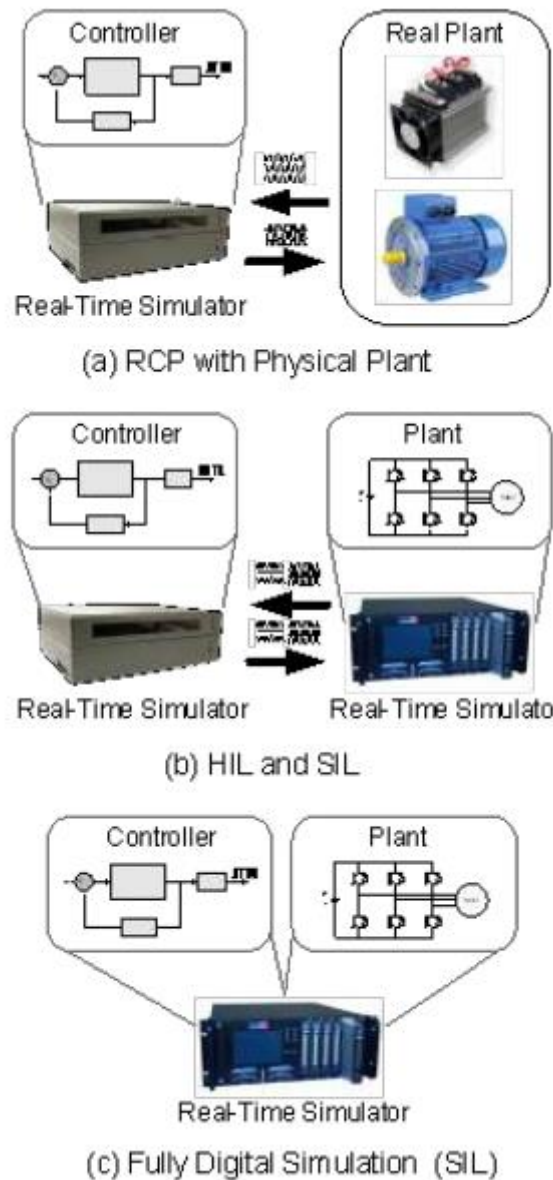


Figure 12: Application categories. Adapted from [29]

1. Rapid Control Prototyping (RCP): in this simulation, a plant controller is implemented using a RT simulator and is connected to a physical plant. A controller prototype developed in such a way is more flexible, faster to implement and easier to debug [29].
2. Hardware-In-the-Loop (HIL): conversely to RCP, in this case a physical controller is connected to a virtual plant executed on a RT simulator, instead of a physical plant. In addition to RCP advantages, HIL allows early testing of controllers when physical test

benches are not available and it can allow testing conditions unavailable on real hardware, such as extreme event testing [29].

3. Software-In-the-Loop (SIL): it is a combination of RCP and HIL as both the controller and the plant are simulated in RT in the same device. SIL shows many advantages, among which no I/O are used therefore the system integrity is preserved.

Timing is no longer critical due to the fact that controller and plant are in the same device: this allows one to run the simulations either slower (if the simulator lacks computing power) or faster than real-time (a large number of tests can be performed in a short period) [29].

4. Power Hardware-In-the-Loop (PHIL): it is an extension of HIL, as it involves the creation of a virtual power interface between the digital simulation and the devices under test (DUTs).

The power interface typically involves power amplifiers (voltage and/or current) which be selected carefully depending on the application to act as a source or sink.

Among different DUTs, one can implement protection devices, protective relays, power converters (inverters, rectifiers and power supplies), electric machines, batteries and their relative Battery Management Systems (BMS) [34].

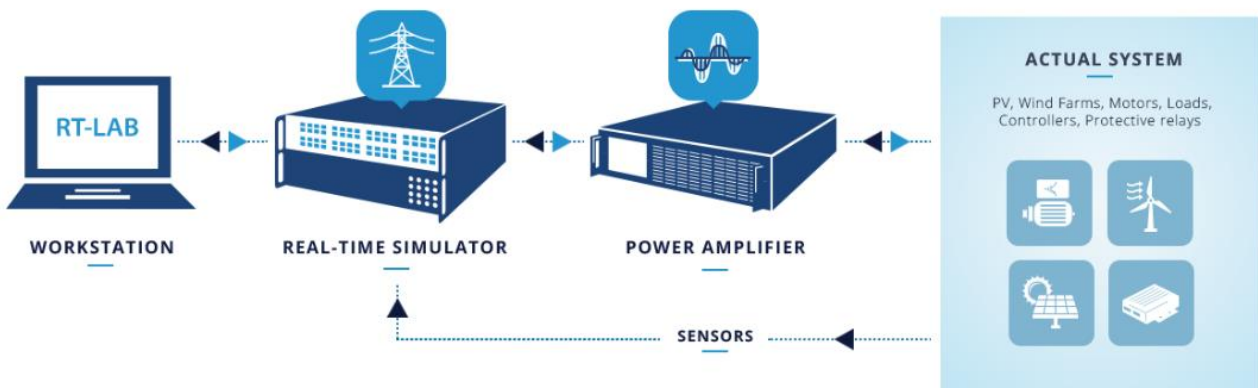


Figure 13: PHIL application. Adapted from [34]

The last trend in RT simulations consist of exporting simulation models to Field Programmable Gate Arrays (FPGAs) since this method shows many advantages: the computation time is almost independent of the system size owing to the parallel nature of FPGAs; overruns cannot occur once the model is running and timing constraints are met and, finally, the simulation steps can be very small in the order of 250 nanoseconds. The only limitations are related to the limited number of gates in FPGAs [29], [35].

3.1.3 Importance of RT tests and validation

Two of the main reasons for the importance of RT simulations are stated below:

1. Model-based Design (MBD), or “V diagram”: it is a mathematical and graphical method of addressing problems associated with the design of complex systems. It is shown in figure 14 and four basic steps can be identified:

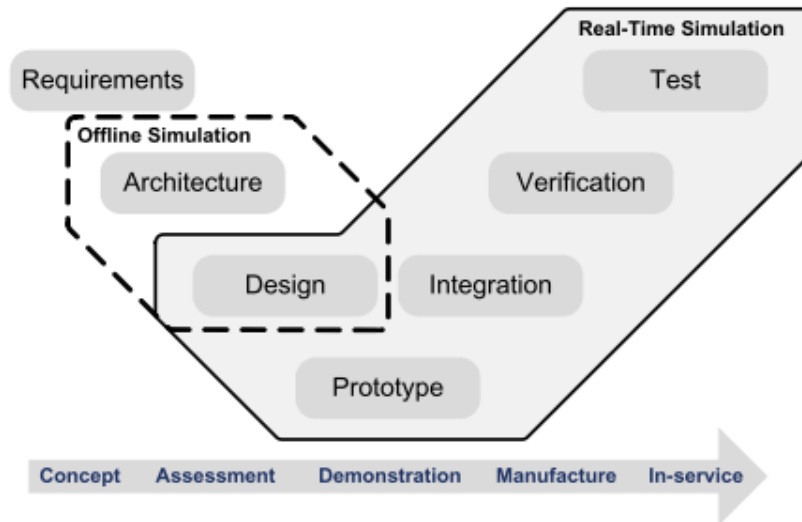


Figure 14: MBD workflow. Adapted from [29]

- a. Build the plant model
- b. Analyse the plant model and synthesize a controller for it
- c. Simulate the plant and the controller together
- d. Deploy the controller

Among the many advantages of MBD, one can identify the possibility of having a common design environment available to every person involved in the project [30].

Moreover, combining RCP and HIL while using MBD approach, has many benefits:

- a. Design issues can be discovered earlier in the process
 - b. The parallelization of the workflow reduces the development cycle duration
 - c. The usage of HIL test setups for multiple applications and projects reduces testing costs
 - d. Repeatability of testing results since RT simulator dynamics do not change over time.
2. Interaction with the model: when a user or a physical equipment interacts with a RT model, they can provide model inputs and get model outputs, as it would with a real plant.

A model executed on a RT simulator can also be modified online and any parameter can be read and/or updated continuously.

Moreover, any model quantity is accessible during execution and it is possible to perform extra ordinary assessment, for instance if a controller can compensate for changes in plant dynamics caused by the aging of components [30].

Having highlighted the importance of RT simulations, it is worth noting that testing and validation of complex systems has become an important part of the design process.

The most critical criterion in conducting a RT digital simulation is how to obtain acceptable model accuracy with an achievable simulation time-step. For such reason, in order to build trust in a simulation tool, a large number of validation tests must be performed using many different applications, configurations, time-steps and I/O cards [30].

3.2 HIL Synchrophasor testing

3.2.1 PMU testing architectures

Regardless of the typology of testing architecture, the main benefits of using a Real-Time Simulator (RTS) in PMU studies are stated as follows:

1. Flexibility to model various power system configurations using generic or specific models of power system equipment.
2. Various dynamic operating conditions and testing scenarios can be planned and repeated over time as testing conditions only depend on the model.
3. External hardware can be connected to the RTS via its analog and digital channels.

RTS also supports a variety of communication protocols as IEEE C37.118.2 and they can simulate network connections such as Controller Area Network (CAN) bus, General Purpose Interface Bus (GPIB) and Ethernet [4], [29], [30].

Four different testing architectures may be identified and, for all of them, for general protection and control system testing, 50 μ s to 100 μ s is considered a sufficient time-step.

1. PMU test: in this test, illustrated in figure 15, the voltage and current measurements from the simulated model in the RTS are sent out to the PMU under test as time-domain sinusoidal waveforms.

The PMU under test calculates the corresponding phasor and frequency data, and reports them back to the RTS using IEEE C37.118.2 [4] to compare them with the reference ones.

Additional PDC can be used for the collection of data if many PMU under test are present [30].

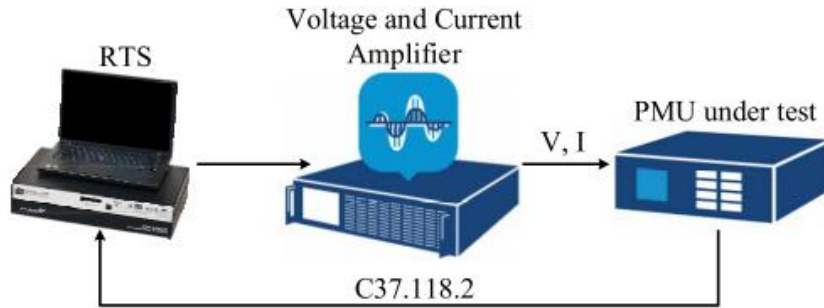


Figure 15: PMU unit tests. Adapted from [30]

- Virtual and under test PMUs: in this case shown in figure 16, a virtual PMU is implemented in the RTS to calculate phasors as a real PMU with P or M class. The data coming from the PMU under test can be sent as C37.118.2 data streams with a reporting rate up to 240 frames/second [4], [30].

This test setup allows one to validate PMUs and WAMPAC schemes.

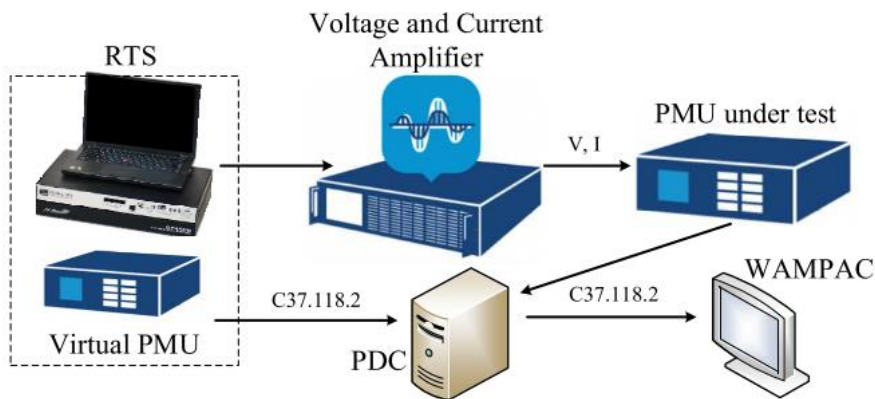


Figure 16: Virtual and PMU under test. Adapted from [30]

- EMT in power systems: voltage and frequency stability, state estimation and system model validation can be modelled using OPAL tools such as ePHASORSim allowing to simulate large networks [30], [36].

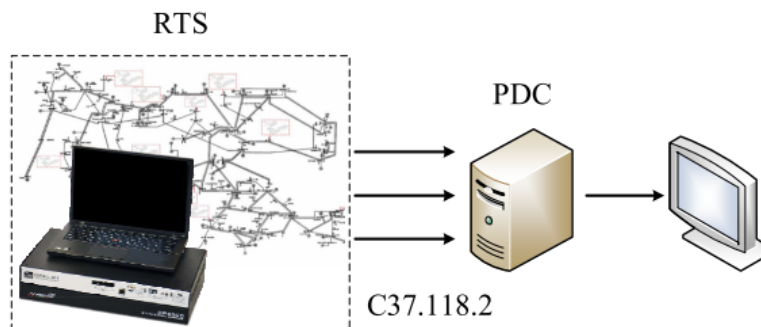


Figure 17: EMT simulations in RT. Adapted from [30]

4. RCP of PMUs: these tests are necessary from the validation of control algorithms, receiving data from either a PDC or multiple real/simulated PMUs.

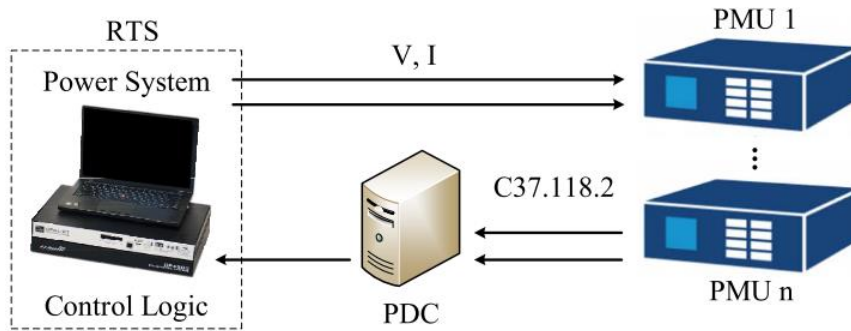


Figure 18: RCP of PMUs. Adapted from [30]

3.2.2 PMU compliance procedures

Measurement accuracy of PMUs must be ensured via calibration procedures, as erroneous PMU signals may trigger false alarms in monitoring applications.

It is worth saying that a RTS test suite has been developed in HYPERSIM [37] and it allows one to perform compliance tests on PMUs defined in IEEE Standard C37.118.1 providing pass/fail test results [9], [30].

In [30], tests have been performed on five M-class PMUs implementing different algorithms. Some of the results are shown in figures 19 and 20.

In the first one, the maximum TVE of each PMU under test is shown for steady-state and frequency ramp tests. As noticeable, all PMUs are compliant with the Standards, as the TVE is below 1% [9].

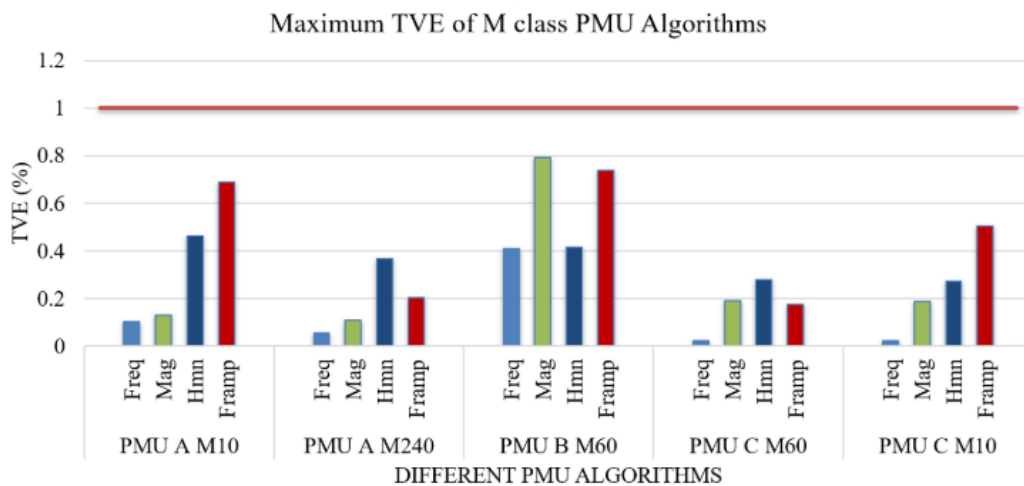


Figure 19: Max TVE results of M-class PMUs. Adapted from [30]

As it can be highlighted, a higher reporting rate results in an improvement in the TVE for a frequency-ramp test, but in the other type of tests the TVE does not change much [30].

Instead, in figure 20, TVE responses are shown during a phase angle step test. It can be observed that a higher reporting rate results in a shorter TVE response time than that of a lower reporting rate as it can be perceived when the reporting rate of PMU A is changed from 10 to 240 frames/second.

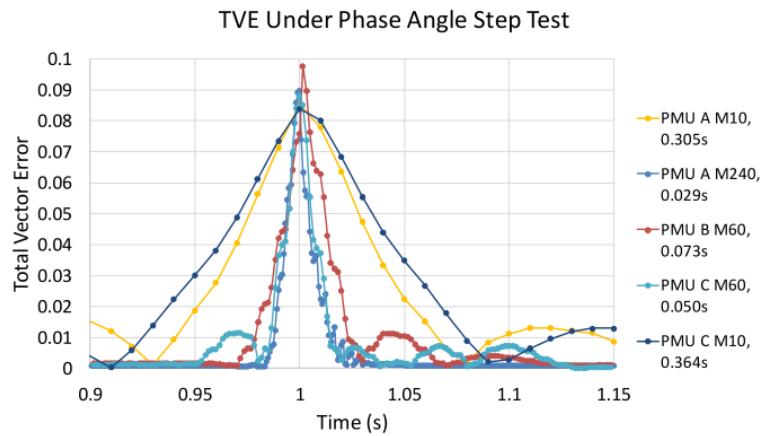


Figure 20: TVE in phase angle step results. Adapted from [30]

The previous tests performed in [30] have been conducted in such a way that the analog signals coming out of the RTS were directly sent to the real PMU under test. In this case, no additional amplifiers have been introduced.

Nevertheless, if one wants to test the PMU under rated voltage and current conditions, then amplifiers in the loop must be introduced. It is important to consider that the input/output delay and the bandwidth of the amplifier need to be assessed, since a narrow bandwidth or a long step response may negatively affect the PMU results [30].

Hence, it is crucial to evaluate the performance of the amplifier and perform its calibration. The amplifier uncertainty is tested and compensated in such a way that the simulation tests obtained with or without the amplifier in the loop are similar, as shown in figure 21.

In [30], the harmonic distortion test results are compared in presence and absence of the amplifier in the loop. Again, the TVE is well below the 1% threshold.

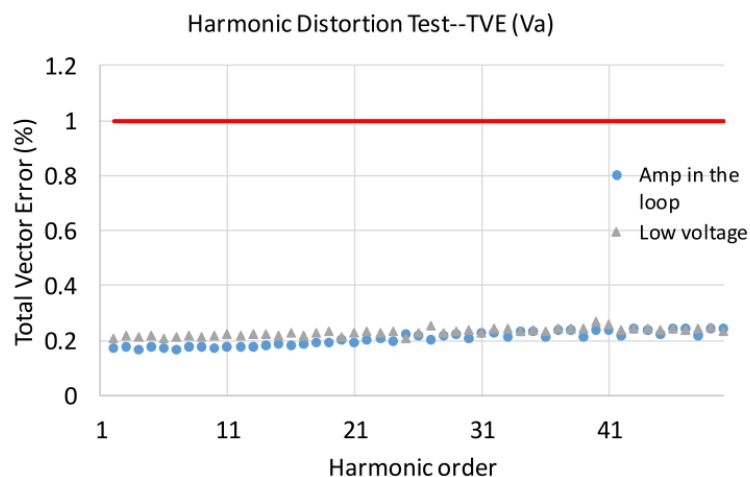


Figure 21: Harmonic distortion tests with (blue) and without (grey) amplifier in the loop. Adapted from [30]

3.2.3 Other HIL tests for PMUs

The deployment of PMUs in a HIL test setup allows one to perform other tests which could not be done otherwise.

For instance, one can assess the communication latency of the wide-area network (WAN) by connecting PMUs devices with a PDC and configuring the network. RTS is able to emulate a network and this test can be relevant for the evaluation of latency issues affecting results [30]. The latter can be eventually compensated by communication latency modules.

Finally, one can also evaluate the impact of Time Synchronization Spoofing Attacks (TSSA) on synchrophasor-based WAMPAC applications.

These tests are important to be performed since PMUs working principle is intrinsically based on the reception of time-synchronization signals which can be done via protocols such as Pulse Per Second (PPS), PTP or via the most common way, the Inter-Range Instrumentation Group-code B (IRIG-B) format. IRIG timecodes are Standard formats for transferring timing information, specifically group B is widely used by electric utilities to ensure precise time synchronization of power system devices like relays and meters.

In [30], [38], tests have been performed in case of a PMU being under TSSA event. The block diagram is illustrated in figure 22.

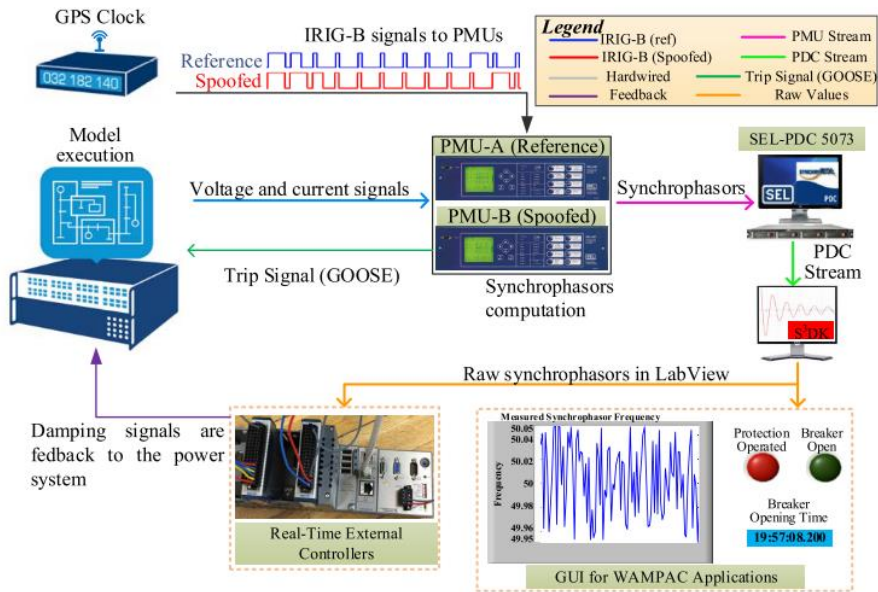


Figure 22: TSSA block diagram test setup. Adapted from [31]

It is critical to prevent TSSA attacks since time synchronization has a strong impact on PMU phase angle computation and many WAMPAC schemes are sensitive to small phase angle error; these tests justify the need for high measurement accuracy [31].

Chapter 4 – Practical Implementation

4.1 Topology observability rules

The majority of Optimal PMU Placement (OPP) algorithms have considered the network base case observability as the optimization constraint; therefore an analysis on observability rules is provided as follows [7], [39].

A system is considered observable if the bus voltage phasors (magnitude and angles) are measured directly or by calculation using circuit rules. Two different kinds of observabilities exist: numerical and topological [39].

Numerical observability is characterized by a high computational burden, whereas topological observability can be accomplished by following these common rules:

1. A bus is considered directly observable if it has a PMU installed that directly measures the voltage and/or current phasors of that bus. They are said to be *directly measurable*.
2. If the voltage and current phasors of one end of a line are known, then using Ohm's law it is possible to calculate the voltage phasor of the other end of the line. The voltage phasor is said to be *indirectly observable*.
3. If the voltage phasors of both ends of a line are directly or indirectly known, the current phasor of the line can be computed knowing the line impedance. The current phasor is *indirectly measurable*.

There exist other constraints regarding the presence of Zero-Injection Buses (ZIB). These buses are characterized by the absence of a generator and/or loads; hence the sum of all the currents flowing to a ZIB is zero [39].

ZIBs have the capability of reducing the required number of PMUs to be installed for the complete observability of the network. Nevertheless, the network which will be considered for the practical tests does not show any ZIBs; hence the study of OPP algorithms in case of ZIBs is out of the scope of this thesis.

Consequently, assuming that a PMU is installed at bus k , then both the voltage phasor at bus k and the current phasors along all lines/branches incident to bus k are quantities assumed to be available [2].

Provided that the network model and its parameters are available, then this information will allow the computation of the phasor voltages at all neighbouring buses as well.

Hence, placing a PMU at a given bus implies the observability of all branches incident to that bus as well [2].

4.2 Optimization of PMU placement and algorithms

The optimization of the number and location of PMUs in future active distribution networks is a crucial task [7], but there does not exist a unique OPP algorithm to be used in order to achieve an optimal placement of PMUs since each algorithm has its own advantages and disadvantages.

In addition to this, technical constraints have been included to the OPP problem for specific functionalities, such as fault location observability, bad data detection in State Estimation (SE), parameter error identification, protection against data injection attacks and so forth [7].

OPP algorithms can be classified in two main techniques: deterministic and heuristic [39].

1. *Deterministic* techniques are based on the mathematical representation of the model and the most widely used technique is integer linear programming (ILP) which is capable of solving large-scale power systems [39].
2. *Heuristic* solutions, instead, are able of solving OPP problems without a mathematical representation of the model. Despite a number of techniques have been proposed in literature, the capability of finding the global optimum is not always ensured [3], [39].

In general, the optimization problem can be formulated as stated in equation (14):

$$\begin{cases} \min \sum_{i=1}^N a_i x_i \\ f(\mathbf{X}) \geq 1 \end{cases} \quad (14)$$

Where:

- N represents the total number of bus candidates for the placement of the PMU
- $\mathbf{X} = \{x_1, x_2, \dots, x_N\}$ is a binary variable vector having:

$$x_i = \begin{cases} 1 & \text{if a PMU is placed at node } i \\ 0 & \text{if no PMU is placed at node } i \end{cases}$$

- a_i is weight associated to the i^{th} node (it could be associated to a different cost for the installation of a PMU at different nodes)
- $f(\mathbf{X})$ is a vector depending on the PMU set and representing the observability of each node with:

$$f_i = \begin{cases} \neq 0 & \text{if node } i \text{ is observable} \\ = 0 & \text{if node } i \text{ is not observable} \end{cases}$$

This vector can be expressed through a set of linear equations obtained by multiplying the connectivity matrix A to the vector \mathbf{X} ($A \cdot \mathbf{X}$). The matrix A is associated to the topology of the network [3], as:

$$A_{ij} = \begin{cases} 1 & \text{if } i = j \\ 1 & \text{if } i \text{ and } j \text{ nodes are connected} \\ 0 & \text{otherwise} \end{cases}$$

4.3 Implementation of the case-study

4.3.1 The IEEE 5-bus system and load flow solution

The network which will be used as a case-study for the HIL implementation is the IEEE 5-bus system $V_r = 10 \text{ kV}$ and $f_n = 50 \text{ Hz}$, illustrated in figure 23.

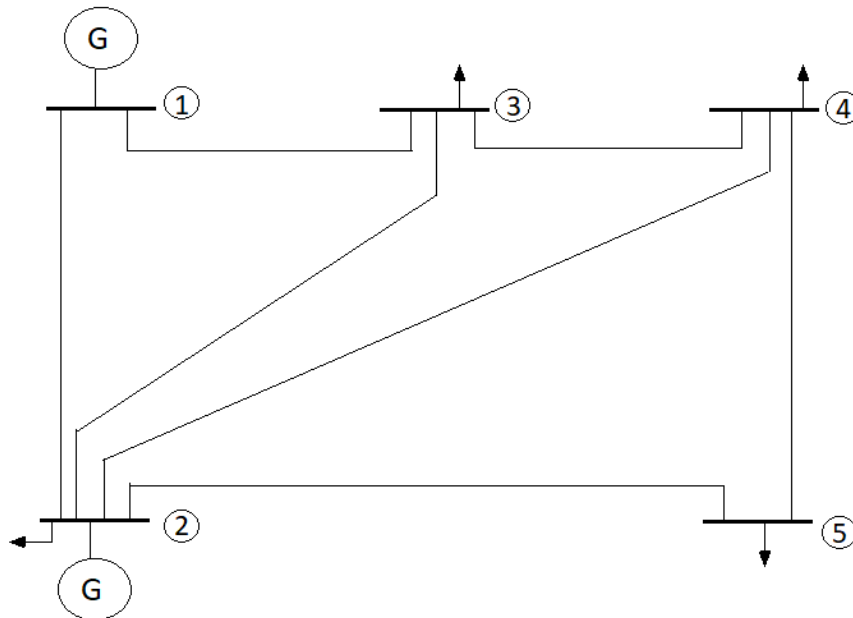


Figure 23: IEEE 5-bus network

The implementation in Simulink is represented in figure 24:

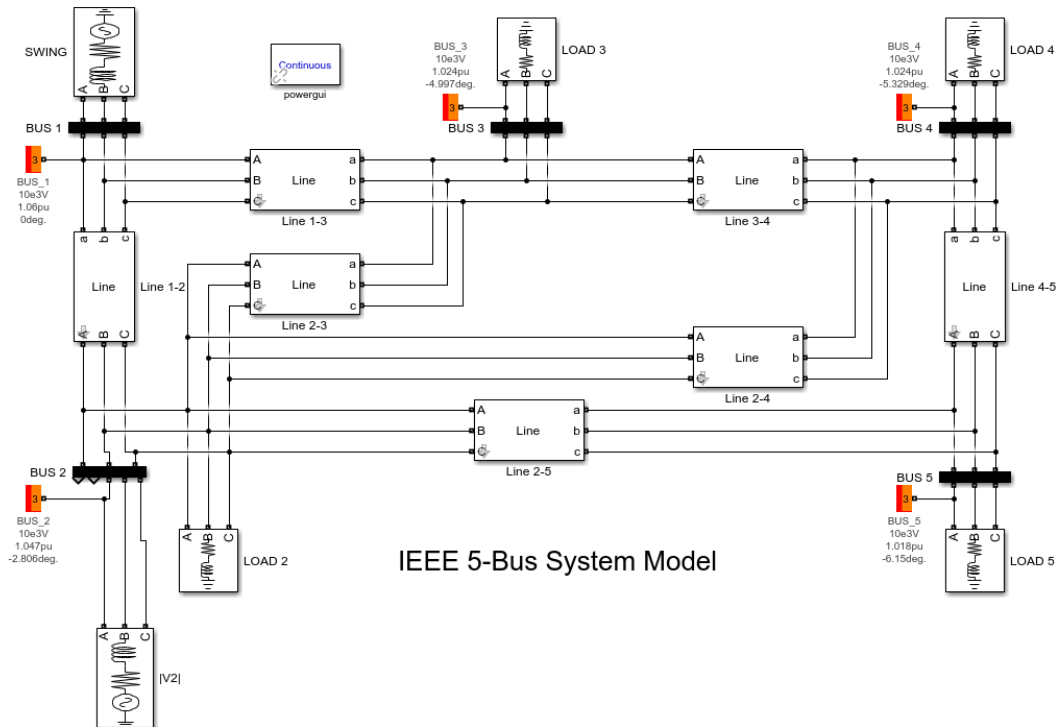


Figure 24: Simulink implementation. Adapted from [40]

The bus data for the considered bus system is illustrated in table 4:

Table 4: Bus data for IEEE 5-bus. Adapted from [40]

Bus no.	Bus Type	Bus Voltage	Generation		Load	
			MW	MVAr	MW	MVAr
1	Slack	1.06 + j 0.0	0	0	0	0
2	PV	1.00 + j 0.0	40	30	20	10
3	PQ	1.00 + j 0.0	0	0	45	15
4	PQ	1.00 + j 0.0	0	0	40	5
5	PQ	1.00 + j 0.0	0	0	60	10

Whereas the line data for the same network is summarized in table 5:

Table 5: Line data for IEEE 5-bus. Adapted from [40]

Line	Line Impedance		Line Charging
	R [p.u.]	X [p.u.]	
1-2	0.02	0.06	0.0 + j 0.03
1-3	0.08	0.24	0.0 + j 0.025
2-3	0.06	0.25	0.0 + j 0.02
2-4	0.06	0.18	0.0 + j 0.02
2-5	0.04	0.12	0.0 + j 0.015
3-4	0.01	0.03	0.0 + j 0.01
4-5	0.08	0.24	0.0 + j 0.025

The load flow solution for the implemented network yields the results illustrated in figure 25:

	Block type	Bus type	Bus ID	Vbase (kV)	Vref (pu)	Vangle (deg)	P (MW)	Q (Mvar)	Qmin (Mvar)	Qmax (Mvar)	V_LF (pu)	Vangle_LF (deg)	P_LF (MW)	Q_LF (Mvar)	Block Name
1	Vsrc	swing	BUS_1	10.00	1.0600	0.00	0.01	0.00	0.00	0.00	1.0600	0.00	129.59	-7.42	SWING
2	RLC load	PQ	BUS_2	10.00	1	0.00	20.00	10.00	-Inf	Inf	1.0474	-2.81	20.00	10.00	LOAD 2
3	Vsrc	PQ	BUS_2	10.00	1	0.00	40.00	30.00	-40.00	50.00	1.0474	-2.81	40.00	30.00	V2
4	RLC load	PQ	BUS_5	10.00	1	0.00	60.00	10.00	-Inf	Inf	1.0179	-6.15	60.00	10.00	LOAD 5
5	RLC load	PQ	BUS_3	10.00	1	0.00	45.00	15.00	-Inf	Inf	1.0242	-5.00	45.00	15.00	LOAD 3
6	RLC load	PQ	BUS_4	10.00	1	0.00	40.00	5.00	-Inf	Inf	1.0236	-5.33	40.00	5.00	LOAD 4

Figure 25: Results of the load flow calculation

For the sake of clarity, the same results are summarized in table 6:

Table 6: Load flow solution computed in Simulink

Bus no.	Bus Type	Bus Voltage		Generation		Load Demand	
		Voltage [p.u.]	Angle	MW	MVAr	MW	MVAr
1	Slack	1.06	0	129.59	-7.42	0	0
2	PV	1.0474	-2.806	40	30	20	10
3	PQ	1.0242	-4.997	0	0	45	15
4	PQ	1.0236	-5.329	0	0	40	5
5	PQ	1.0179	-6.150	0	0	60	10

4.3.2 Optimization of PMU placement

Based on the criteria illustrated in paragraph 4.1, it is clear that the IEEE 5-bus network considered for this case-study, as depicted in figure 23, requires the placement of one PMU only at bus 2.

This is due to the fact that bus 2 is linked to all the other buses of the system, therefore, placing one PMU there guarantees the observability of all the buses incident to it.

Nonetheless for the sake of completeness, the application of the basic optimization problem illustrated in paragraph 4.2 and in [3] is performed.

First of all, the network topology can be sketched as in figure 26:

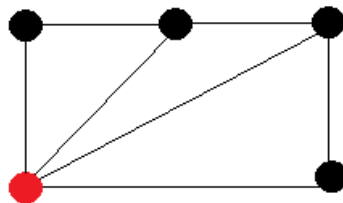


Figure 26: Network Topology (nodes in black, node with PMU installed in red)

The corresponding connectivity matrix A is filled in according to previously stated rules:

$$A = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 \end{bmatrix} \quad (15)$$

The resulting constraints to be verified are summarized in equation 16:

$$\begin{cases} f_1 = x_1 + x_2 + x_3 \geq 1 \\ f_2 = x_1 + x_2 + x_3 + x_4 + x_5 \geq 1 \\ f_3 = x_1 + x_2 + x_3 + x_4 \geq 1 \\ f_4 = x_2 + x_3 + x_4 + x_5 \geq 1 \\ f_5 = x_2 + x_4 + x_5 \geq 1 \end{cases} \quad (16)$$

Each constraint implies that at least one of the binary variables included in the equation has to be nonzero [3], indeed it is straightforward to point out that the full observability of the network can be achieved if x_2 is equal to 1, as it would satisfy all five inequalities.

Placing a PMU at bus 2 yields the following solution vector \mathbf{X} :

$$\mathbf{X} = [0|1|0|0|0] \quad (17)$$

In which in (17), 1 represents the presence of a PMU at the corresponding bus, 0 otherwise.

Finally, the vector function $f(\mathbf{X})$ resulting for the placement is obtained by multiplying \mathbf{X} to the connectivity matrix \mathbf{A} :

$$f(\mathbf{X}) = [1|1|1|1|1] \quad (18)$$

In the vector function $f(\mathbf{X})$, 1 represents the observability of the bus corresponding the index of the vector, 0 represent the non-observability of the bus and any number bigger than 1 would represent the possibility of observing that bus by multiple PMUs.

Indeed, it could be proven that adding more PMUs, in addition to that already placed at bus 2, would increase the observability of other buses, as well. On the contrary, removing the PMU installed at bus 2, but adding others in different buses would not guarantee the full observability of the network.

Further algorithms have been proposed in literature [41], [42] and it is understandable that, generally, the OPP problem might have multiple optimal solutions, among which, the one ensuring maximum measurement redundancy is the most desirable one, as it would lead to a more precise SE and higher robustness against element outages [7].

4.4 Simulink code and practical tests

4.4.1 Implementation of a PMU in Simulink

In the previous 4.3.2 paragraph, it has been observed that, in order to achieve the full observability of the IEEE 5-bus network, it is necessary to install a PMU on bus 2 notwithstanding this is not ensuring measurement redundancy.

The updated Simulink code is depicted in figure 27.

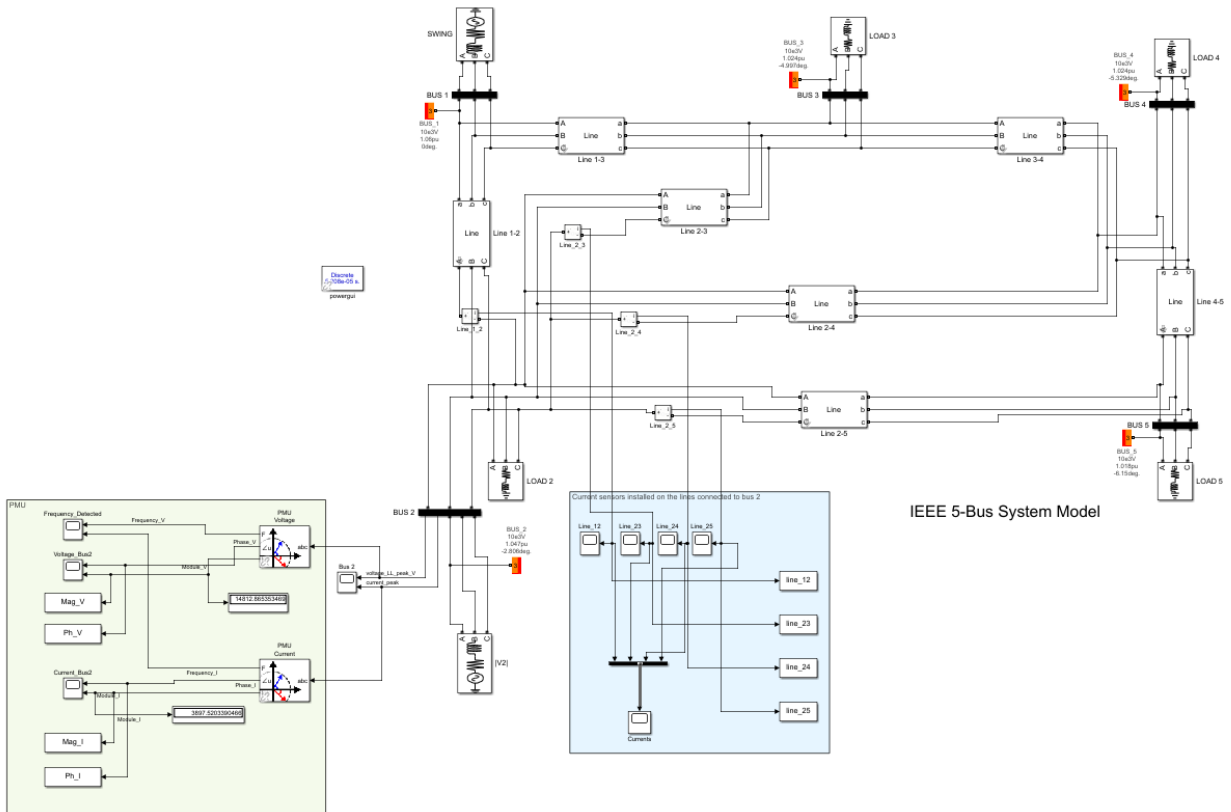


Figure 27: IEEE 5-bus with a PMU installed on bus 2

In order to better understand the meaning of the two highlighted areas, the following figures 28 and 29 respectively, allow one to take a closer look at the code.

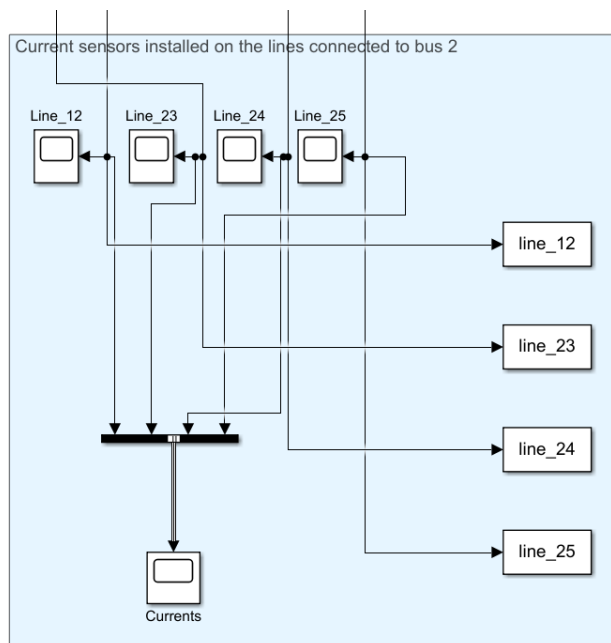


Figure 28: Scopes for the current sensors installed on the lines connected to bus 2

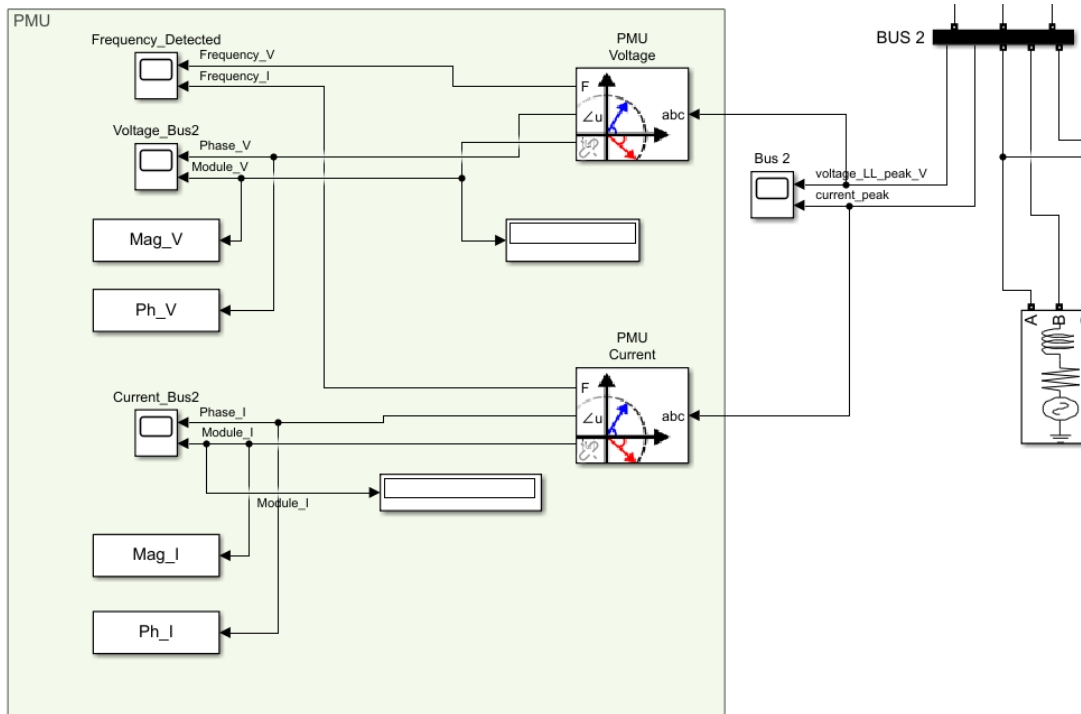


Figure 29: PMU installed at bus 2

As one could observe in figure 27, current sensors have been installed on all lines incident to bus 2, therefore, as depicted in figure 28, the data collected from these sensors is visualized by means of scopes and sent to Matlab Workspace for a further elaboration. Indeed, knowing the network parameters and using Ohm's law, it is possible to know the voltage and current phasors at the other end of a line, achieving the so-called "indirect observability" [39].

This is representing a real PMU which would have its inputs connected to the incident lines of a bus to measure the currents.

In figure 29, the remaining part of the measurement system is illustrated. Bus 2 block provides an ideal measurement of both the line-to-line voltage and the current in peak values, which are visible in the "Bus 2" scope.

These measured values, which would be sensed by the in-field PMU, are sent to two PMU blocks in the highlighted light green area, respectively for the voltage and the current. This has been done in order to separate the two different measurands, but in practice only one PMU would be needed for this measurement.

Both PMU blocks provide in output the measured frequency of the synchrophasor, as well as its module in peak value and its phase.

Displays and scopes allow one to take a preliminary check between the steady-state measurements and the PMU ones. Furthermore, this measured data is sent to Matlab for an additional analysis of the results over the whole simulation period. This part will be dealt with in paragraph 4.5.4.

The steady-state measurements in peak values, computed using powergui block, are illustrated in figure 30. These values are considered to be the ideal ones, when performing the comparison with respect to the PMU outputs.

It is worth underlying that this data is automatically rounded up by the powergui block; thus, in order to properly compute the accuracy indices, these values are processed in Matlab by means of the script in figure 31, so that they can be elaborated without rounding.

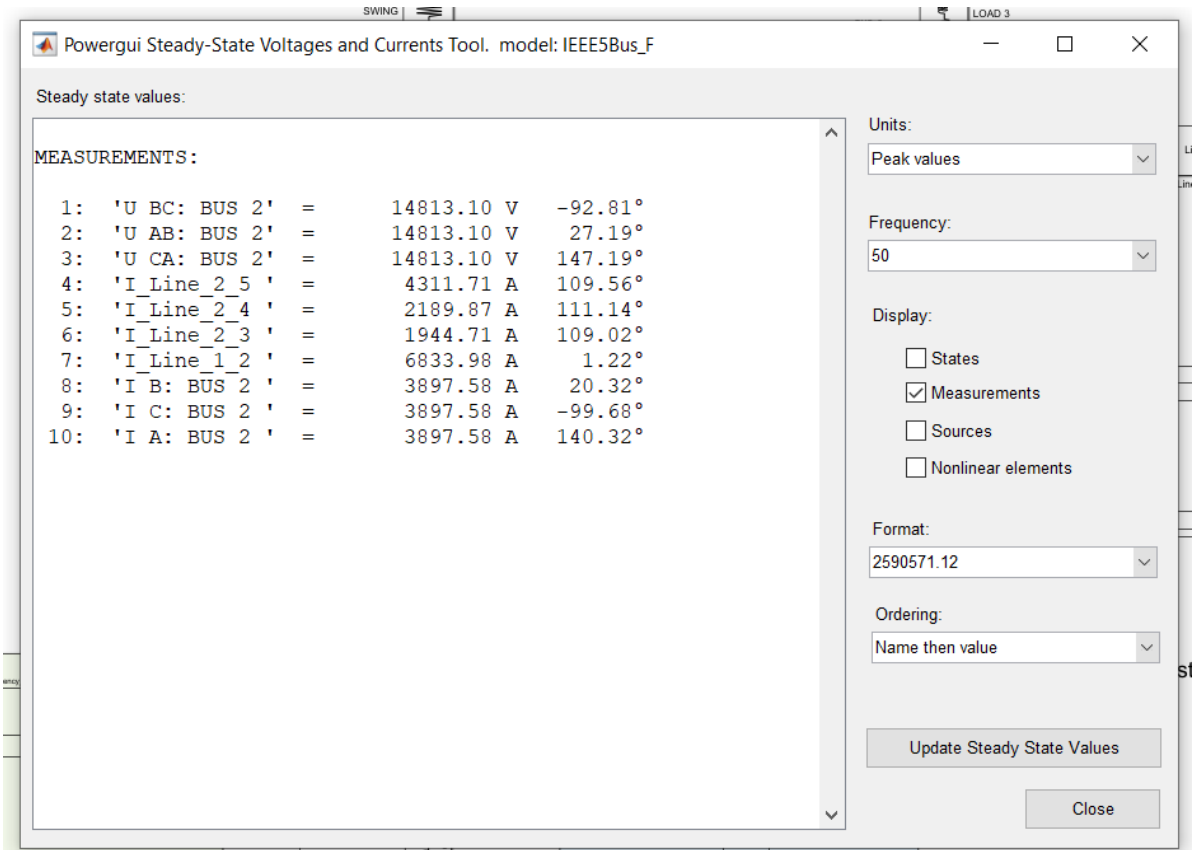


Figure 30: Steady-state measurements in the network

```

Final_Elab.m x  Dati.m x  Accuracy.m x  +
1  %%% ELAB. ACCURACY INDICES %%%
2
3  %Importing steady-state theoretical measurements: other voltages and currents are not needed in the TVE calculation
4  sps = power_steadystate('IEEE5Bus_F');
5  V_AB_th = sps.Yss_Measurements(1,1);
6  I_A_th = sps.Yss_Measurements(8,1);
7
8  %Conversion into magnitude/phase and real/imag format for the theoretical voltage
9  V_AB_magn_th = abs(V_AB_th);
10 V_AB_phase_th_rad = angle(V_AB_th); %phase in radians
11 V_AB_phase_th_deg = 180*V_AB_phase_th_rad/pi; %phase in degrees
12 V_AB_real_th = real(V_AB_th); %real part of theoretical phasor
13 V_AB_imag_th = imag(V_AB_th); %imag part of theoretical phasor
14
15 %Conversion into magnitude/phase format for the theoretical current
16 I_A_magn_th = abs(I_A_th);
17 I_A_phase_th_rad = angle(I_A_th); %phase in radians
18 I_A_phase_th_deg = 180*I_A_phase_th_rad/pi; %phase in degrees
19 I_real_th = real(I_A_th); %real part of theoretical phasor
20 I_imag_th = imag(I_A_th); %imag part of theoretical phasor

```

Figure 31: Import of steady-state data and conversions

The three-phase PLL block is needed to track the frequency and phase of the input signal by means of a controlled oscillator and the objective of the control system is to keep the phase difference at zero.

This is necessary in order to ensure that a window of one cycle of the fundamental frequency (50 Hz, in this case) is obtained. During this window, the PLL-Driven block computes the positive-sequence components in magnitude and phase of the sinusoidal input.

The reference frame needed for the computation (ωt) is given by the angle expressed in radians and varying between 0 and 2π , and synchronized on the zero-crossing of the fundamental of phase A.

In this schematic, the time synchronization from the common time source of the GPS is implicit in the model [43].

The PMU provides in output the magnitude, in the same unit as the three-phase input, the phase and the frequency of the measured synchrophasor.

The PMU mask presents itself as in figure 34.

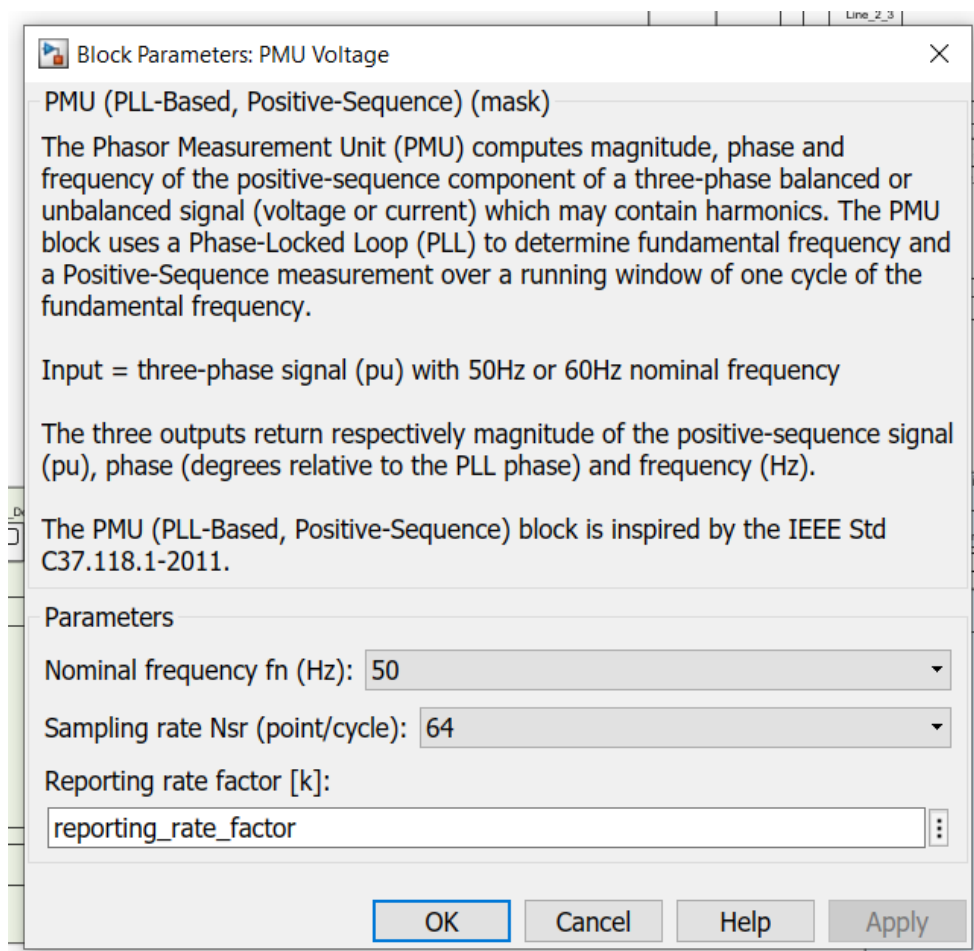


Figure 34: PMU Voltage mask

The nominal frequency has been chosen equal to the power system frequency, hence 50 Hz.

The sampling rate N_{sr} , measured in [points/cycle], has been picked equal to 64. This parameter represents the number of times for which data is reported in output of the PMU, with respect to a period of the fundamental frequency considered as a reference and equal to 20 ms, in this case.

Consequently:

$$\frac{64 \text{ points}}{20 \text{ ms}} = 3200 \frac{\text{points}}{\text{second}} = 3200 \text{ Hz} \quad (19)$$

In this case, the PMU would have a reporting rate of 3.2 kHz. According to [9] and to Table 1 illustrated in paragraph 1.1.2, the suggested reporting rates for a PMU to be compliant with the Standard are stated.

For a power system whose nominal frequency is 50 Hz, reporting rates of 10, 25 or 50 frames per second are allowed, as long as they are multiple of the fundamental frequency. Lower reporting rates than 10 are acceptable as well, as higher rates of 100 and 125 frames per second [9].

It is experienced that if a PMU is tested at the highest reporting rate and it results to be compliant with the Standard, then it is compliant also at lower reporting rates. Nonetheless, tests must be performed at all reporting rates [44].

Therefore, it is clear that the reporting rate of 3.2 kHz has to be reduced to a proper value.

To do so, the reporting rate factor k present in the PMU mask properly serves to this purpose. It is a dimensionless coefficient which reduces the number of output points of the PMU, as k itself increases.

A more detailed analysis focused on the choice of these parameters is carried out in paragraph 4.4.3 where tables summarizing simulation and PMU parameters are reported.

Based on the chosen parameters in the mask, it is now possible to compute the sample time as in equation (20):

$$T_s = \frac{1}{f_n N_{sr}} = \frac{1}{50 \text{ Hz} \cdot 64 \text{ points/cycle}} = \frac{1}{3200} = 3.125 \cdot 10^{-4} \text{ s} \quad (20)$$

T_s represents the sample time selectable inside the PMU. In practice, it would represent the sampling rate of the ADC of the PMU.

In the Simulink code, it is depicted as follows:

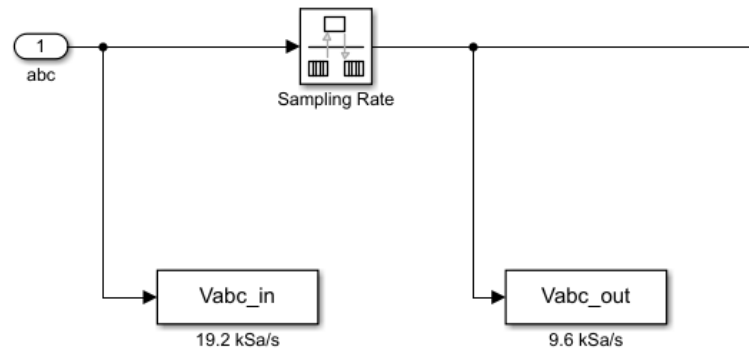


Figure 35: Sampling time of the PMU in Simulink

The Sampling Rate block accepts in input the number of samples coherent with the simulation time step.

In this study, a sampling time of $5.208 \cdot 10^{-5}$ s has been selected: it corresponds to a sampling frequency of 19.2 kSa/s which is equal to 384 points/period in case of a 20 ms sinusoid.

Inside the Sampling Rate block, it is possible to uncheck “deterministic sampling” which would allow one to choose whatever sampling rate. Otherwise, only sampling rates which are submultiples of the sample time T_s are allowed.

In this study, it has been chosen to select a submultiple of T_s , specifically a $k = 3$, as in equation (21):

$$T_s = \frac{1}{k f_n N_{sr}} = \frac{1}{3 \cdot 50 \text{ Hz} \cdot 64 \text{ points/cycle}} = \frac{1}{9600} = 1.0417 \cdot 10^{-4} \text{ s} \quad (21)$$

This choice has been driven by the fact that commercial PMUs have typical sampling rates bigger or equal than 10 kHz, therefore a sampling frequency of 9.6 kHz would be reasonable.

Moving on to the next part of the PMU block, as depicted in figure 36, the PMU PLL-based is implemented.

The content of the block is depicted in figure 38. In this section, also a conversion of the measured entities into real and imaginary parts is performed.

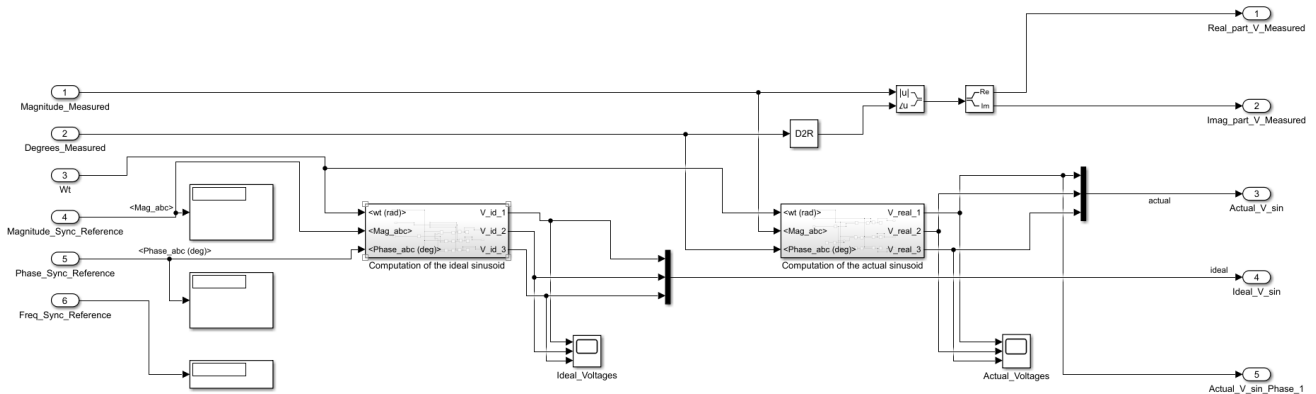


Figure 38: Sinusoid comparison code and data conversion

Moving back to figure 36, one could note that a conversion from radians to degrees is performed before reporting the measured angle in output.

In that block, the following scheme illustrated in figure 39 is implemented.

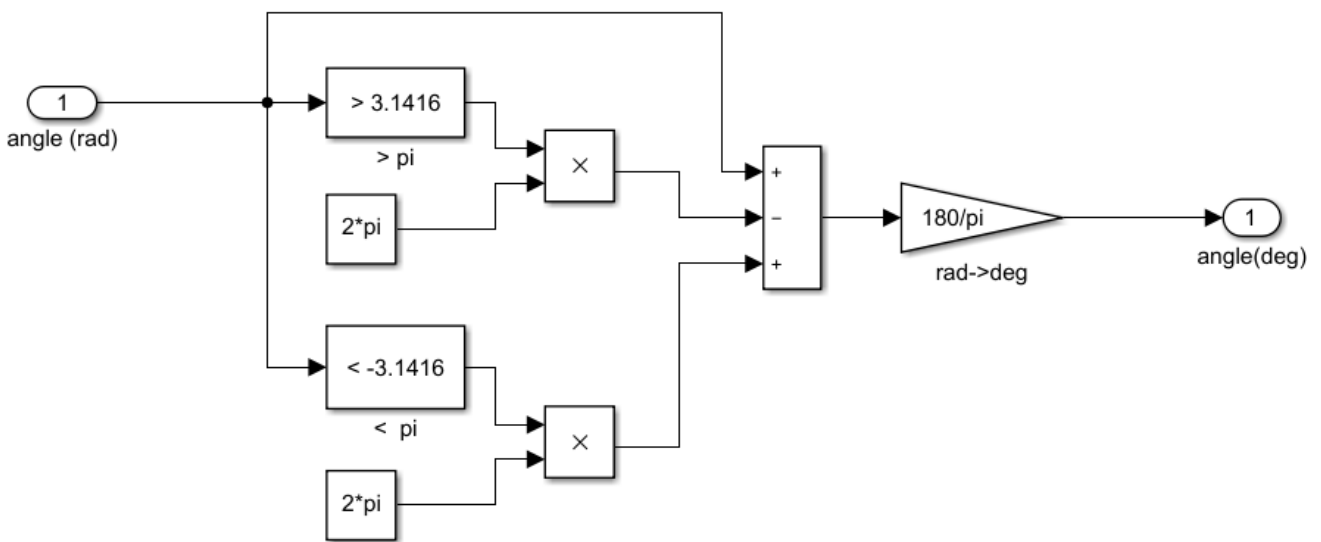


Figure 39: Radians to degrees conversion

The angle is not directly converted in degrees, as it is firstly compared to π and 2π . This is necessary in order to avoid spikes in proximity to 0° and 180° in the sensed angle, occurring every 20 ms and lasting as long as the reporting rate of the PMU, 3.2 kHz in this case.

The sensed angle ranges between 0° and 180° so that the information regarding a leading or lagging sinusoid, with respect to the ideal one, is also provided.

Subsequently, the Reporting Rate generator, represented in figure 40, is necessary to adapt the number of samples from the sampling rate (3.2 kHz) to the chosen reporting rate, according to the parameter k defined in the PMU mask.

These values are the only ones which are reported and accessible in output of the sensor. At the same reporting rate, also accuracy indices as TVE, FE, ROCOF and RFE will be reported.

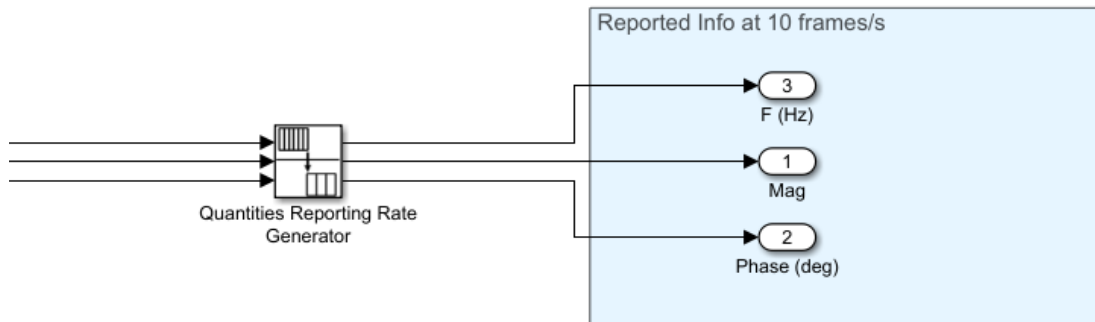


Figure 40: Reported synchrophasors information

Finally, in this PMU block the computation of the accuracy indices is also implemented.

In the realized code, FE, Relative Amplitude Error (RAE), Phase angle Error (PhE) as well as TVE and the approximate TVE formula are implemented.

All these indices are reported at the same rate as the synchrophasor data, they are visualized by means of scopes and sent to Matlab workspace for a further elaboration over the simulation time in order to compute their average values.

The content of the Indices Calculation block is analysed in paragraph 4.4.4 together with the realized Matlab code.

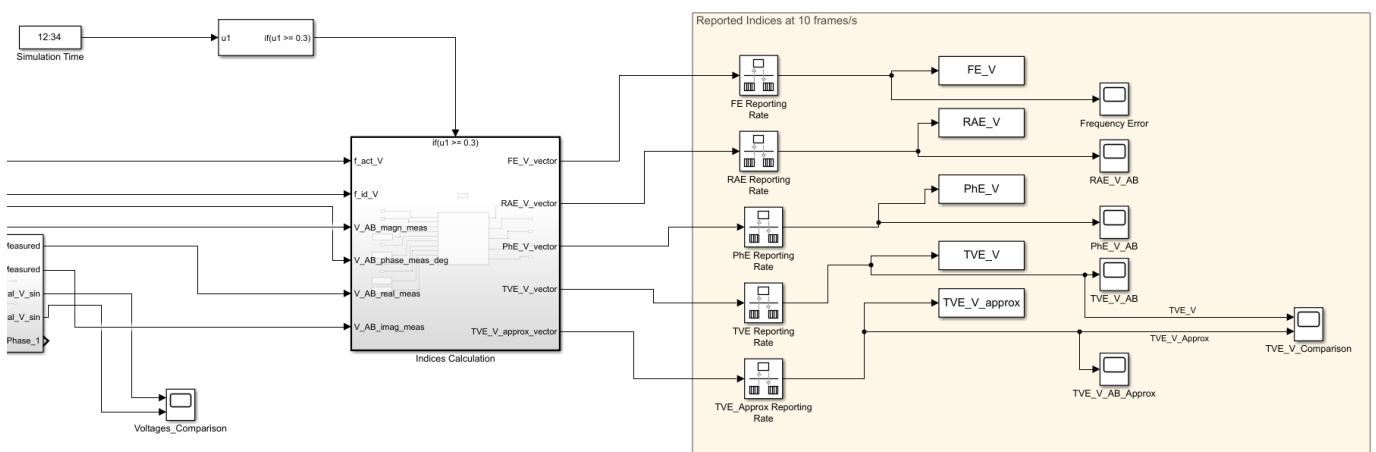


Figure 41: Indices calculation and reporting rate generators

4.4.3 Preliminary test results

Before proceeding to the assessment of the PMU in the network, a preliminary Matlab script has been realized in order to set the parameters and the sampling time of the sensors, as shown in figure 42.

```
1  %% PMU PARAMETERS %%
2  nominal_frequency = 50; %[Hz]
3  nominal_period = 1/nominal_frequency; %[s]
4  reporting_rate = 64; %Sampling_Rate Nsr in the sub mask
5  f_rep_rate = reporting_rate/nominal_period; %[Hz]
6  requested_frame_second = 10; %[frames/s] requested. For a 50 Hz system -> 10, 25, 50
7  reporting_rate_factor = f_rep_rate/requested_frame_second; %[] -> it reduced the # of reported points in output to get only the required # of Frames/s
8
9  %PMU sampling time
10 Sampling_Time = 1/(nominal_frequency*reporting_rate); %[s]
11 k = 3; %[] choosing a submultiple of Ts for ADC of the PMU
12 Sa_t = 1/(k*nominal_frequency*reporting_rate); %[s] -> actual sampling time of the ADC
13 f_sampling = 1/Sa_t; %[Hz] -> actual sampling frequency of the PMU
```

Figure 42: Matlab script for PMU parameters

In order to choose a proper reporting rate, the following considerations have been carried out.

The algorithm inside the PMU PLL-based block extracts the measurements on a window which is usually set to 1 second. When the PPS signal is received, the samples have to be acquired over a window which needs to be shorter than 1 second so that some time is still available for the measurement computation and the restarting of the acquisition, before the arrival of the next PPS signal [45].

A reporting rate of 10 frames/s corresponds to $1/10 = 0.10$ s. Having selected a sampling frequency for the ADC of 9.6 kHz, this means that during an interval of 0.10 s, 960 samples are acquired.

The remaining 0.90 s before the next PPS can be used for the computation of the indices and the restarting of the sampling.

During these 0.10 s, considering a perfectly sinusoid signal with a period of 20 ms, 5 full sinusoids are acquired, meaning that 192 samples/sinusoid are obtained.

In accordance with [9], for reporting rates smaller or equal than 10 frames/s no dynamic compliance tests are required to be performed for the assessing of the performances of the PMU.

Lastly, the duration of the simulation has been selected according to [9] suggesting tests to be no shorter than 5 seconds for steady-state compliance assessments. For this case study, the simulations are therefore set for 5 seconds.

For the sake of clarity, in table 7 and 8 all the parameters used in the simulation are summarized.

Table 7: Selected sampling times

	Sampling Frequency [Hz]	Sampling Time [s]
Simulink Simulation	19200	$5.20833 \cdot 10^{-5}$
Sampling rate of the PMU	9600	$1.04167 \cdot 10^{-4}$
Reporting rate of the PMU	10	0.10

Table 8: Selected simulation parameters

	Value
Nominal frequency of the network (f_n)	50 Hz
Reporting Rate (N_{sr})	64 points/20 ms
Actual Reporting Rate	10 frames/s
Reporting Rate Factor (k)	320
PMU type	M

Having initialized the PMU data and the steady-state information, after running the simulation one could check the voltages and currents measured at bus 2 by means of a scope. They are represented in figure 43.

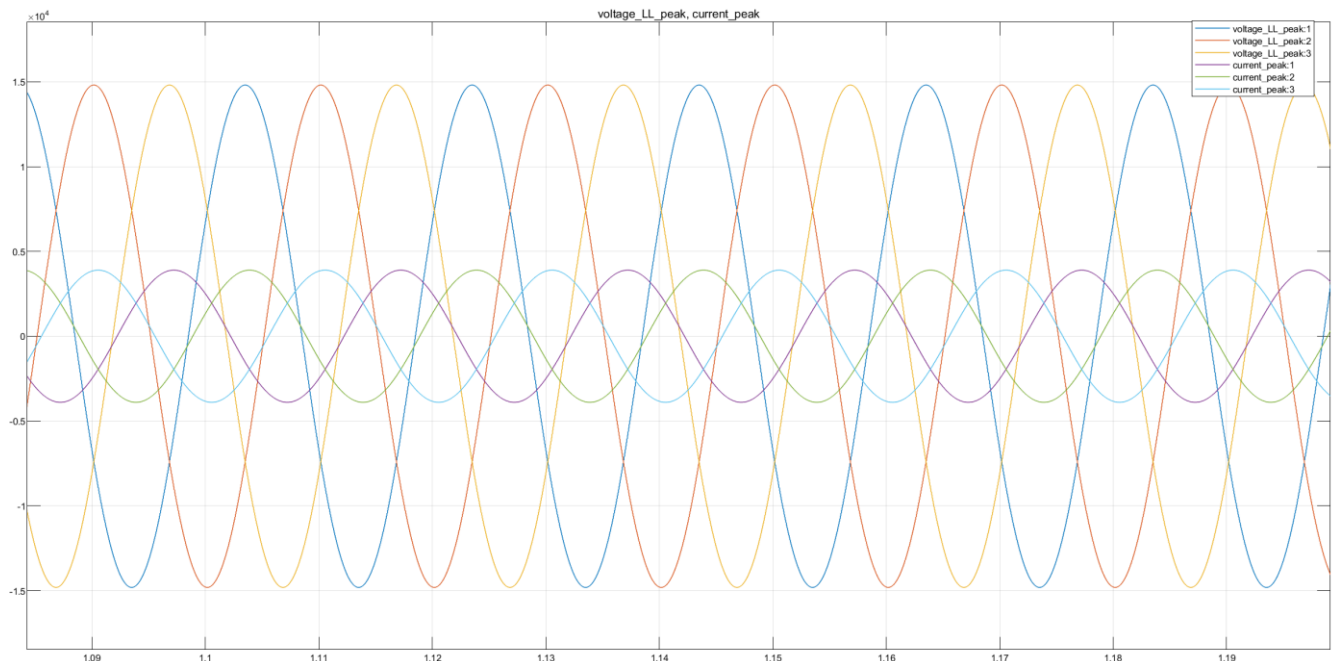


Figure 43: Line-to-line voltages and currents in peak values measured at bus 2

Using the Signal Statistics and Measurement tools in Simulink, both the peak and the RMS values of all signals can be assessed.

They are summarized in table 9 for a better comprehensibility.

Table 9: Measured voltages and currents at bus 2

	Peak value	RMS value
Line-to-line voltages	14.812,882 V	10.474,289 V
Phase currents	3.897,521 A	2.755,964 A

For the sake of completeness, the currents measured by current sensors incident to bus 2 are reported in figure 44:

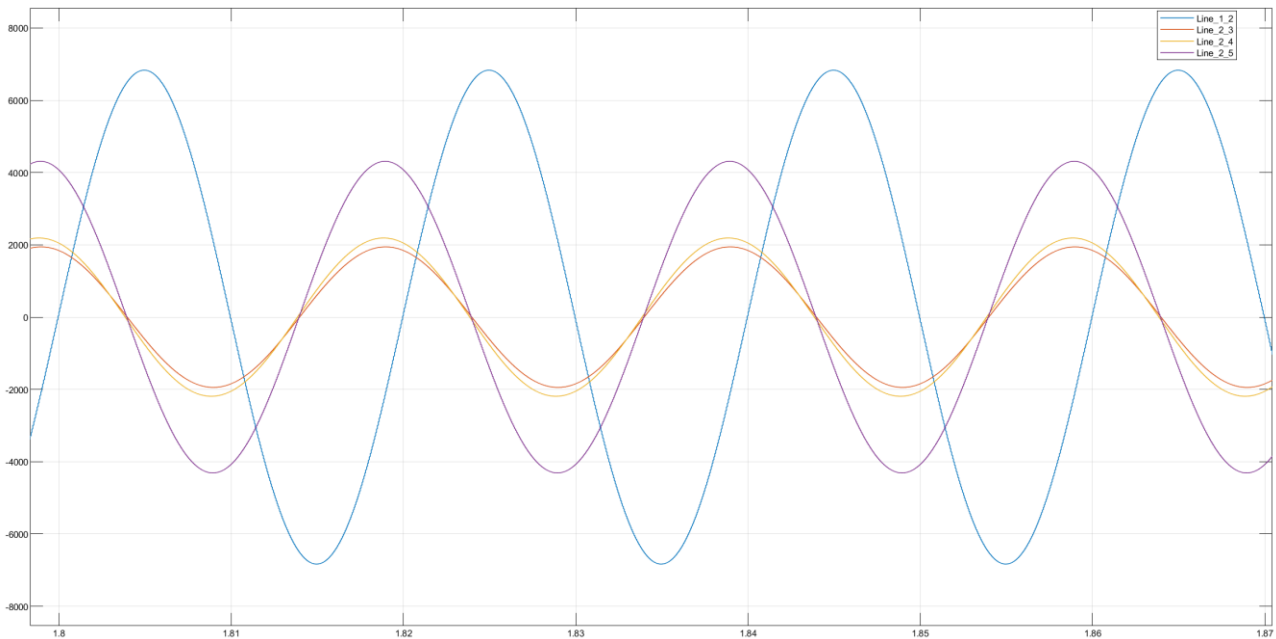


Figure 44: Currents measured in the lines incident to bus 2

Their peak and RMS values are summarized in table 10.

Table 10: Current sensors data in Simulink

Line Number	Max Value [A]	RMS [A]
1-2	$6.834 \cdot 10^3$	$4.770 \cdot 10^3$
2-3	$1.945 \cdot 10^3$	$1.393 \cdot 10^3$
2-4	$2.190 \cdot 10^3$	$1.568 \cdot 10^3$
2-5	$4.312 \cdot 10^3$	$3.088 \cdot 10^3$

This data would be necessary in a real-case scenario so that all the voltages in the other bus could be indirectly calculated.

In this next section, the obtained voltage and current synchrophasors at bus 2, obtained from the Simulink code, are preliminary analysed.

The PMU Voltage block provides in output the module and phase of the voltage synchrophasor, as depicted in figure 45.

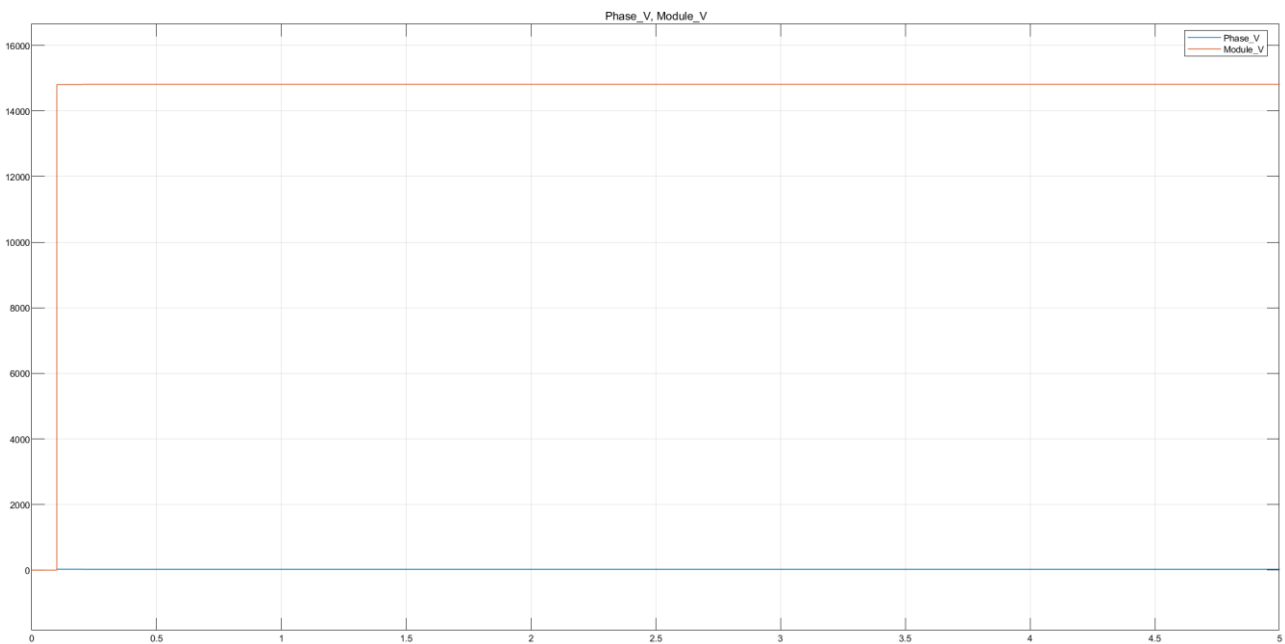


Figure 45: Module and phase of PMU Voltage

Taking an initial check at the measured values, one could affirm that the amplitude is roughly equal to 14.812,77 V, whereas the phase is about 26.79° . These values are indeed comparable to the ones measured in steady-state for phase AB.

The detected frequency is represented in figure 46 for both the voltage and current and, as it is clearly visible in the graph, the correct frequency of 50 Hz is measured.

It is worth highlighting the fact that the PLL circuits implemented inside the PMUs require about 200 ms in order to lock with the actual power system frequency of 50 Hz, as presented in figure 46. Indeed from 0.3 s henceforth, the detected frequency differs from the ideal one of a quantity in the range of 0.06 %. This oscillation is not present in real power systems which clearly do not have frequencies oscillating from 0 to 50 Hz, therefore this is the reason why the first 300 ms will be neglected in the indices calculation.

Nevertheless, further calculations concerning the accuracy of the measurements are carried out in the next paragraphs.

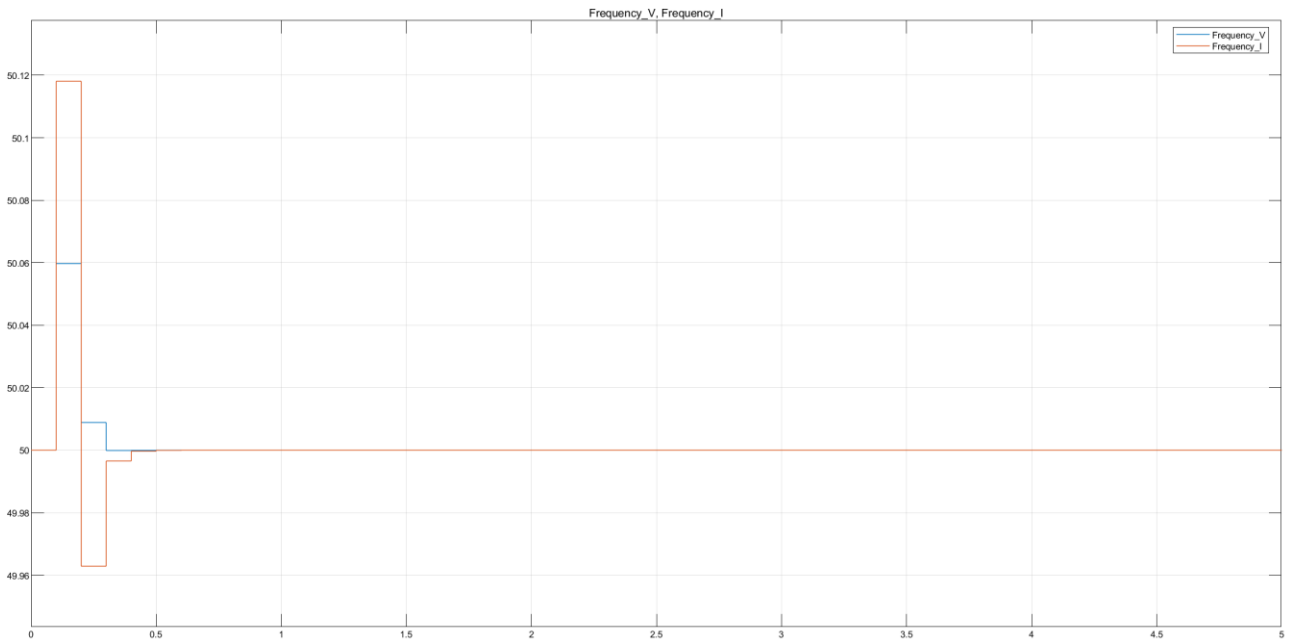


Figure 46: Detected frequency for voltage and current phasors

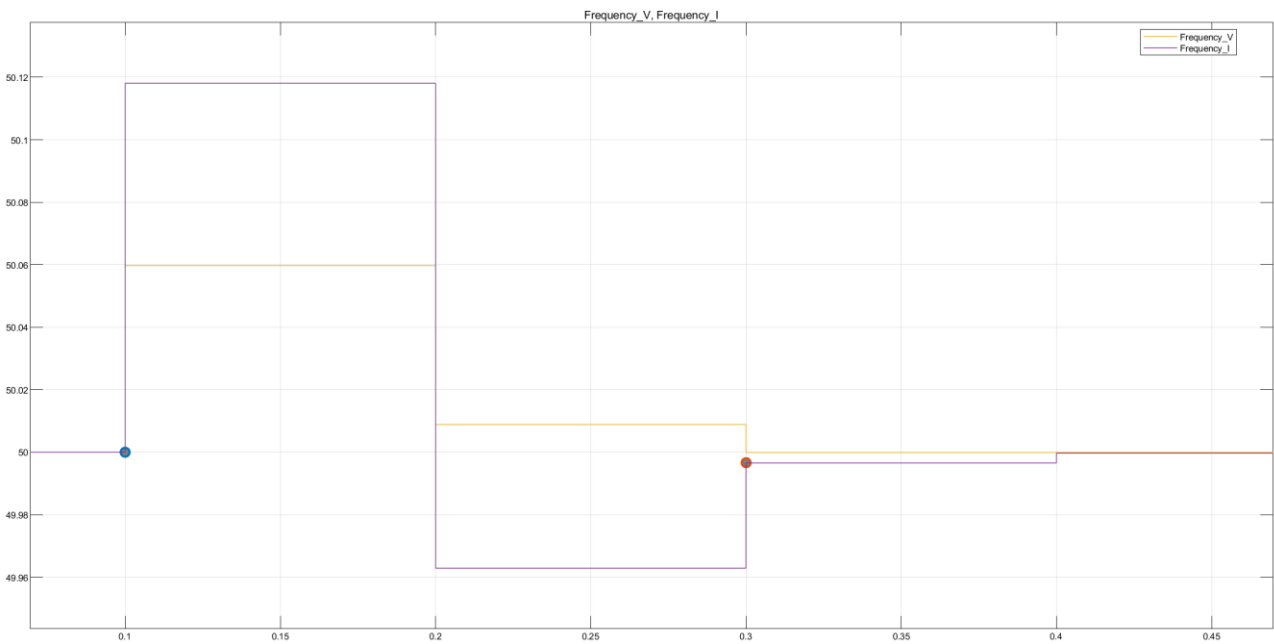


Figure 47: PLL circuit oscillations in the frequency detection

Inside the PMU Voltage block, the synchronous reference has been set equal to the peak value of the voltage in amplitude, so that it is possible to visually compare the two sinusoids, as previously anticipated: the ideal one and the PMU synchrophasor.

Considering the fact that the sensor measures only the positive sequence, to visualize the three sinusoids, a phase shift of $\pm 120^\circ$ has been applied to the obtained phasor. The results are plotted in figure 48.

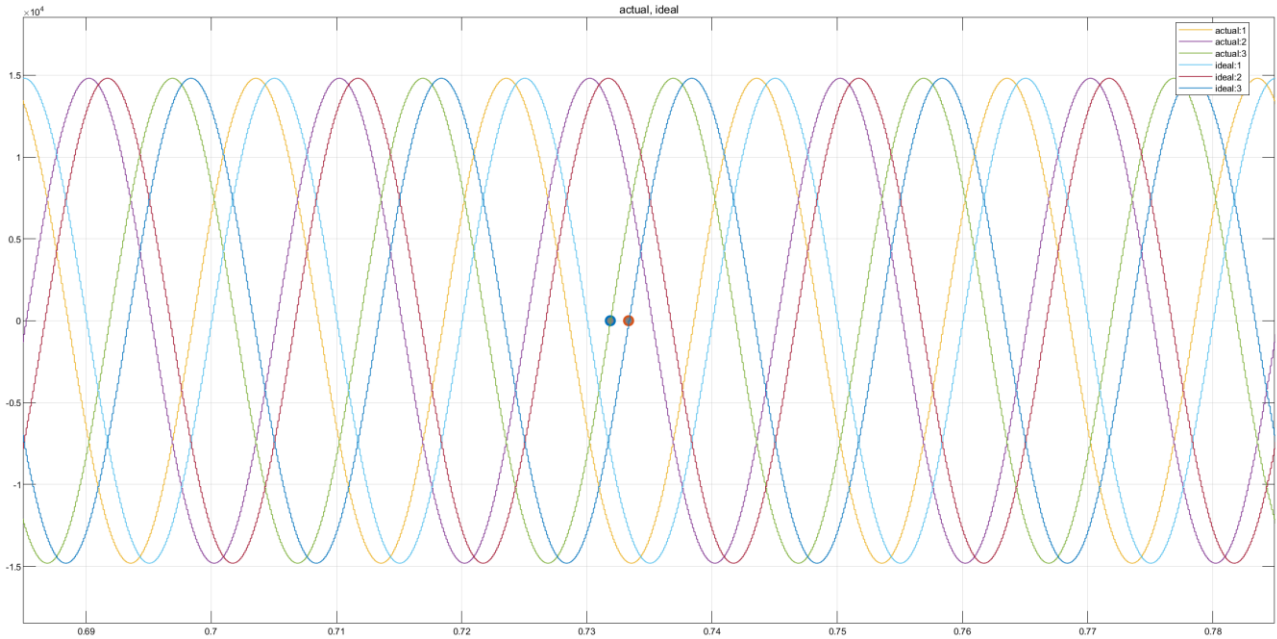


Figure 48: Comparison between ideal and measured voltages inside the PMU

In order to get a sense of the goodness of the results, a preliminary comparison in the phase shift has been carried out.

According to [9], the maximum phase error allowable in a 50 Hz power system is 0.573° if no amplitude error is measured.

In the graph illustrated in figure 48, a time delay between the ideal and the actual sinusoids of 1.458 ms has been measured.

Comparing the ratio:

$$20 \text{ ms} : 0.573^\circ = 1.458 \text{ ms} : x [^\circ] \quad (22)$$

One could say that this corresponds to a phase shift of 0.0418° which would confirm the compliance of the PMU, provided that no magnitude error is present.

Nonetheless, this rough calculation provides only an initial idea of the accuracy of the results.

At this point, the same analysis is carried out for the PMU Current measurements, for which the rationale is exactly the same.

In figure 49, the detected magnitude and frequency of the current synchrophasor are plotted.

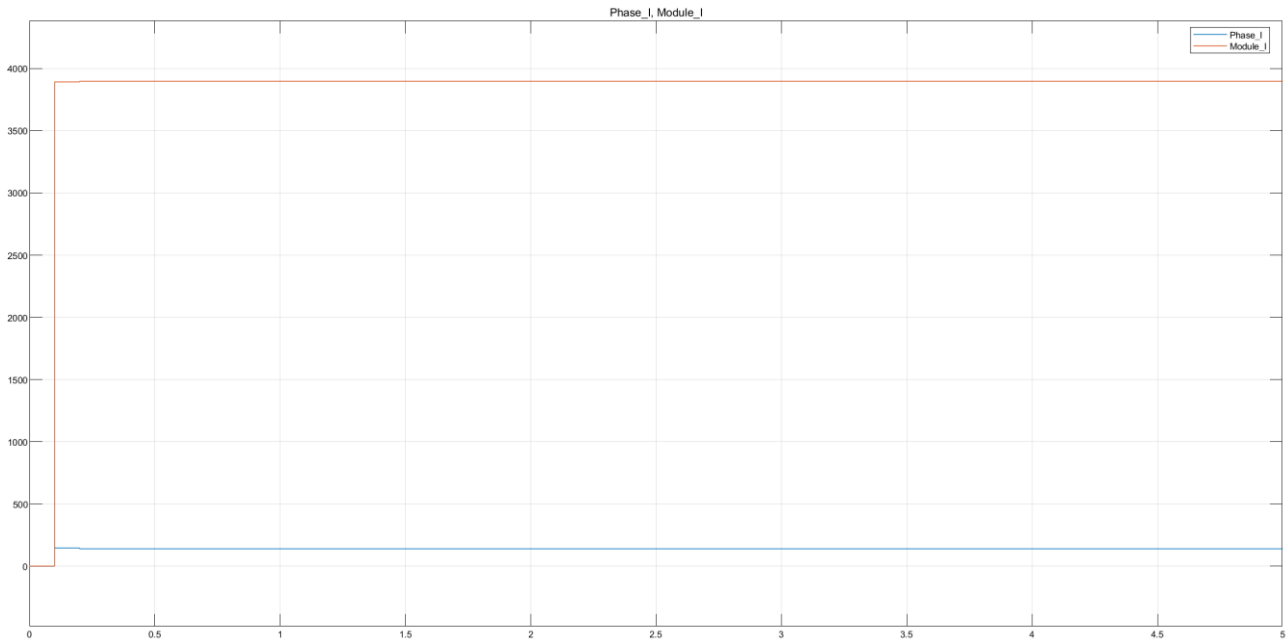


Figure 49: Module and phase of PMU Current

Taking a preliminary check at the measured values, one could affirm that the amplitude is roughly equal to 3.897,49 A, whereas the phase is about 139.91° . These values are indeed comparable to the ones measured in steady-state for the current of phase A.

The analysis of the detected frequency has already been carried out before together with the voltage, and the results are plotted in figure 46 and 47.

Lastly, a comparison between the phase shift has been carried out, similarly to the one performed for the voltages.

In the graph illustrated in figure 50, a time delay between the ideal and the actual sinusoids of 1.146 ms has been measured.

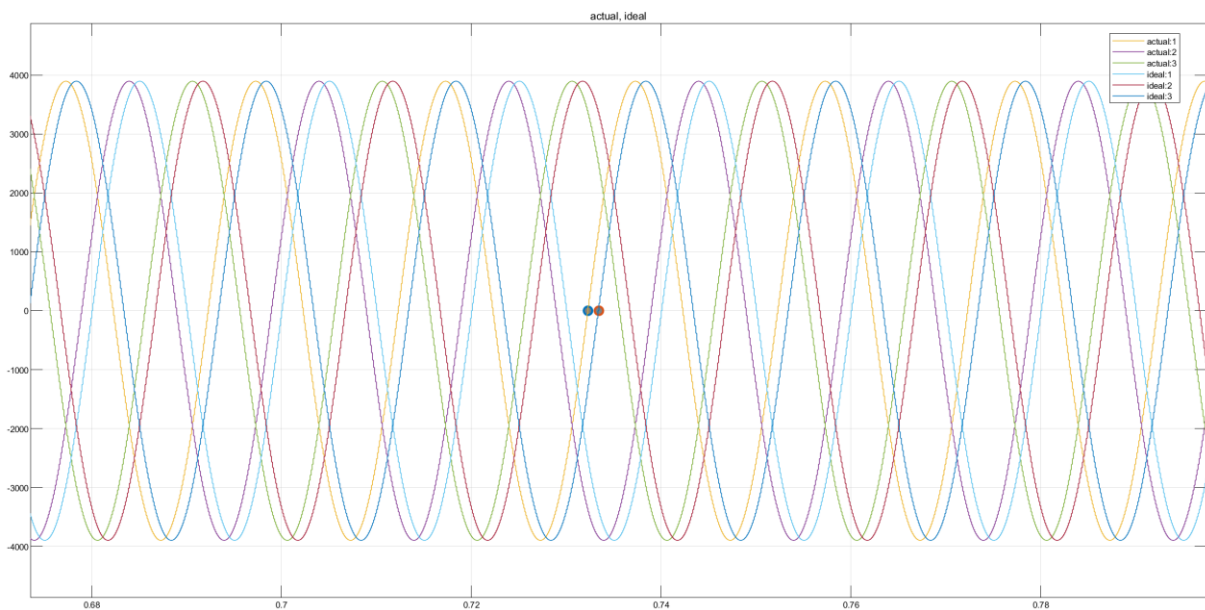


Figure 50: Comparison between ideal and measured currents inside the PMU

Comparing the ratio stated before in equation (22), one could say that this corresponds to a phase shift of 0.0328° which would confirm the compliance of the PMU, provided that no magnitude error is present.

Once again, this rough calculation provides only an initial idea of the accuracy of the results which needs to be further evaluated.

4.4.4 Computation of the accuracy indices

Having got an idea of the goodness of the results, a proper computation of the indices is required. To do that, the Indices Calculation block serves to this purpose as firstly anticipated. It is depicted in figure 51 and it has been implemented in such a way that only after 300 ms, the calculation of indices starts for which the reason has been explained in previous paragraph.

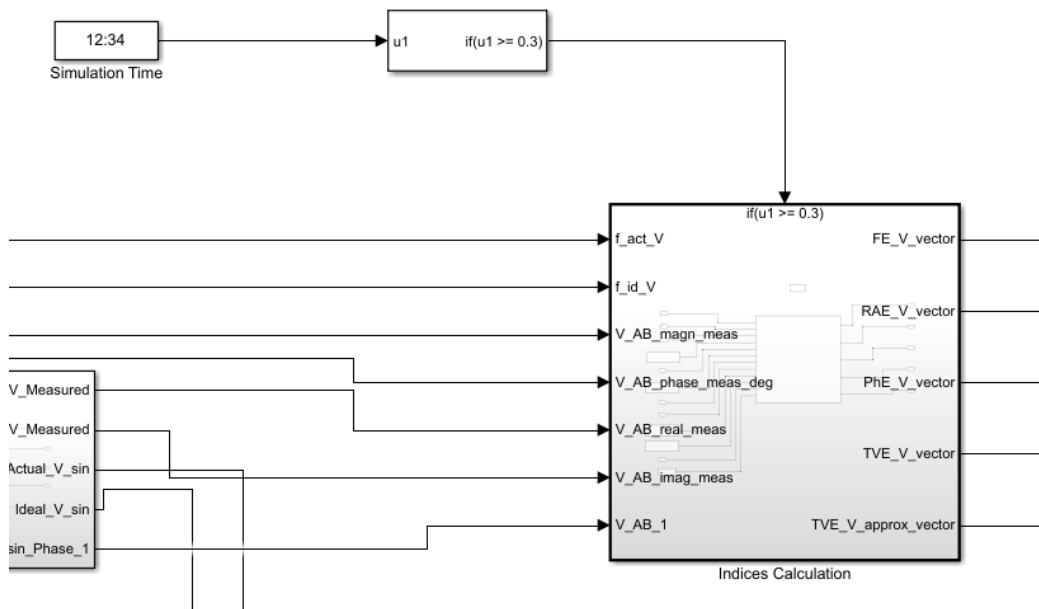


Figure 51: Indices Calculation block

This block needs as inputs all the measured quantities from the PMU, therefore the frequency, amplitude, phase in degrees, real and imaginary part of the voltage synchrophasor.

Inside it, measured values are compared and analysed with respect to steady-state data acquired by means of the previous Matlab code.

The content of the block and the Matlab function realized are visualized in figure 52 and 53, respectively.

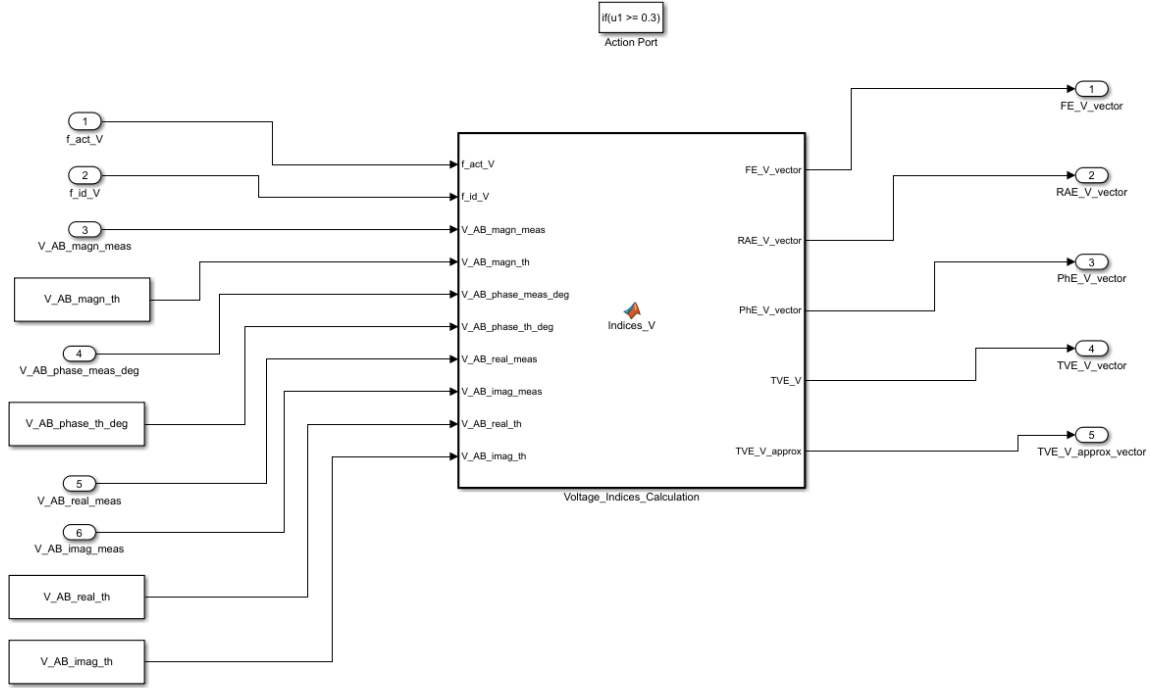


Figure 52: Matlab function in the Indices Calculation block

```

1 function [FE_V_vector, RAE_V_vector, PhE_V_vector, TVE_V, TVE_V_approx] = Indices_V (f_act_V, f_id_V, V_AB_magn_meas, V_AB_magn_th, V_AB_phase_meas_deg, V_AB_phase_th_deg,
2
3 FE_V_vector = abs(f_id_V - f_act_V); %calculating |ftrue-fmeasured|
4 delta_amp = V_AB_magn_meas - V_AB_magn_th; %calculating delta amplitude
5 RAE_V_vector = delta_amp/V_AB_magn_th; %calculating relative amplitude error
6 PhE_V_vector = abs(V_AB_phase_meas_deg - V_AB_phase_th_deg); %calculating phase angle error
7
8 TVE_V_approx = (((RAE_V_vector).^2) + ((PhE_V_vector).^2)).^(1/2); %approx TVE calculation
9
10 RE_diff = (V_AB_real_meas - V_AB_real_th).^2;
11 IM_diff = (V_AB_imag_meas - V_AB_imag_th).^2;
12 Reference_Input = (((V_AB_real_th).^2) + ((V_AB_imag_th).^2));
13
14 TVE_V = sqrt((RE_diff + IM_diff)./Reference_Input); %TVE calculation

```

Figure 53: Matlab script required to compute accuracy indices for the PMU Voltage block

For a better comprehensibility of the code, the implemented formulas in the Matlab script are reported as follows.

The FE is determined as in equation (10) but it is reported also here for the sake of completeness:

$$FE = |f_{true} - f_{measured}| \quad (23)$$

For real PMUs, typical values of FE range between 10 to 20 μ s.

The RAE is defined as in (24):

$$RAE = \frac{\tilde{V}-V}{V} = \frac{\Delta V}{V} \quad (24)$$

Where \tilde{V} is the measured voltage and V the nominal value.

The PhE is defined as in (25):

$$PhE = \tilde{\varphi} - \varphi \quad (25)$$

Where $\tilde{\varphi}$ is the measured angle and φ the nominal value.

The standard TVE formula is reported here, but it has been already defined as in equation (9):

$$\text{TVE}(n) = \sqrt{\frac{[\widehat{X}_r(n) - X_r(n)]^2 + [\widehat{X}_i(n) - X_i(n)]^2}{[X_r(n)]^2 + [X_i(n)]^2}} \quad (26)$$

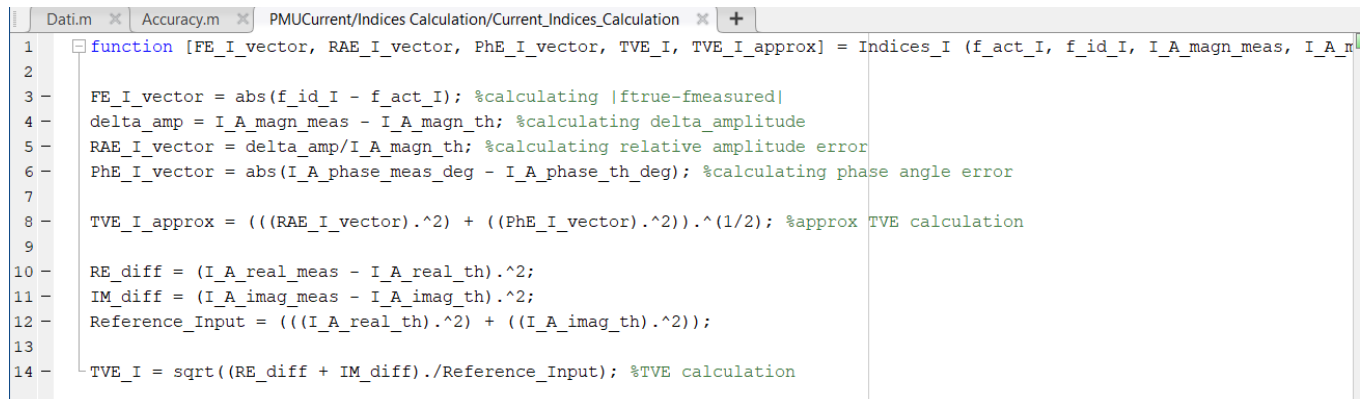
In this indices analysis, an approximated TVE formula present in literature [3] is also used and reported in (27). This has been done, so that its results can be compared with the ones coming from equation (26).

$$\text{TVE}_{\text{approx}} \cong \frac{|\tilde{a}e^{j\tilde{\phi}} - ae^{j\phi}|}{a} \cong \sqrt{\left(\frac{\Delta a}{a}\right)^2 + (\Delta\phi)^2} \quad (27)$$

Once again, it is of the utmost importance to verify that the TVE is lower or equal than 1% in all operating conditions, which means a RAE lower or equal than 1% or a PhE lower or equal than 1 crad, which is roughly equal to 0.573° for a 50 Hz system [9].

Performing these calculations, arrays for each of the accuracy indices are obtained and they are reported at 10 frames/s. They will be processed later on so that a unique value every second is obtained.

For the sake of completeness, the Matlab script computing accuracy indices in the PMU Current block is reported as follows in figure 54.



```

1 function [FE_I_vector, RAE_I_vector, PhE_I_vector, TVE_I, TVE_I_approx] = Indices_I (f_act_I, f_id_I, I_A_magn_meas, I_A_r
2
3 FE_I_vector = abs(f_id_I - f_act_I); %calculating |ftrue-fmeasured|
4 delta_amp = I_A_magn_meas - I_A_magn_th; %calculating delta_amplitude
5 RAE_I_vector = delta_amp/I_A_magn_th; %calculating relative amplitude error
6 PhE_I_vector = abs(I_A_phase_meas_deg - I_A_phase_th_deg); %calculating phase angle error
7
8 TVE_I_approx = (((RAE_I_vector).^2) + ((PhE_I_vector).^2)).^(1/2); %approx TVE calculation
9
10 RE_diff = (I_A_real_meas - I_A_real_th).^2;
11 IM_diff = (I_A_imag_meas - I_A_imag_th).^2;
12 Reference_Input = (((I_A_real_th).^2) + ((I_A_imag_th).^2));
13
14 TVE_I = sqrt((RE_diff + IM_diff)./Reference_Input); %TVE calculation

```

Figure 54: Matlab script required to compute accuracy indices for the PMU Current block

In this following section, the results of the voltage indices calculations are reported.

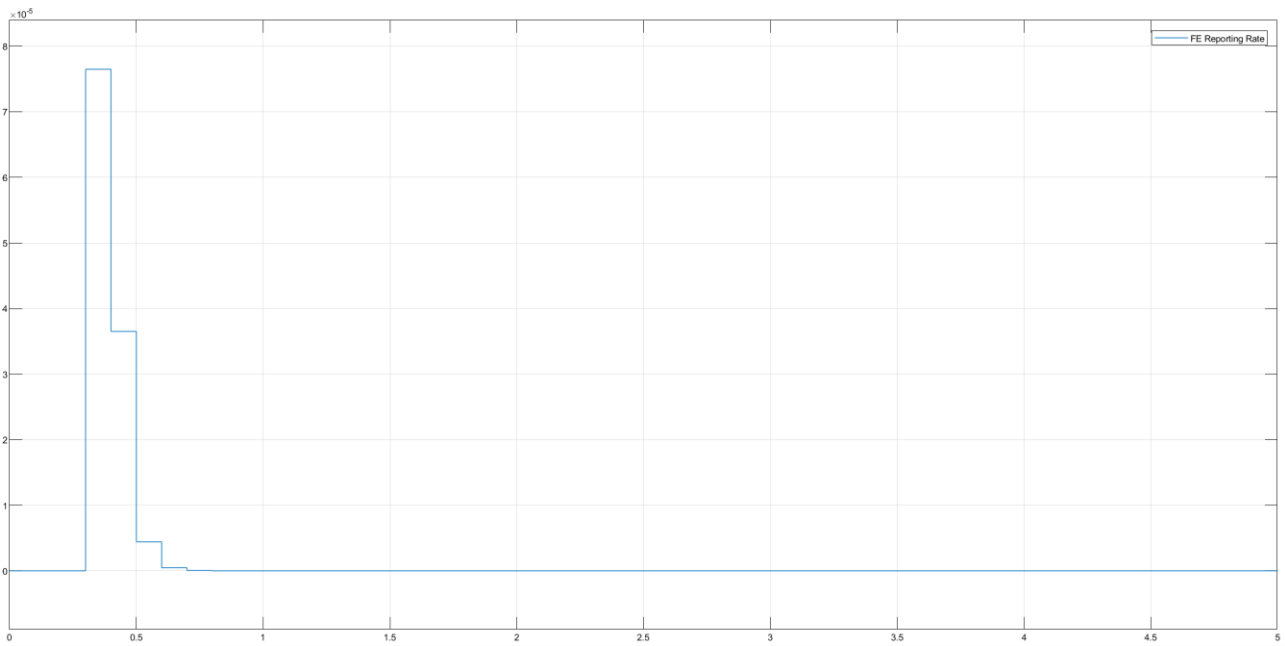


Figure 55: Voltage FE

After a spike of about $7.5 \cdot 10^{-5}$ between 0.3 to 0.6 s, during which the PLL still needs to perfectly lock to the power system frequency, the FE stabilizes around zero, meaning that 50 Hz have been correctly detected.

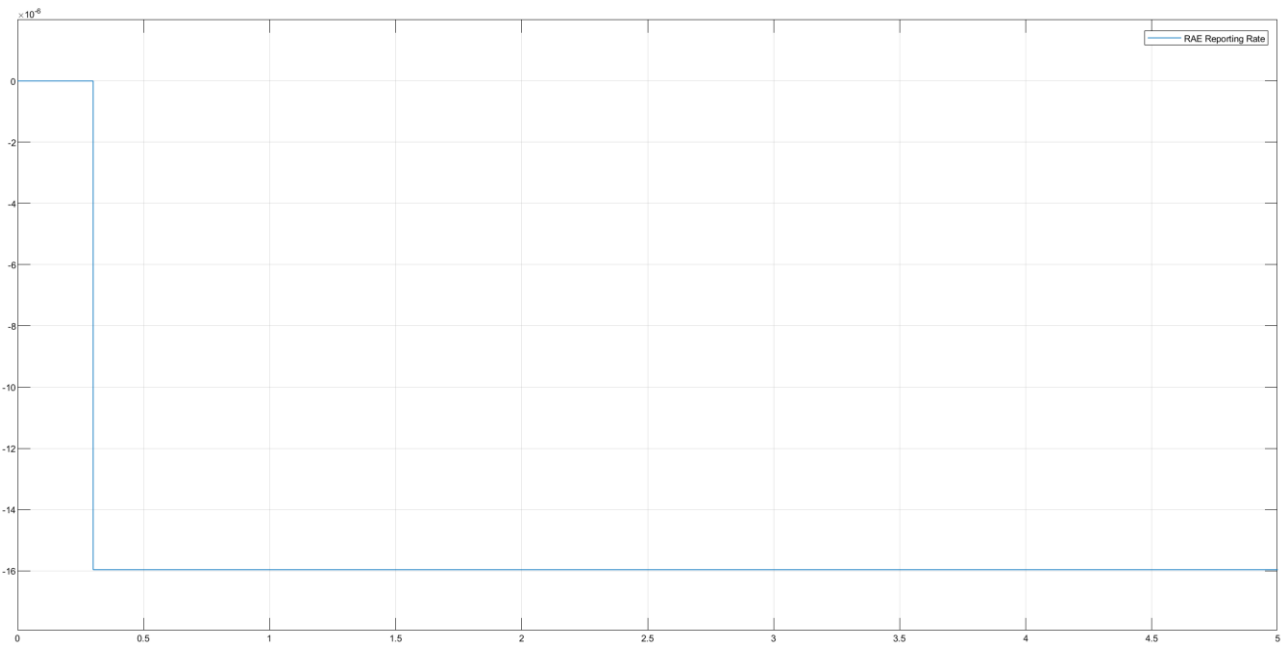


Figure 56: Voltage RAE

The RAE is equal to zero until 0.3 s because indices are not calculated before that time, but after that this index shows a stable value around $-16 \cdot 10^{-6}$.

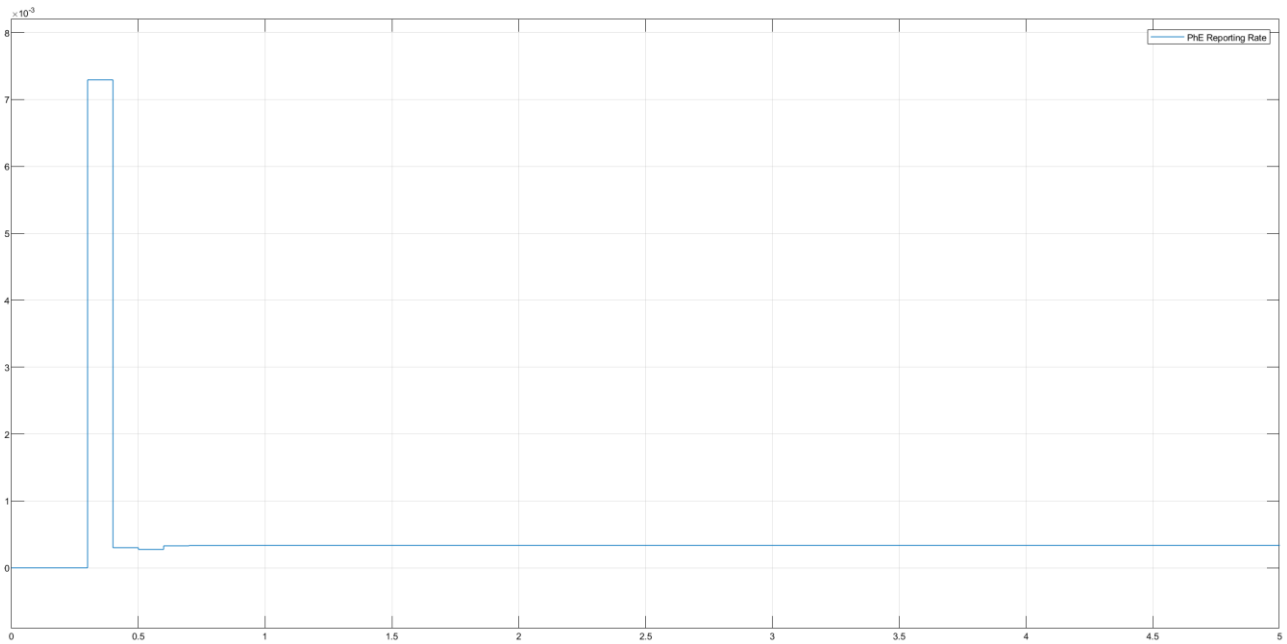


Figure 57: Voltage PhE

After the stabilization of the PLL circuit, the voltage PhE is stable around $0.5 \cdot 10^{-3}$.

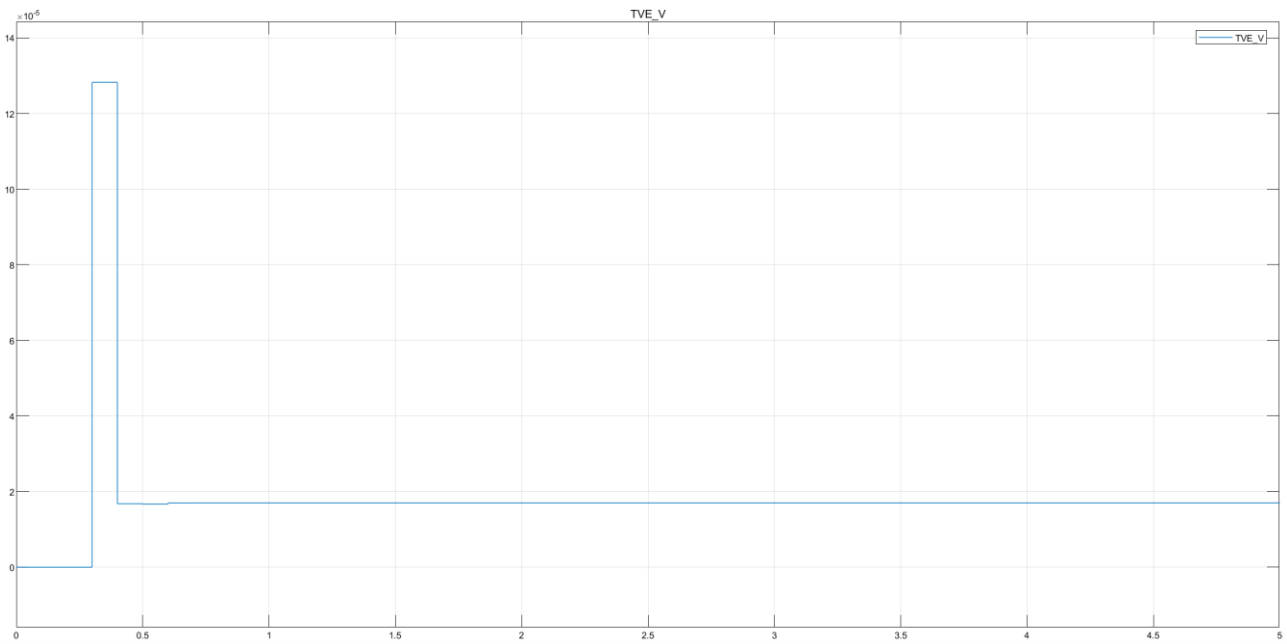


Figure 58: Voltage TVE

Even before proceeding with further calculations, one could notice that the voltage TVE at steady-state is slightly less than $2 \cdot 10^{-5}$, meaning that it is much below the required 1% ensuring the compliance of the PMU.

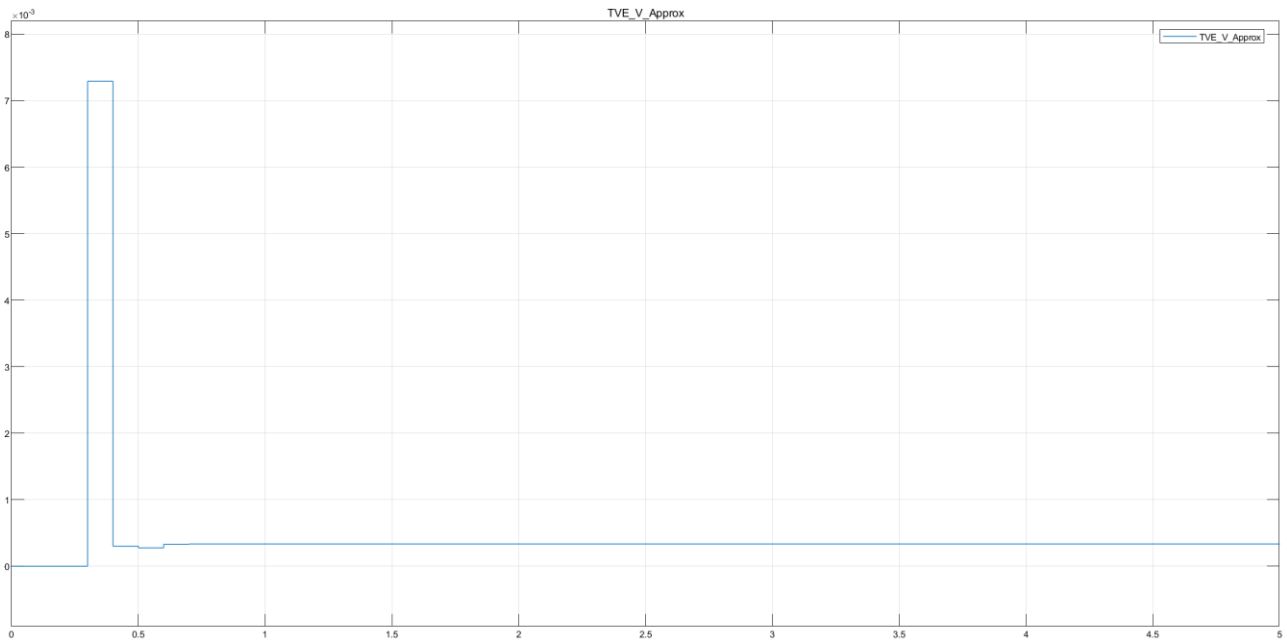


Figure 59: Approximate Voltage TVE

The same rationale holds for the approximate voltage TVE, as it implies the compliance of the PMU albeit a bigger value is calculated with respect to the TVE in figure 58.

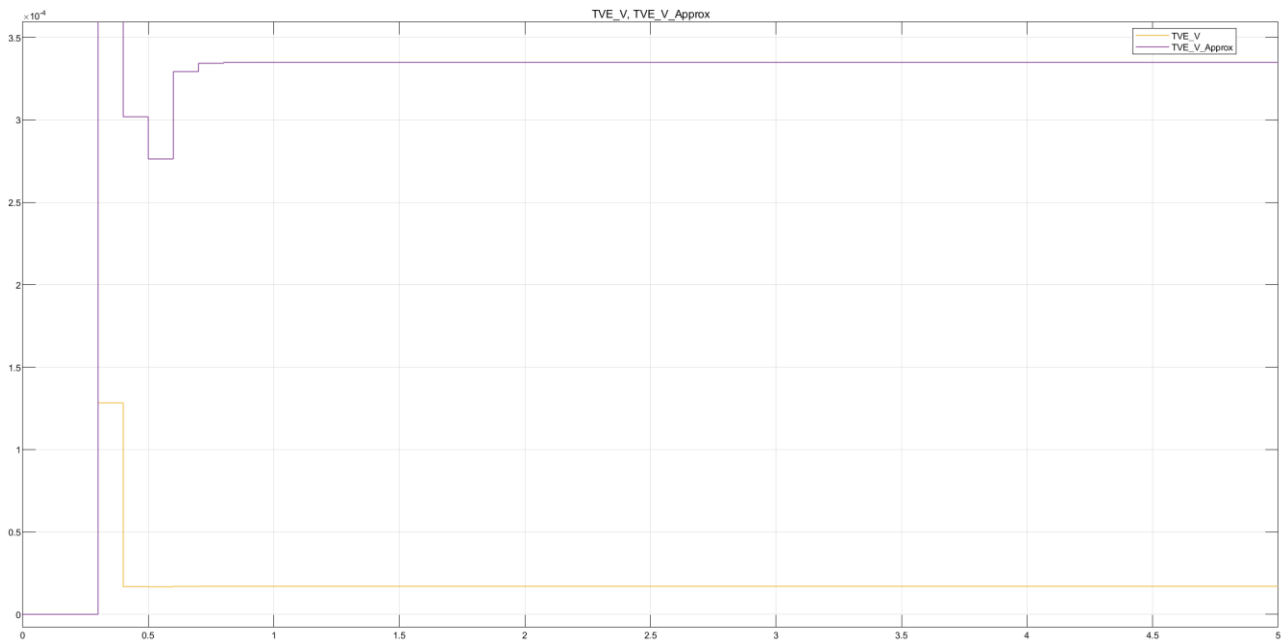


Figure 60: Comparison between voltage TVE and approximate TVE

The comparison of the two TVE formulas is depicted in figure 60. At steady-state, the TVE is about $1.6 \cdot 10^{-5}$ whereas the approximate one is roughly $3.4 \cdot 10^{-4}$. Hence, their difference is in the order of $3.2 \cdot 10^{-4}$, meaning that the approximate TVE is overestimating the voltage TVE.

The same indices analysis is reported as follows for the PMU current block. This data has been plotted after the reporting rate generator which provides one only with 10 frames/s information.

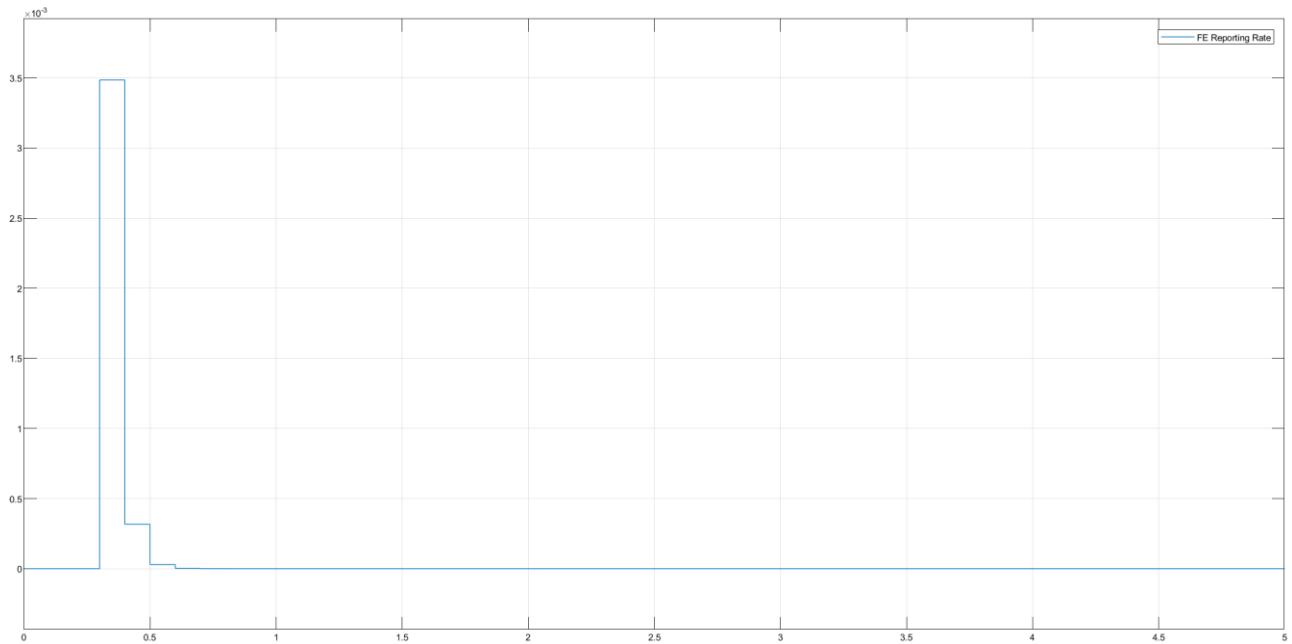


Figure 61: Current FE

As in figure 55, a spike of about $3 \cdot 10^{-3}$ between 0.3 to 0.6 s is present due to the PLL trying to perfectly lock to the power system frequency. After that, the FE stabilizes around zero, meaning that 50 Hz have been correctly detected, as for the voltage FE.

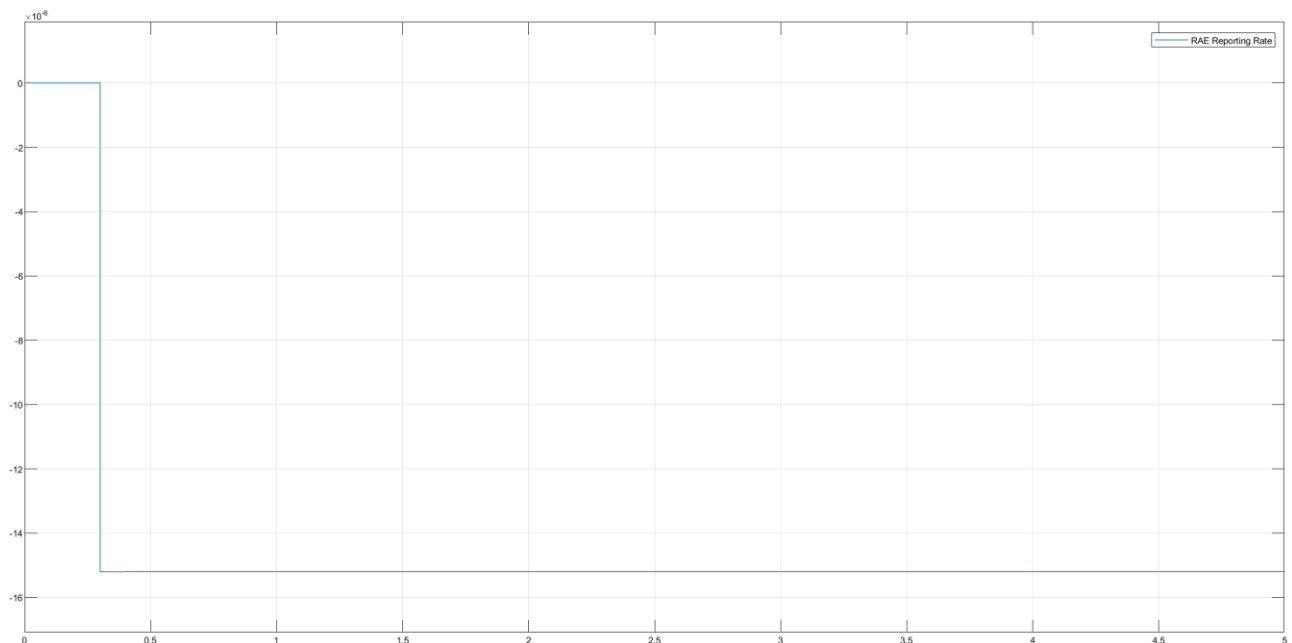


Figure 62: Current RAE

The current RAE is slightly bigger than the voltage one, but at steady-state it is about $-15 \cdot 10^{-6}$.

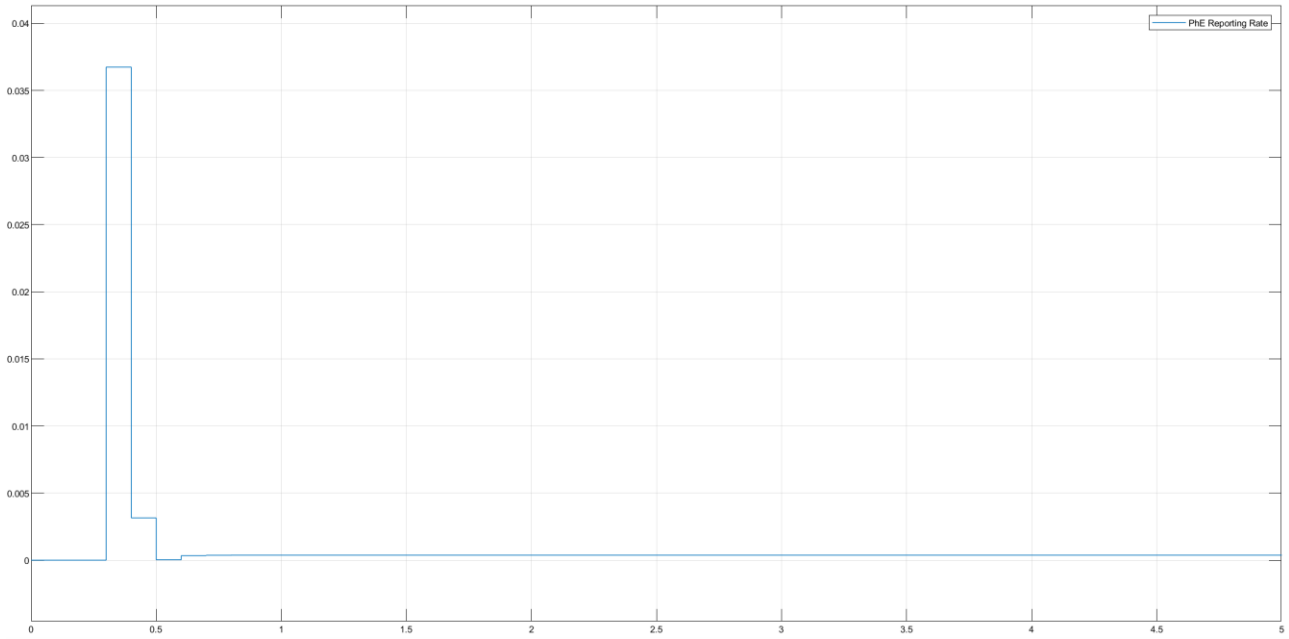


Figure 63: Current PhE

After the stabilization of the PLL circuit, the current PhE is stable around a value slightly bigger than 0 which will be properly evaluated in Matlab.

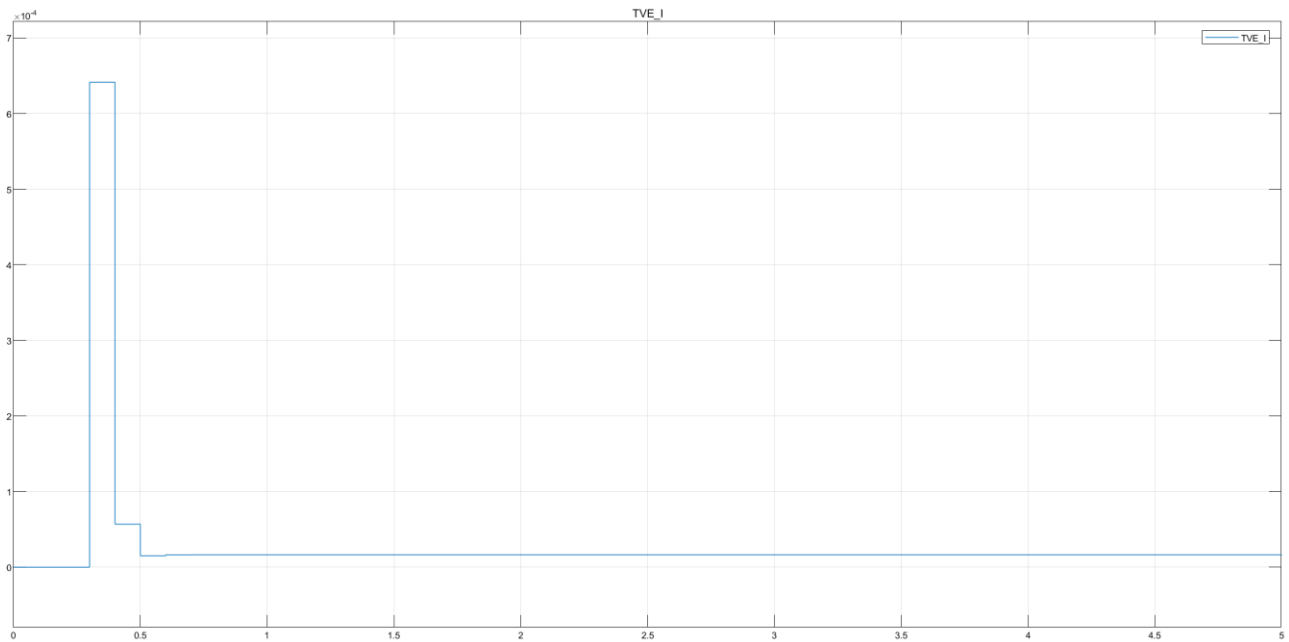


Figure 64: Current TVE

The same rationale used for the voltage TVE can be applied here, as even before proceeding with further calculations, one could notice that the current TVE at steady-state is less than $1 \cdot 10^{-4}$, meaning that it is much below the required 1% ensuring the compliance of the PMU.

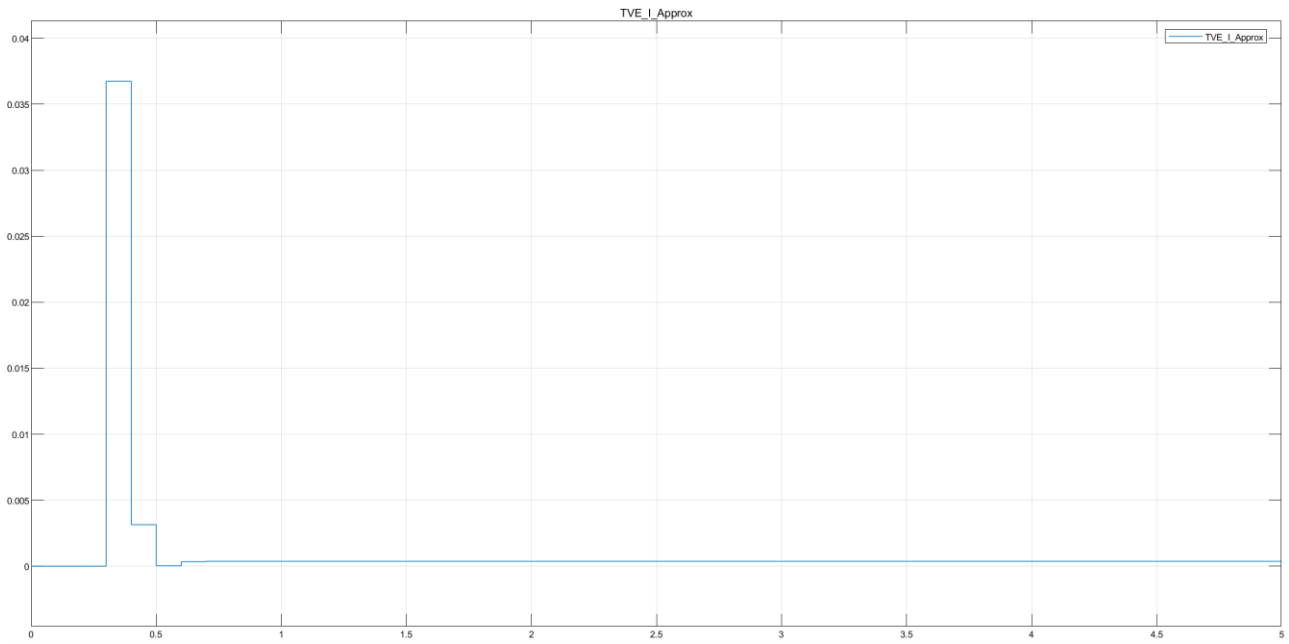


Figure 65: Approximate Current TVE

The same motivation holds for the approximate current TVE, as it implies the compliance of the PMU although a bigger value is calculated with respect to the TVE in figure 64.

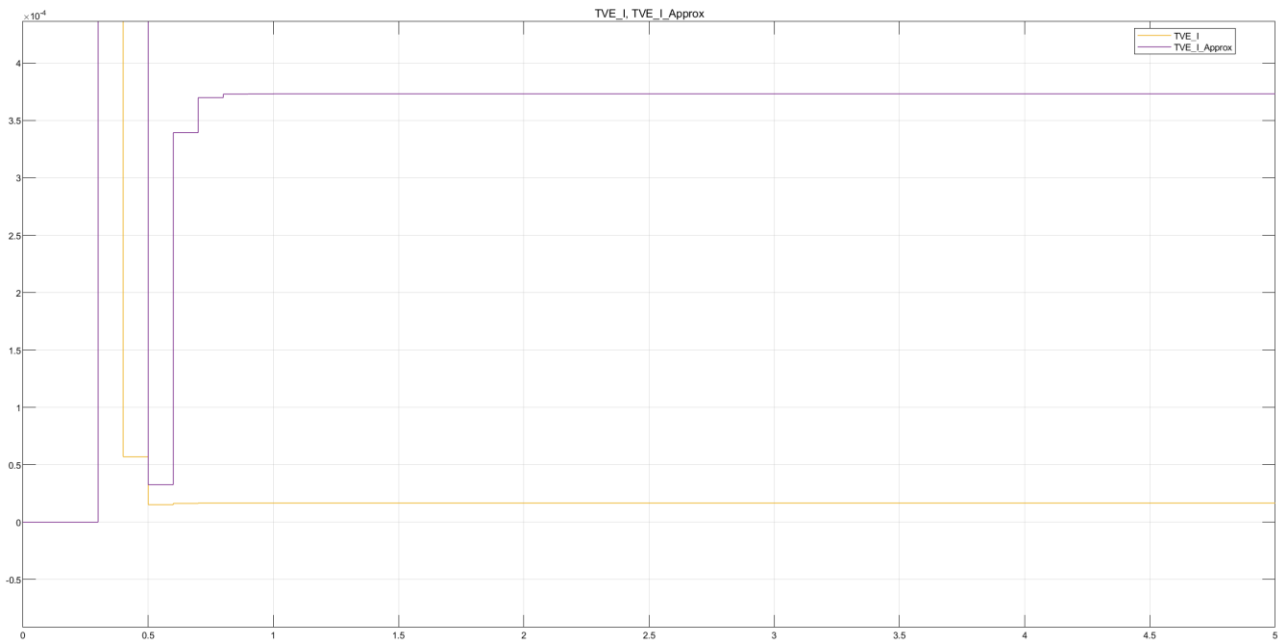


Figure 66: Comparison between current TVE and approximate TVE

The comparison of the two TVE formulas is shown in figure 66. At steady-state, the TVE is about $1.6 \cdot 10^{-5}$ whereas the approximate one is roughly $3.7 \cdot 10^{-4}$.

Hence, their difference is in the order of $3.6 \cdot 10^{-4}$, meaning that the approximate TVE is overestimating the current TVE as for the voltage.

Having obtained a preliminary confirmation of the compliance of the PMU, all this data will be processed in a Matlab script so that the average value and the standard deviation of 10 subsequent frames is calculated; meaning that every second, a useful information is provided.

This analysis will be carried out in paragraph 4.5.3 together with the data obtained from OPAL real-time simulator.

4.5 OPAL Real-Time Simulator

4.5.1 Basic structure

Before moving on to the description of the Real-Time Simulator which will be used for testing the realized Simulink scheme, one may wonder what difference would make between running a simulation in Simulink or running one in real-time.

A real-time simulation refers to a physical system that can execute a model at the same rate as the actual "wall clock" time. It means that, for instance, if a tank takes 10 minutes to fill in the real world, the simulation would take 10 minutes as well [46].

The real-time simulation is performed using a discrete time with a constant time step, known as "fixed-step simulation" as the time moves forward in equal duration of time. The time required to solve the internal state equations and functions representing the system must be less than the fixed step otherwise "overrun" occurs, as shortly anticipated in paragraph 3.1.1.

Therefore, a real-time simulation progresses at the same rate as the actual time passes and the simulation time step can be anything from 1 μ s sampling time up to minutes. On the contrary, a computer simulation can be slower or faster than the real time [47].

What is more important is that, with a real-time simulation, one can test ideas in seconds to see how changes will impact on a project.

Considering a typical development process, the simulation is usually placed after the design nears completion. At this stage of the process, making unsafe and/or risky changes to the design affects simulations, as changes might be expensive or time consuming. Moreover, cost-to-fix escalates exponentially as project progress, and by the time a design reaches simulation, decisions regarding materials and manufacturing processes have already been done.

With real-time simulations, instead, the development process is compressed in such a way that the simulation takes place during concept and design phases, as design decisions are being made. In this way, multiple changes can be evaluated before the prototype realization [48].

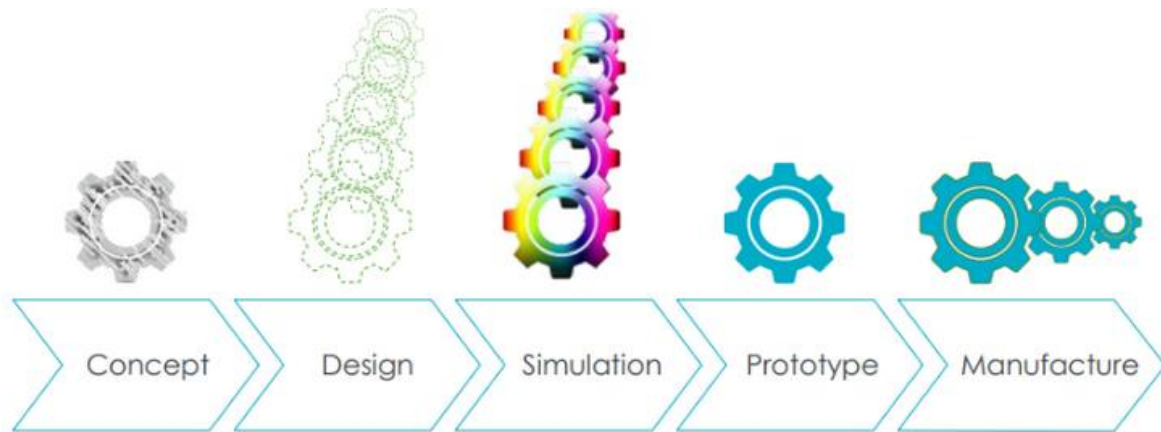


Figure 67: Typical development process. Adapted from [48]



Figure 68: New approach involving simulations during the concept and design phases. Adapted from [48]

The hardware device which will be used for the practical monitoring in real-time of the installed PMU in the IEEE 5-bus network is OPAL RT-4510.

It is a real-time digital simulator suitable for RCP and HIL application. It is equipped with 128 fast I/O channels with signal conditioning and additional high-speed communication ports [49].

The full range of I/O channels is depicted in figure 69, whereas the simulator structure is summarized in figure 70:

- A** Small Form Factor (SFP) 5Gbits/s optical interface modules connectors
- B** JTAG, USB and ETHERNET connectors
- C** Synchronization connectors and status LEDs
- D** Target computer status LEDs

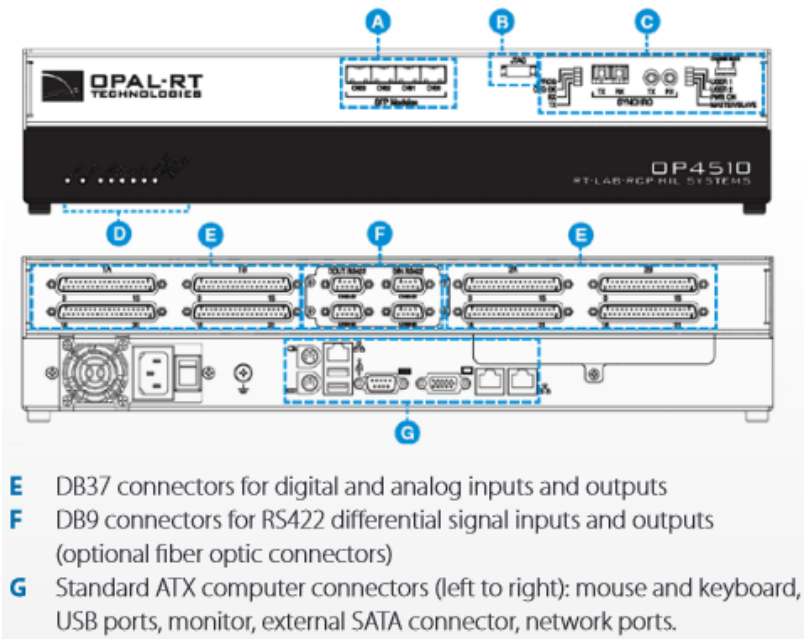


Figure 69: OPAL RT-4510 I/O channels. Adapted from [50]

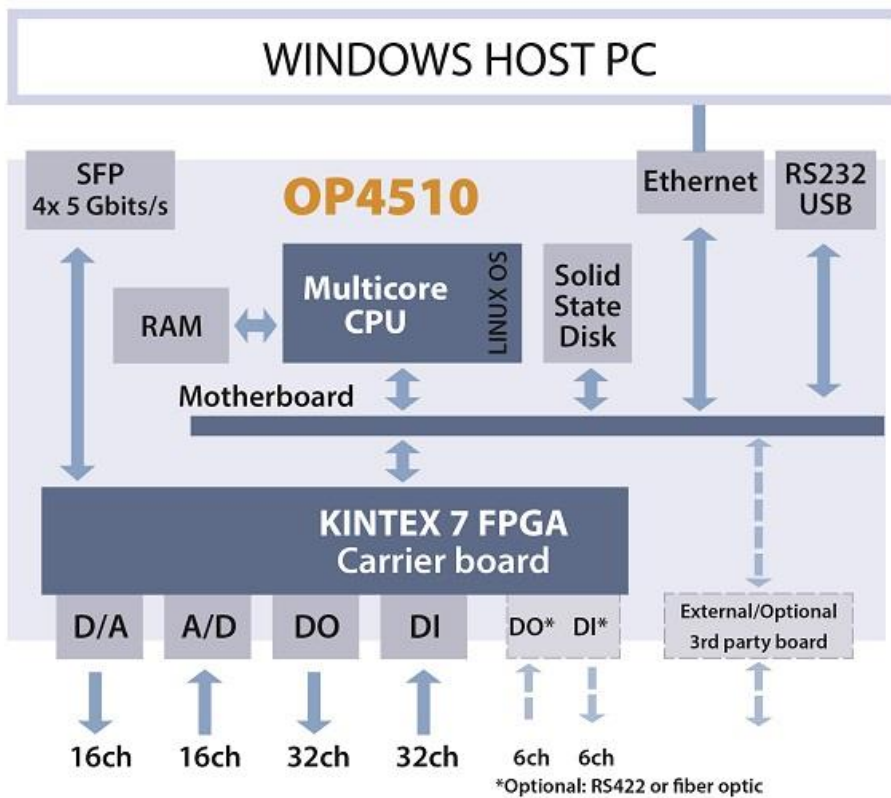


Figure 70: OPAL RT-4510 internal structure. Adapted from [50]

This device made up of a multicore Central Processing Units (CPUs) and an FPGA. This integration allows one to reach real-time simulation time steps below 7 μ s and 250 ns, respectively for model subsystems executed on the CPU and on the FPGA, with a time resolution up to 10 ns [51].

It is worth saying that the description of the I/O channels has been emphasized only for the sake of completeness since no inputs or outputs will be used in the practical implementation.

The latter, indeed, will consist in the deployment of the realized model of the network and the installed PMU in the simulator.

This test is necessary in order to make sure that the model behaves as expected. Furthermore, data collected during these preliminary assessments, also regarded as “Model-In-the-Loop” simulations (MIL), can be used for reference during future tests, as the SIL and HIL implementations.

4.5.2 Re-arrangement of the Simulink code

RT-LAB is the software which needs to be used as an interface with OPAL RT-4510. It is necessary to load, execute and run projects in the real-time simulator.

In order to properly upload Simulink projects in the simulator, some re-arrangement of the code must be performed in advance.

In RT-LAB platforms, subsystems must be created and they have two uses only:

1. Distinguish computation elements, as mathematical calculations, from the Graphical User Interface (GUI) elements, such as inputs, scopes and displays.
2. Assign separate computation processes to separate CPU cores. Owing to the simplicity of the implemented model, this part will not be considered.

Taking a look at the hardware in figure 71, one can understand why these subsystems have been made.

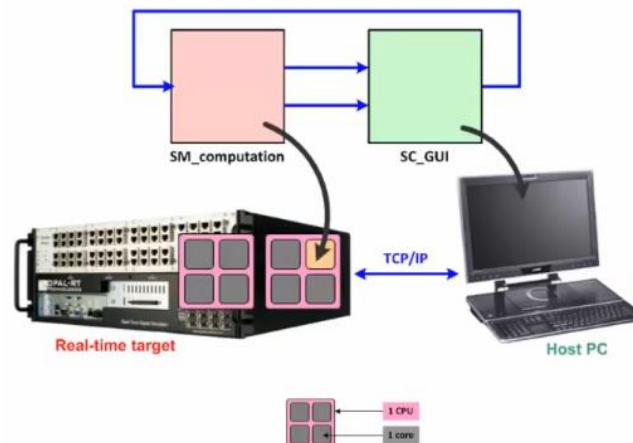


Figure 71: Typical OPAL RT simulator setup. Adapted from [52]

The computation block will be assigned to one core of the simulator, also known as real-time target (displayed on the left in figure 71), whereas the GUI block will be assigned to the host pc (on the right, in the same figure).

It is worth underlying that prefixes carry a particular meaning in RT-LAB, indeed:

- SC: It is the prefix of the GUI block and there must be no more than one of these in the Simulink model.
- SM: It stands for Master Subsystem. There must always be one master subsystem in the scheme.
- SS: It stands for Slave Subsystem. They are additional blocks which can be added in order to increase the computation power and they can be added as many as the number of available cores.

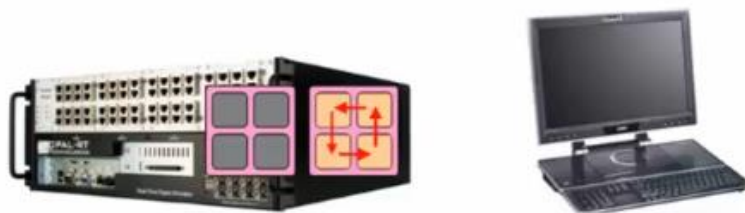
Nonetheless, the simplicity of the code allows one to have only one SC and one SM blocks, thus the simulation setup will resemble the one displayed in figure 71.

Having defined the type and number of blocks needed, the attention is now focused on the communication between subsystems.

Two types of communications between subsystems exists:

- *Synchronous communication*: it is used for communications between CPU cores. It is characterised by extremely high speed and it is capable of real-time simulation.
- *Asynchronous communication*: it is used for communications between the Target and the Host. As it is relatively low speed, it is not capable of Real-Time Simulation, thus it is just used for proving feedback to the user.

1. Synchronous Communication



2. Asynchronous Communication

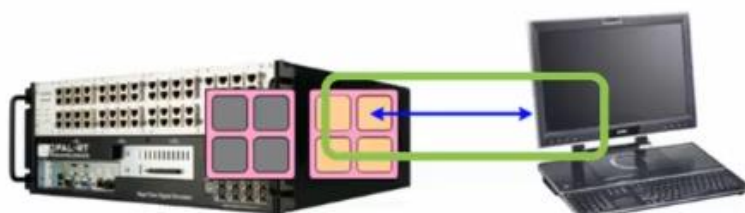


Figure 72: Communication between Subsystems. Adapted from [52]

All the inputs of the subsystems (SM, SS and SC) must first go through an OpComm block before any operation can be done to them. This is done in order to properly read signals, before analysing and elaborating them.

These OpComm blocks are used in SM and SS subsystems to receive real-time-synchronized signals from other real-time subsystems, whereas the OpComm block in the GUI subsystem is the only one receiving asynchronous data to be displayed.

In the considered case study, the SM subsystem will receive only asynchronous signals from the SC block, as in figure 73.

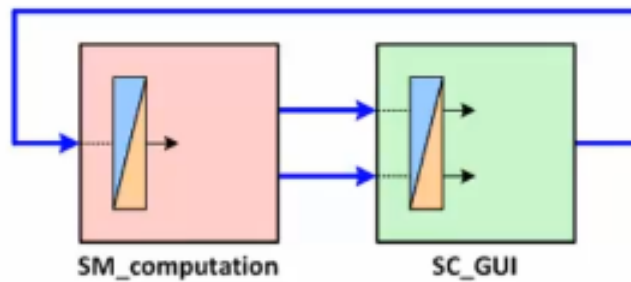


Figure 73: OpComm blocks placement. Adapted from [52]

Having explained all of this, the Simulink code has been properly re-organized following all the needed rules. The re-arranged code is now illustrated in figure 74.

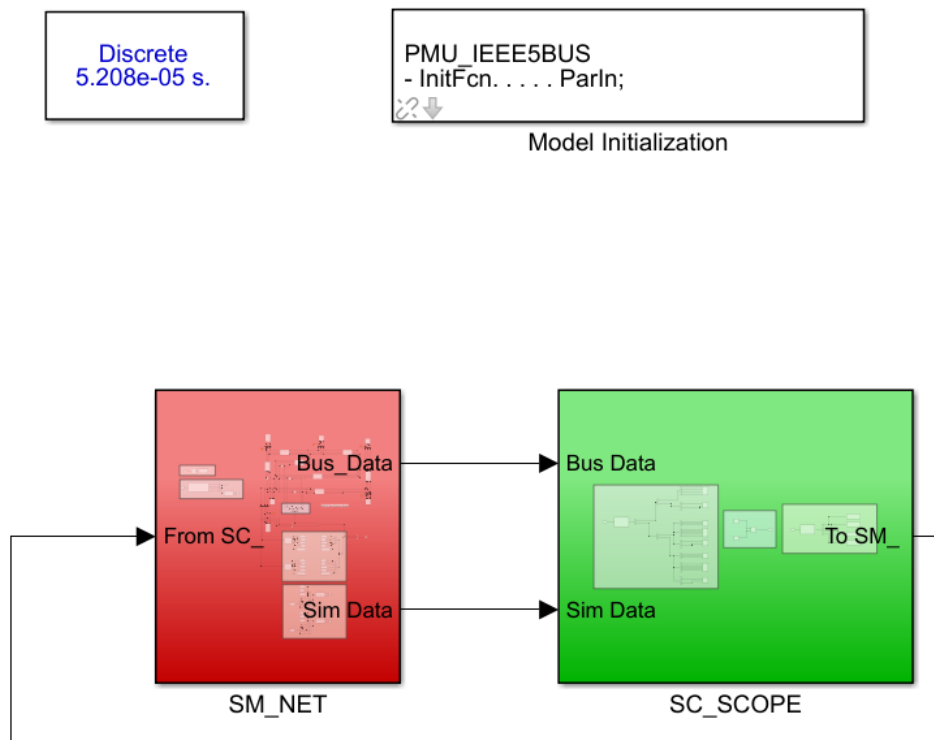


Figure 74: Re-arranged code for implementation on OPAL-RT

The real-time simulation is carried out at a fixed-time step equal to the Simulink simulation, thus at 19200 Hz, corresponding to $5.208 \cdot 10^{-5}$ s or 52.08 μ s.

In figure 74, one could also notice a Model Initialization block. It is needed to initialize the PMU data with the correct sampling time and reporting rate.

It is also required to import load flow steady-state data for the calculation of the accuracy indices.

This block allows one to execute a Matlab script before the compilation of the code in RT-LAB, therefore acquiring all the necessary variables. The code is shown in figure 75.

```

1  %%% PMU PARAMETERS %%%
2 - nominal_frequency = 50; %[Hz]
3 - nominal_period = 1/nominal_frequency; %[s]
4 - reporting_rate = 64; %Sampling_Rate Nsr in the PMU mask
5 - f_rep_rate = reporting_rate/nominal_period; %[Hz]
6 - requested_frame_second = 10; %[frames/s] requested (according to C37.118.1)
7 - reporting_rate_factor = f_rep_rate/requested_frame_second; %[] -> it reduces the # of reported points in output
8 -                               %to get only the required # of Frames/s
9
10 %PMU sampling time
11 - Sampling_Time = 1/(nominal_frequency*reporting_rate); %[s]
12 - k = 3; %[] choosing a submultiple of Ts for ADC of the PMU
13 - Sa_t = 1/(k*nominal_frequency*reporting_rate); %[s] -> actual sampling time of the ADC
14 - f_sampling = 1/Sa_t; %[Hz] -> actual sampling frequency of the PMU
15
16
17 %%% Load Flow Steady-State Data %%%
18 - V_AB_th = 1.317576485778211e+04 + 6.769579534029052e+03i; %From LFA
19 - V_AB_magn_th = abs(V_AB_th);
20 - V_AB_phase_th_rad = angle(V_AB_th); %phase in radians
21 - V_AB_phase_th_deg = 180*V_AB_phase_th_rad/pi; %phase in degrees
22 - V_AB_real_th = real(V_AB_th); %real part of theoretical phasor
23 - V_AB_imag_th = imag(V_AB_th); %imag part of theoretical phasor
24
25 %Conversion into magnitude/phase format for the theoretical current
26 - I_A_th = -2.999835564507649e+03 + 2.488395656144568e+03i; %From LFA
27 - I_A_magn_th = abs(I_A_th);
28 - I_A_phase_th_rad = angle(I_A_th); %phase in radians
29 - I_A_phase_th_deg = 180*I_A_phase_th_rad/pi; %phase in degrees
30 - I_real_th = real(I_A_th); %real part of theoretical phasor
31 - I_imag_th = imag(I_A_th); %imag part of theoretical phasor

```

Figure 75: Matlab script for data initialization in RT-LAB

The re-arranged code of figure 74 shown the two main subsystems previously described.

As previously explained, in the SM_NET subsystem all the computational part is implemented together with the IEEE 5-bus system as well as the voltage and current PMUs.

The indices calculations are also done in here.

On the contrary, in the SC_SCOPE subsystem the GUI is realized. This subsystem is the only one which will be accessible and visible during the real-time simulation.

In the next sections, the content of these two blocks will be analysed.

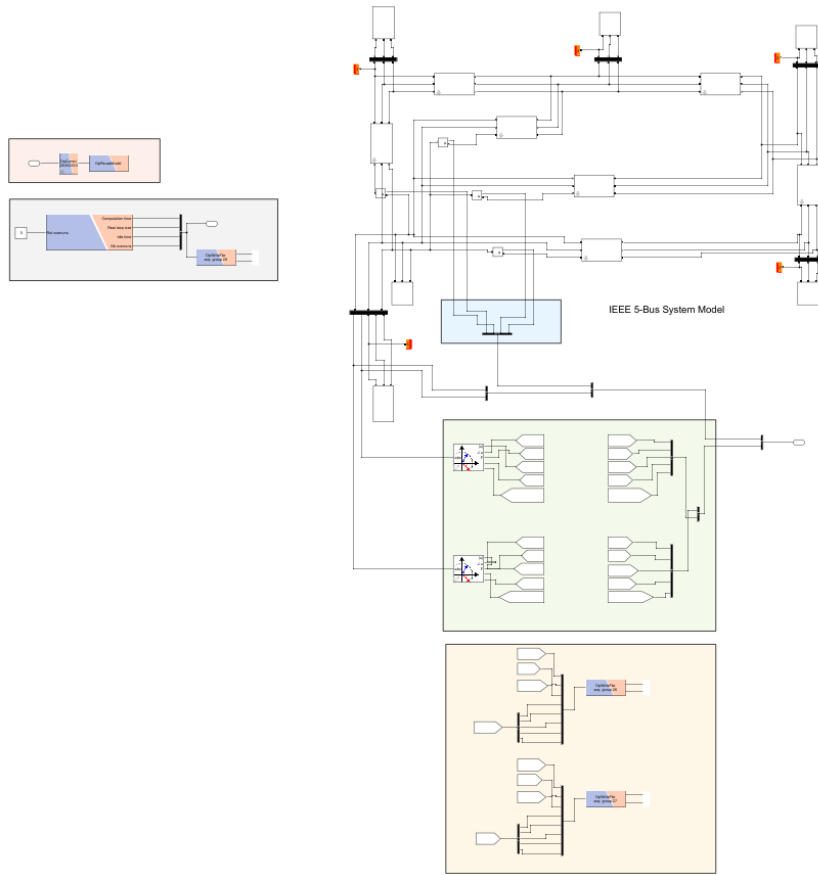


Figure 76: SM_NET subsystem

The computation subsystem SM_NET is depicted in figure 76. Given its complexity, each of the following figures will analyse a part of it to fully describe its behaviour.

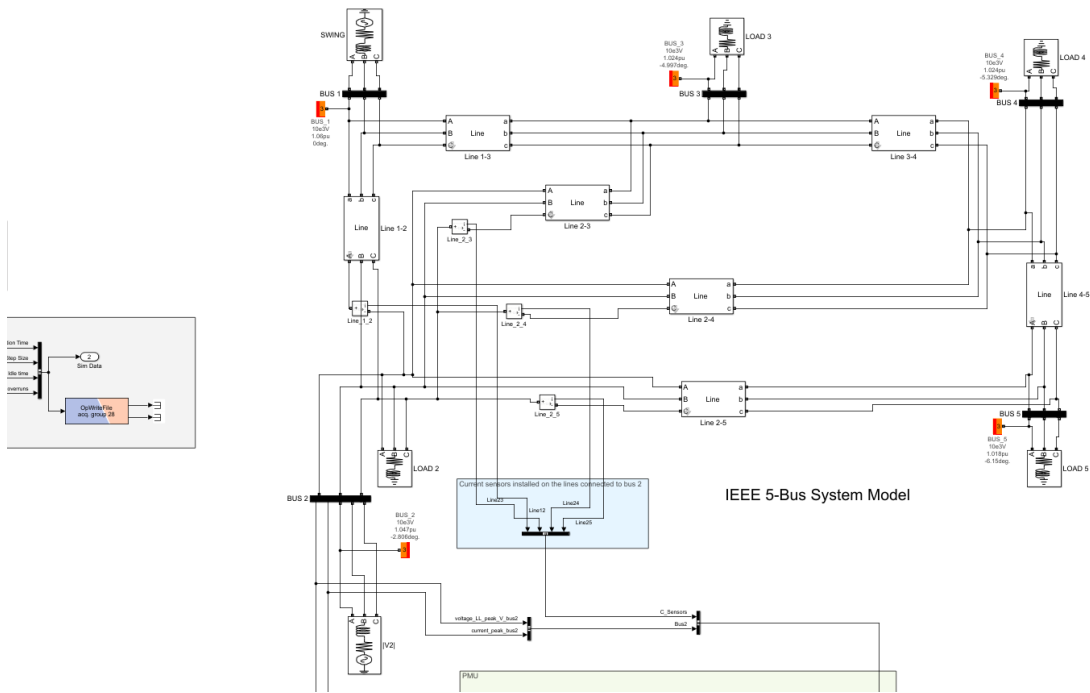


Figure 77: IEEE 5-bus network in SM_NET

The IEEE 5-bus network is kept the same as in the Simulink code, as it does not need any modifications for its implementation in OPAL RT, as illustrated in figure 77.

The most important adjustments in this subsystem, though, consist in the removal of all displays and scopes, since they will not be visible for the user unless present in the GUI, represented by the SC_SCOPE block.

Moving on to the next figure, the light green area of figure 78 comprises both the voltage and the current PMUs. In this figure, the detected quantities of voltage and current synchrophasors, together with their accuracy indices and ideal signal comparisons are sent to a bus creator and to an output port “Bus_Data” so that they will be accessible in the GUI.

Ideal sensor data detected by the four current sensors is sent to the GUI as well.

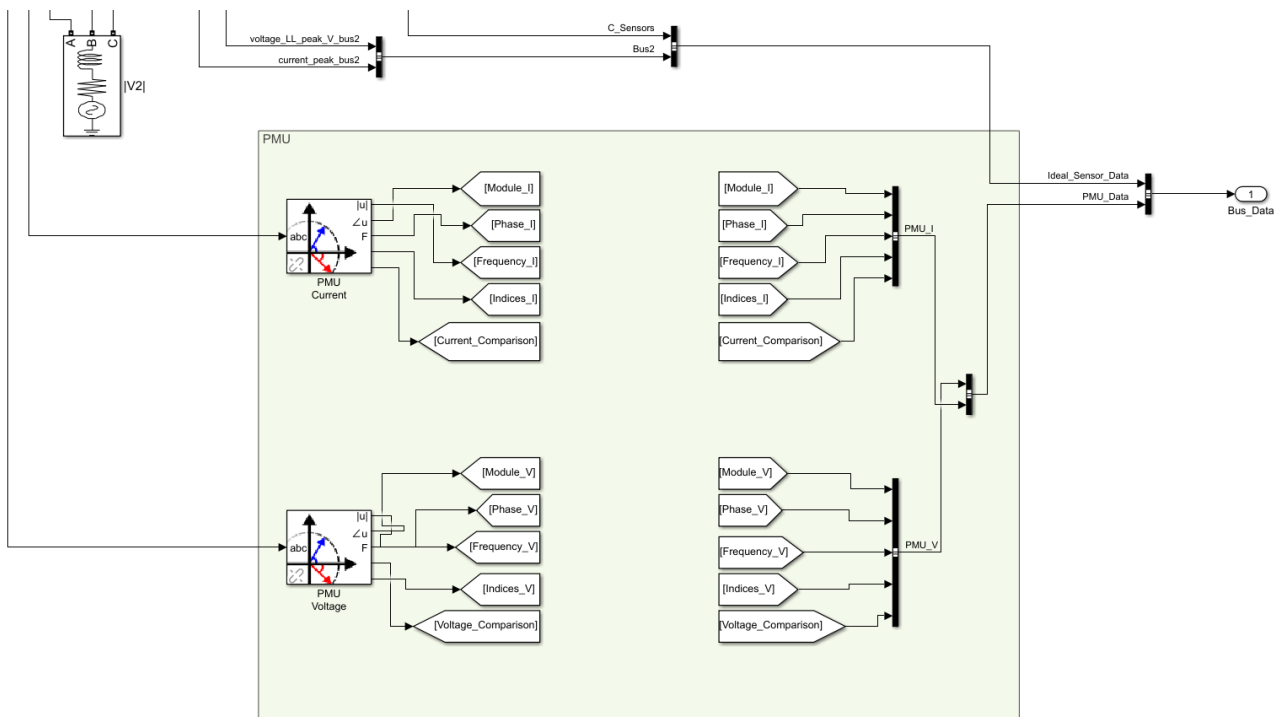


Figure 78: PMUs blocks in the SM_NET subsystem

In the yellow area of figure 79 OpWriteFile blocks are used. These blocks, present in RT-LAB libraries in Simulink, allow one to store the real-time simulation data in a .mat file which could be used for a later analysis and elaboration of the results.

In order to analyse the same variables as in Simulink, module, phase, frequency and all the five indices for both voltage and current PMUs are acquired.

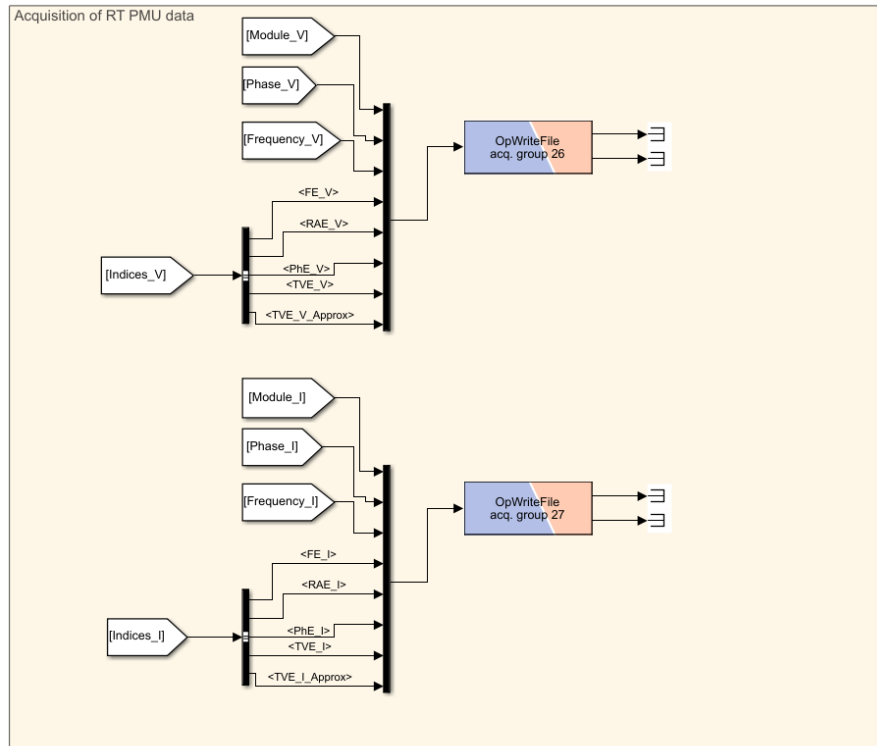


Figure 79: OpWriteFile blocks for voltage and current PMUs in SM_NET

The parameters of OpWriteFile blocks must be properly chosen in order to correctly acquire the needed information. As an example, the OpWriteFile mask for the voltage PMUs is shown in figure 80.

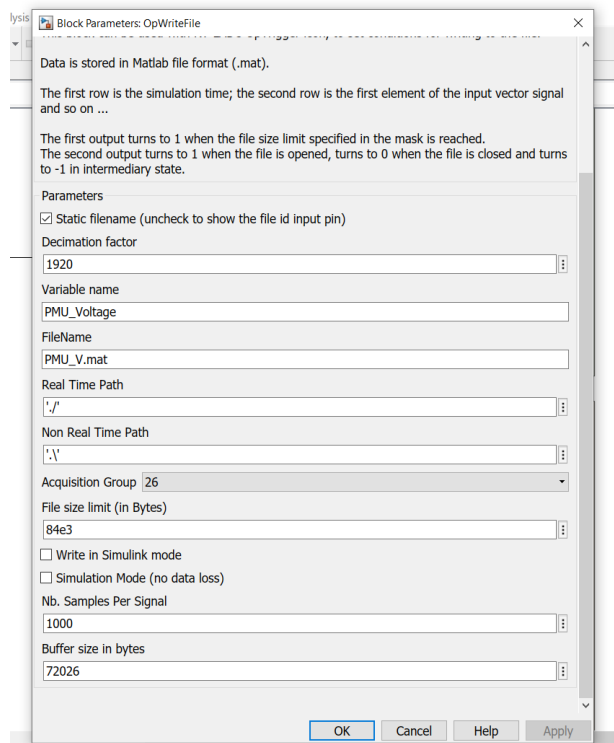


Figure 80: OpWriteFile parameters mask in SM_NET

The most important parameters to be correctly set in this mask are the buffer size, the decimation factor and the file size.

The buffer size in [bytes] must be bigger or equal than the result of equation (28):

$$\text{Buffer size} \geq (\# \text{ logged signals} + 1) \cdot (\# \text{ samples}) \cdot 8 \quad (28)$$

In this case study, the number of logged signals is equal to 8 and the number of samples is considered equal to 1000, guaranteeing that there will not be data loss every group of 1000 samples. The number of logged signals is increased by 1, due to the fact that the first row will be automatically added by RT-LAB and it will contain the simulation time.

According to (28), the minimum buffer size is equal to 72026.

The decimation factor is a coefficient that, if equal to 1, allows one to acquire in the .mat file all the samples at the exact time step of the real-time simulation, which would be equal to 52.08 μs in this case.

Since the PMUs reporting rates are equal to 10 frames/s, it is reasonable to store the real-time data at the same frequency, representing what will happen in reality. Therefore:

$$\frac{0.1 \text{ s/frame}}{52.08 \cdot 10^{-6} \text{ s}} = 1920 = \text{decimation factor}$$

This ratio represents the decimation factor. Indeed, choosing 1920 allows one to store only the necessary information.

The file size in [bytes] is a parameter which must be chosen properly as its default value is equal to zero. The equation (29) serves to this purpose:

$$\text{File size} = 1.5 \cdot (\# \text{ logged signals} + 1) \cdot 8 \cdot \frac{\text{logging duration}}{\text{decimation factor} \cdot \text{sample time [s]}} \quad (29)$$

Considering a logging duration of 60 seconds, as all the other parameters are known, allows one to conclude that a file of at least 65 kb must be set.

Finally, one could get an idea of the writing time interval in [seconds] which will be stored according to equation (30):

$$\text{Writing time interval} = \# \text{ samples} \cdot \text{sample time} \cdot \text{decimation factor} \quad (30)$$

Introducing all data in equation (30), permits one to conclude that a time interval of 99.9 seconds will be acquired and stored at time steps of 0.1 seconds.

All these considerations are evidently valid for both voltage and current PMU data.

The last part of the SM_NET subsystem to be discussed is illustrated in figure 81.

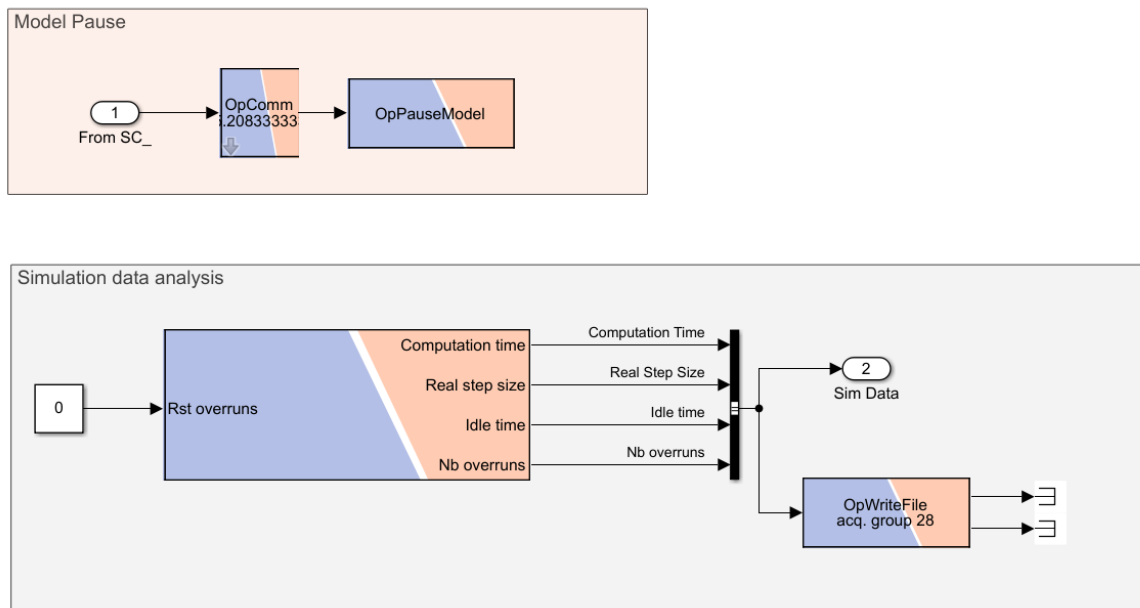


Figure 81: Model pause and simulation data in SM_NET

In the orange area, the OpPauseModel block is used. This block can be used in a Master or Slave subsystem to pause the model, whenever the input signal passes from a value less than 0.5 to a greater one. The model is paused on the next step exchange between subsystems.

The command needed to pause the model comes from the GUI in the SC_SCOPE subsystem.

In the grey area, instead, the OpMonitor block is placed. This RT-LAB block allows one to retrieve information about the subsystem where it is inserted.

As outputs, the following parameters have been selected:

1. *Computation time*: it specifies the time spent in calculations during the execution of the previous time step.
2. *Real step size*: it specifies the total length of execution of the previous time step, including the communication time.
3. *Idle time*: it specifies the value of the idle time during the execution of the previous time step.
4. *Number of overruns*: it reports the number of overruns detected during the execution of the subsystem where the OpMonitor block is inserted.

It is worth saying that the three time quantities are in the same unit as the chosen time step, therefore they will be expressed in [μ s].

All the simulation data is also written in its own .mat file so that a further analysis can be carried out on these values in paragraph 4.5.4.

The GUI, instead, presents itself as in figure 82 and it is in the SC_SCOPE subsystem.

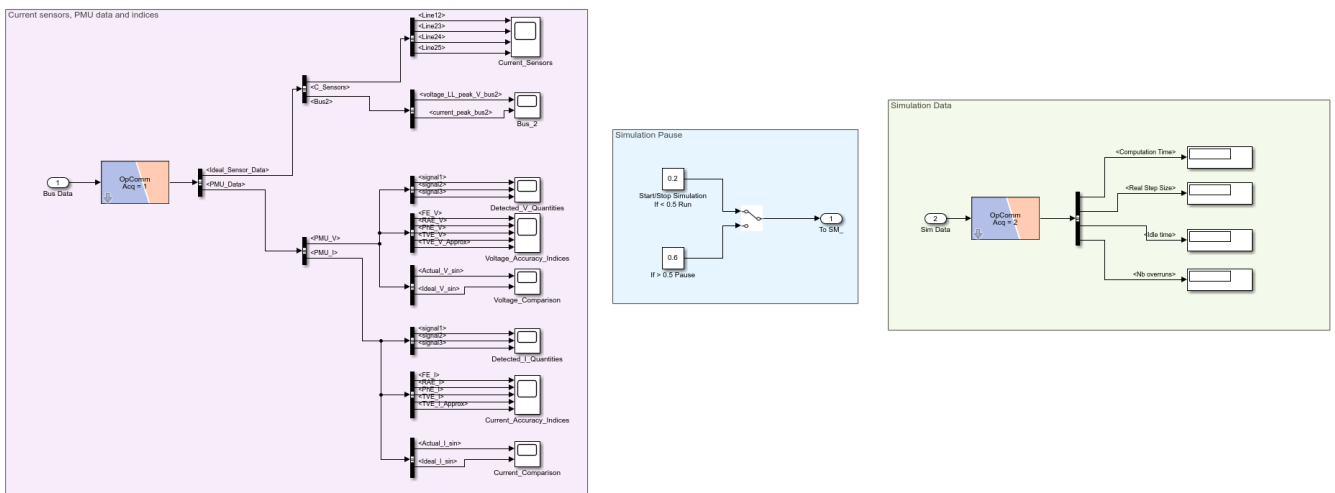


Figure 82: SC_SCOPE subsystem

In the pink area, all scopes are placed. The first bus allows one to retrieve data from the ideal sensors, as one could visualize the four currents incident to bus 2 as well as voltage and current at the same bus.

PMU data coming from both the voltage and current blocks are shown in scopes, together with accuracy indices and a visual comparison with respect to the ideal sinusoids.

In the light blue area, the simulation pause command can be sent to the SM_NET subsystem by means of a simple switch.

Finally, in the light green area, simulation data concerning computation and idle time, real step size and the number of overruns is visualized with displays.

4.5.3 Analysis of the real-time data

After having correctly compiled the programme, load it in OPAL-RT 4510 simulator and execute it, one could visualize the GUI as depicted in figure 83.

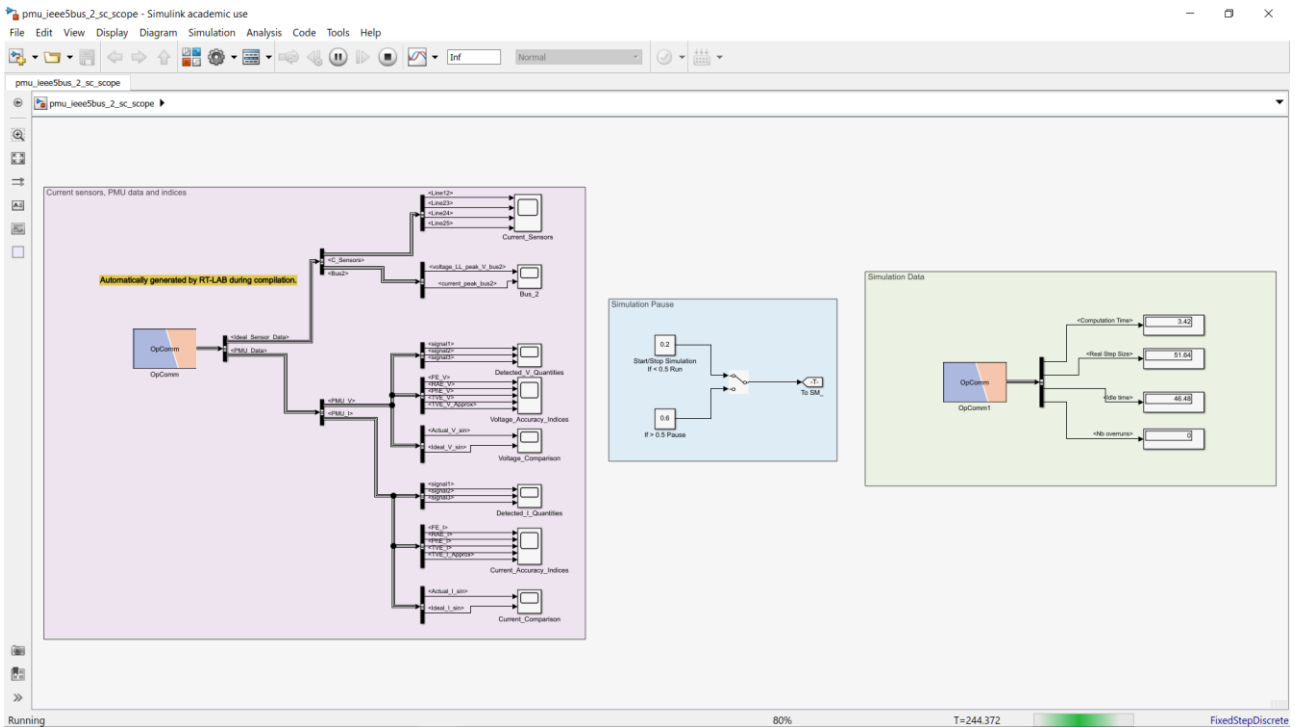


Figure 83: Programme running in RT on OPAL simulator

Notwithstanding that the number of overruns shown in the display of figure 83 provides one with an output equal to zero, one could exploit the monitoring real-time functionalities to validate this information and obtain more of them about the model.

Indeed, while the code is running in real-time, in RT-LAB one can make sure that no overruns occur if all subsystems, in this case the SM_NET subsystem, have their CPU usage below 100 %.

As illustrated in figure 84, the usage of the CPU the whole Master subsystem is around 10 % which is well below the 100 % threshold and for most of the time step the system is idling (87.48 %).

Overview Development Execution Variables Files Assignment Diagnostic Hardware Simulation Tools					
Display Properties Compilation Console Variables Table Variable Viewer Monitoring Log Viewer					
Model: PMU_IEEE5Bus Ts=5.20833333333E-5[s] T=13.71495[s] Number of overruns=0					
Probes	Info				
	Usage [%]	Min	Max	Mean	
PMU_IEEE5Bu...3333E-5[s]	10.4%				
SM_NET Ts...9932E-5[s]	10.4%	dt= 4.02 [us]	dt= 6.90 [us]	dt= 5.42 [us]	
New data acquisition	0.09%	dt= 0.04 [us]	dt= 0.10 [us]	dt= 0.05 [us]	
Major comp...tion time	3.77%	dt= 0.67 [us]	dt= 3.31 [us]	dt= 1.97 [us]	
Minor comp...tion time	5.19%	dt= 2.63 [us]	dt= 2.81 [us]	dt= 2.70 [us]	
Execution cycle	10.4%	dt= 4.02 [us]	dt= 6.90 [us]	dt= 5.42 [us]	
Total step size	99.87%	dt= 42.25 [us]	dt= 61.66 [us]	dt= 52.02 [us]	
Total idle	87.48%	dt= 37.10 [us]	dt= 53.87 [us]	dt= 45.56 [us]	

Figure 84: Real-time overruns monitoring

Pausing the model before the next time step, allows one to visualize data from the previous time step from the GUI.

The following section will be dedicated to the analysis of the graphs from the SC_SCOPE subsystems and a preliminary analysis will be carried out on them, as done for the Simulink simulation.

In figure 85, the four currents measured by the ideal current sensors are represented. Their values are measured by means of the scope measurement tool and the results are summarized in table 11.

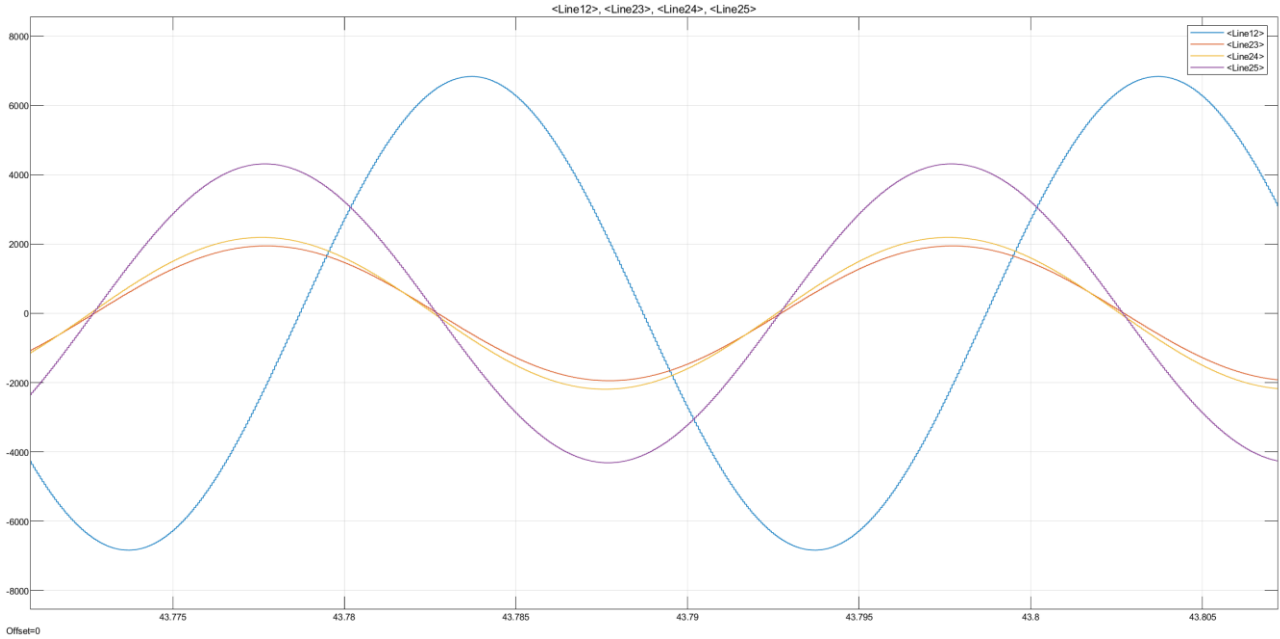


Figure 85: Currents measured in the lines incident to bus 2 in OPAL-RT

Table 11: Current sensors data in OPAL-RT

Line Number	Max Value [A]	RMS [A]
1-2	$6.834 \cdot 10^3$	$4.793 \cdot 10^3$
2-3	$1.945 \cdot 10^3$	$1.370 \cdot 10^3$
2-4	$2.190 \cdot 10^3$	$1.541 \cdot 10^3$
2-5	$4.312 \cdot 10^3$	$3.036 \cdot 10^3$

It is clear that the current data in table 11 is comparable with the one obtained performing the simulation in Simulink and summarized in table 10.

The line-to-line voltage and phase current at bus 2 are shown in figure 86. A preliminary measurement allows one to get the data shown in table 12, which is reasonably comparable to the one obtained in Simulink.

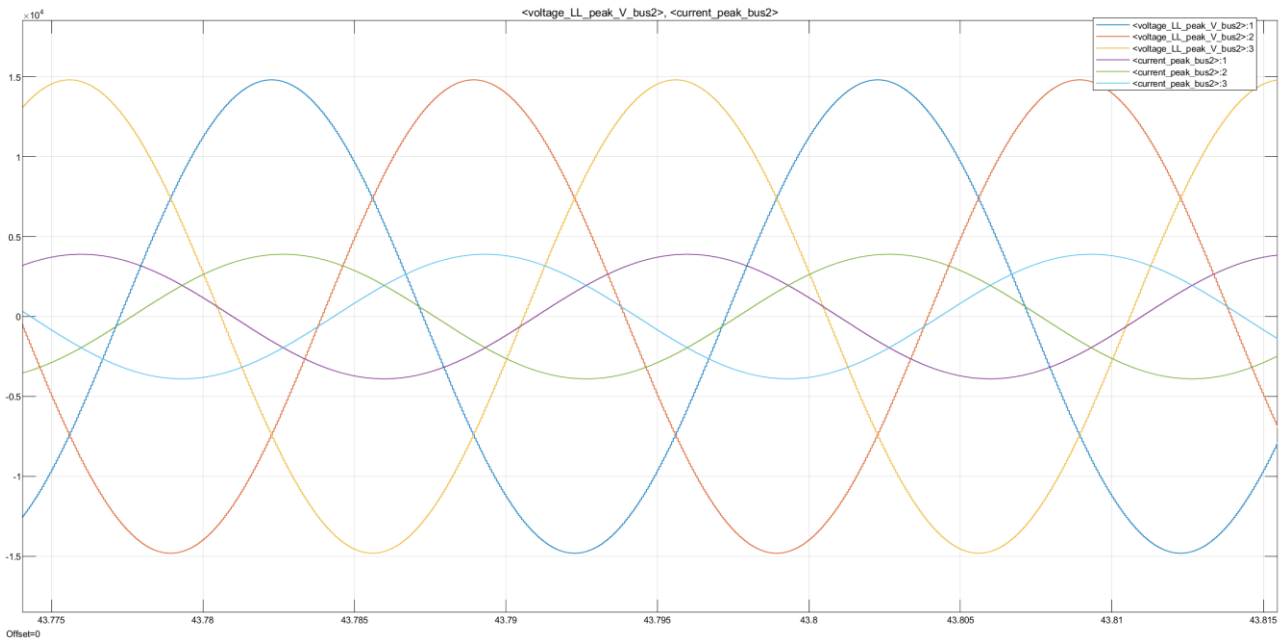


Figure 86: Voltage and current at bus 2 in OPAL-RT

Table 12: Voltage and current at bus 2 in OPAL-RT

Line Number	Max Value [V]; [A]	RMS [V]; [A]
V _{AB} Bus 2	$1.481 \cdot 10^4$	$1.052 \cdot 10^4$
I _A Bus 2	$3.897 \cdot 10^3$	$2.793 \cdot 10^3$

The following three graphs illustrate the outputs of the voltage PMU.

In figure 87, the detected amplitude, phase in [degrees] and frequency of the voltage synchrophasor are plotted.

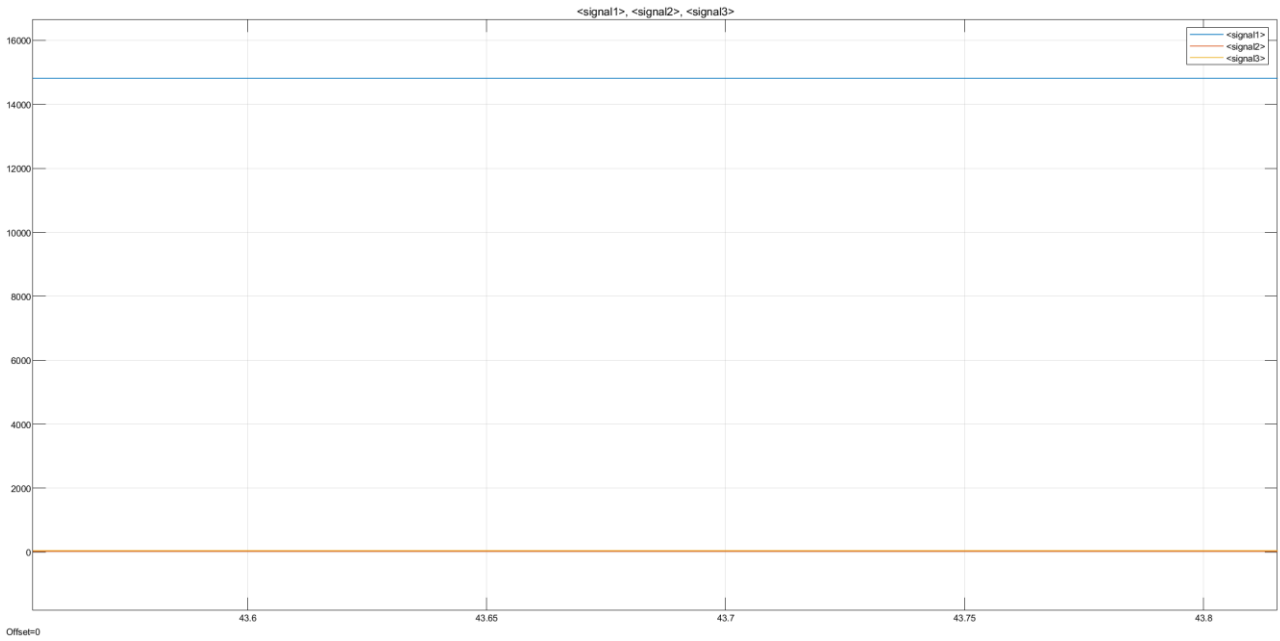


Figure 87: Amplitude (blue), phase (red) and frequency (yellow) of voltage synchrophasor in OPAL-RT

In figure 88, the voltage accuracy indices are represented. This time they have been calculated in real-time with respect to the previous Simulink simulation.

Taking a preliminary look at them, one could notice that they are all very small and they would provide the compliance of the PMU. Nonetheless, the approximate TVE is the biggest one as it is roughly equal to $3.5 \cdot 10^{-4}$, it is anyhow smaller than 1 %.

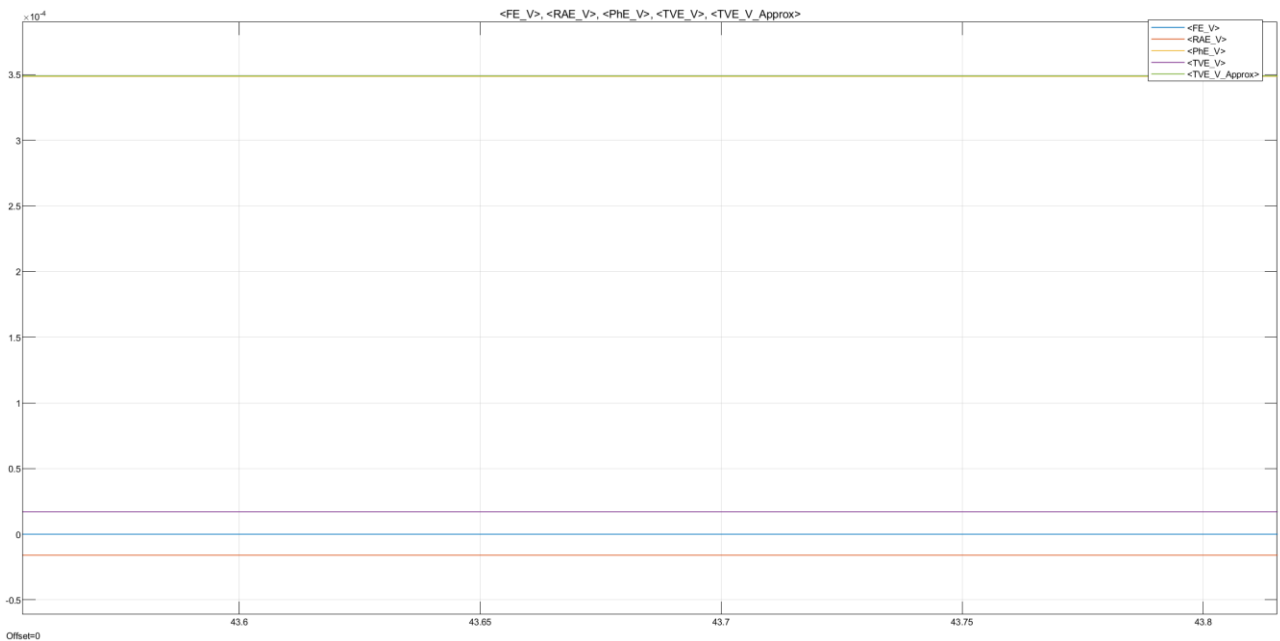


Figure 88: Accuracy indices of voltage PMU in OPAL-RT

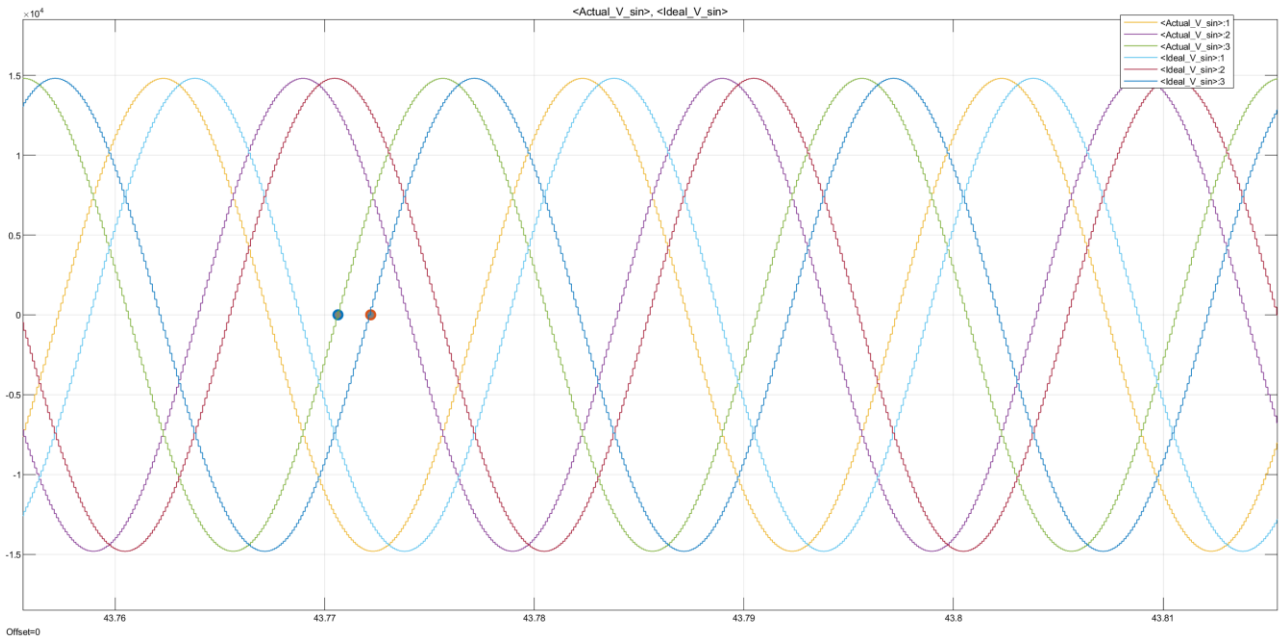


Figure 89: Comparison between ideal and measured voltage sinusoid in OPAL-RT

In figure 89, as done in Simulink, a rough comparison is carried out between the ideal and the measured voltage sinusoid.

Comparing the two signals, one could measure a time shift of approximately 1.564 ms which is only slightly bigger than the 1.458 ms detected in Simulink.

This information provides one with a first idea about the validation of the results.

The three subsequent plots refer to the outputs of the current PMU.

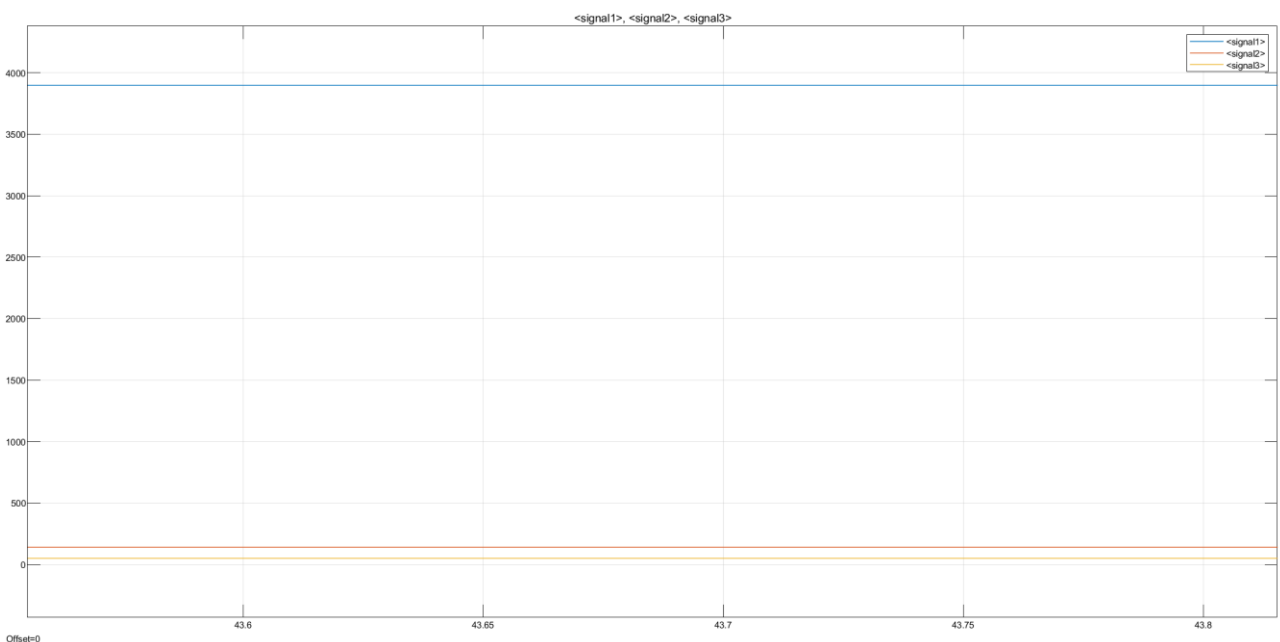


Figure 90: Amplitude (blue), phase (red) and frequency (yellow) of current synchrophasor in OPAL-RT

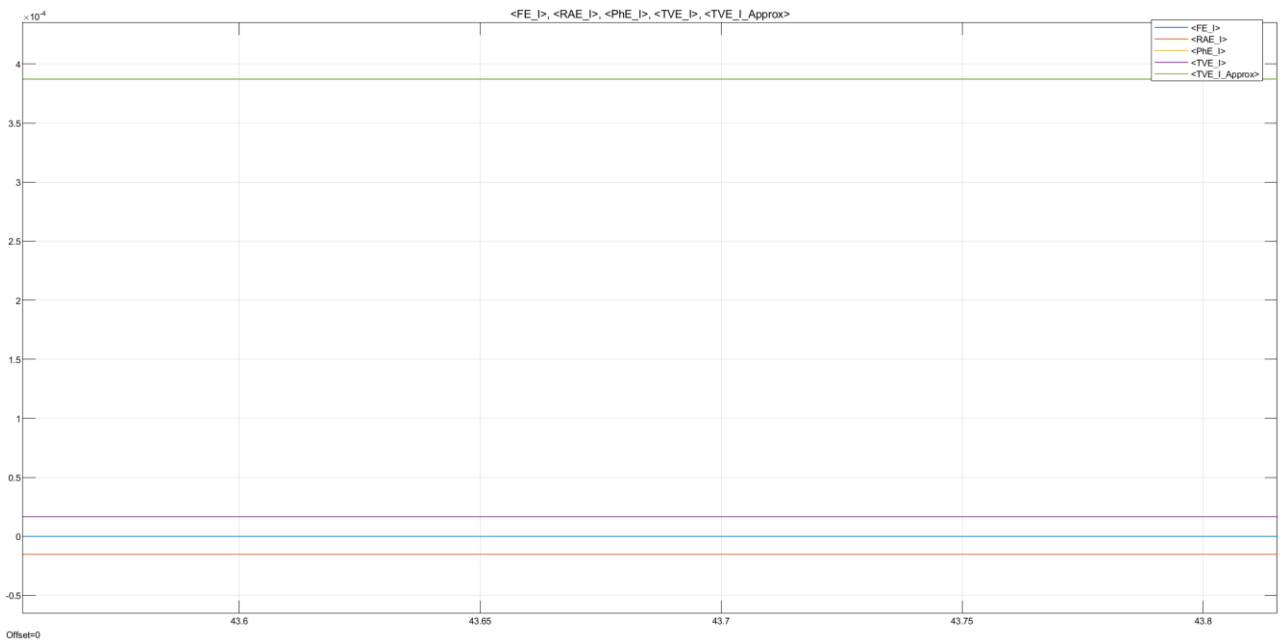


Figure 91: Accuracy indices of current PMU in OPAL-RT

In figure 91, the current accuracy indices are represented. As done for the voltage indices, taking a preliminary look at them, one could notice that they are all very small and they would provide the compliance of the PMU. The approximate TVE is still the biggest one as it is roughly smaller than $4 \cdot 10^{-4}$, but it is anyhow smaller than 1 %.

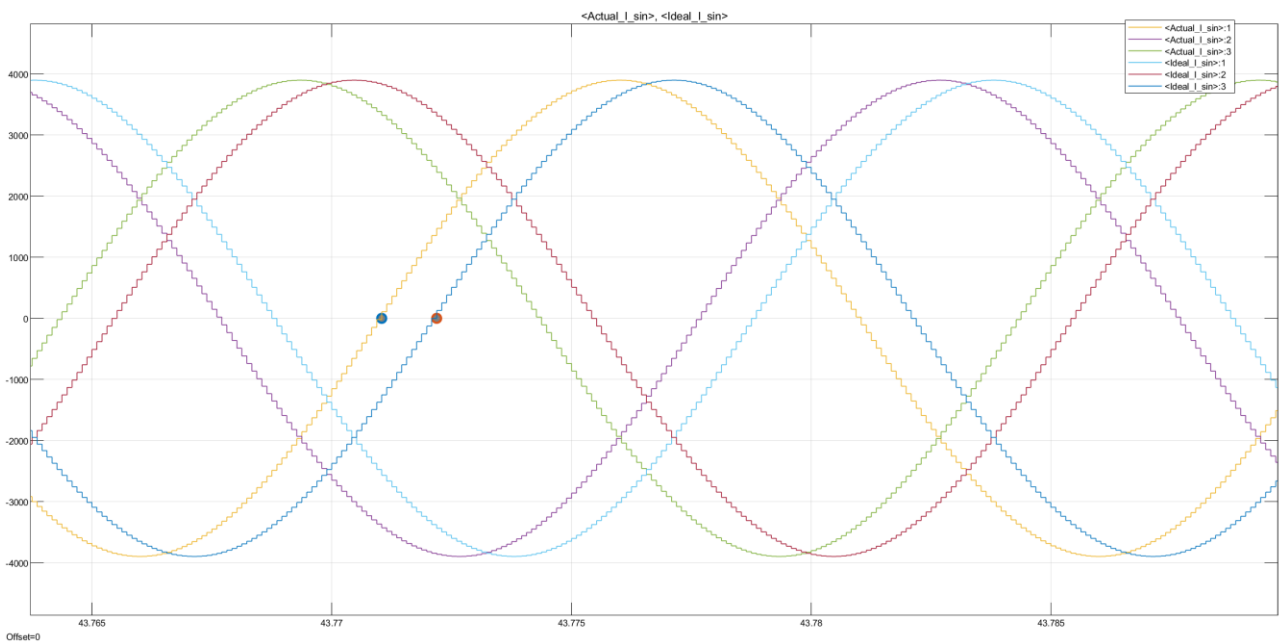


Figure 92: Comparison between ideal and measured current sinusoid in OPAL-RT

In figure 92, as done in Simulink, a rough comparison is carried out between the ideal and the measured current sinusoid.

Comparing the two signals, one could measure a time shift of approximately 1.147 ms which is only slightly bigger than the 1.146 ms detected in Simulink.

This information, again, implicates a clear validation of the results.

4.5.4 Final results and considerations

In this last paragraph, a final elaboration is performed on the attained outcomes. To serve this purpose, a Matlab script has been written in such a way that both Simulink simulation data and OPAL RT-4510 simulation data are analysed and compared at the same time.

First of all, Simulink data is considered. The earliest thing to do is to remove the first element from the arrays as it is equal to zero in all cases.

This is followed by the evaluation of FE, RAE, PhE, TVE and approximate TVE indices every second for the 5-second-long simulation. Since 10 frames are reported per each second, the average value and standard deviation is computed every 10 samples.

In this way, a vector containing 5 means and standard deviations has been created for each index. This has been done although the first element corresponding to the average value of the first second of sampled data must be discarded in all cases, as it corresponds to the interval of time 0 – 1 s during which the PLL was still oscillating and measurements were set to 0.

Finally, a Matlab structure called “PMU” has been created in such a way that it contains all the voltage and current accuracy indices data.

In the structure, just the last value of the array, thus the one corresponding to the 4 to 5 s time interval is stored. This has been done so that the steady-state regime was guaranteed in the measurands.

All this procedure is repeated for both the voltage and current data retrieved from Simulink simulations and the corresponding code is shown in figures 93 and 94.

```

Final_Elab.m x Datim x Accuracy.m x +
1  %%%% PMU Data Elab %%%%
2
3  %%% PMU Voltage %%%
4  %deleting the sample at 0 s from the vector
5 - FE_V(1) = [];
6 - RAE_V(1) = [];
7 - PhE_V(1) = [];
8 - TVE_V(1) = [];
9 - TVE_V_approx(1) = [];
10
11 %Given a reporting rate equal to 10 frames/s, every 1 second a mean + std
12 %value is provided in output
13 - FE_V_data = [mean(reshape(FE_V,10,[])); std(reshape(FE_V, 10,[]))]; %1st row => average, 2nd row => stdev (the first sample must be discarded -> still PLL locking)
14 - RAE_V_data = [mean(reshape(RAE_V,10,[])); std(reshape(RAE_V, 10,[]))];
15 - PhE_V_data = [mean(reshape(PhE_V,10,[])); std(reshape(PhE_V, 10,[]))];
16 - TVE_V_data = [mean(reshape(TVE_V,10,[])); std(reshape(TVE_V, 10,[]))];
17 - TVE_V_approx_data = [mean(reshape(TVE_V_approx,10,[])); std(reshape(TVE_V_approx, 10,[]))];
18
19 % In the Matlab structure we put just the last value of the 5-second
20 % simulation (st.state for sure)
21 - PMU(1).Entity = 'Voltage Indices - Simulink';
22 - PMU(1).FE = FE_V_data(:,5);
23 - PMU(1).RAE = RAE_V_data(:,5);
24 - PMU(1).PhE = PhE_V_data(:,5);
25 - PMU(1).TVE = TVE_V_data(:,5);
26 - PMU(1).TVE_approx = TVE_V_approx_data(:,5);

```

Figure 93: Final elaboration of Simulink for voltage data

```

28
29 %%% PMU Current %%%
30 %deleting the sample at 0 s from the vector
31 - FE_I(1) = [];
32 - RAE_I(1) = [];
33 - PhE_I(1) = [];
34 - TVE_I(1) = [];
35 - TVE_I_approx(1) = [];
36
37 %Given a reporting rate equal to 10 frames/s, every 1 second a mean + std
38 %value is provided in output
39 - FE_I_data = [mean(reshape(FE_I,10,[])); std(reshape(FE_I, 10,[]))]; %1st row => average, 2nd row => stdev (the first sample must be discarded -> still PLL locking)
40 - RAE_I_data = [mean(reshape(RAE_I,10,[])); std(reshape(RAE_I, 10,[]))];
41 - PhE_I_data = [mean(reshape(PhE_I,10,[])); std(reshape(PhE_I, 10,[]))];
42 - TVE_I_data = [mean(reshape(TVE_I,10,[])); std(reshape(TVE_I, 10,[]))];
43 - TVE_I_approx_data = [mean(reshape(TVE_I_approx,10,[])); std(reshape(TVE_I_approx, 10,[]))];
44
45 - PMU(3).Entity = 'Current Indices - Simulink';
46 - PMU(3).FE = FE_I_data(:,5);
47 - PMU(3).RAE = RAE_I_data(:,5);
48 - PMU(3).PhE = PhE_I_data(:,5);
49 - PMU(3).TVE = TVE_I_data(:,5);
50 - PMU(3).TVE_approx = TVE_I_approx_data(:,5);

```

Figure 94: Final elaboration of Simulink for current data

Having evaluated Simulink results, it is time to deal with OPAL real-time data which have been saved in .mat files.

Firstly, the three .mat files have to be loaded in the Workspace so that they can be manipulated.

In the case of real-time data, the writing time interval is not equal to 5 seconds as in Simulink, but it is equal to 99.9 seconds as calculated in paragraph 4.5.2. Therefore, only the first 50 samples corresponding the first 5 seconds of simulation must be extracted in order to validate the model over the same time interval.

After this part, the average values and standard deviations are computed every second considering a 10-sample interval, as done for the Simulink data.

The last value of the array is put in the same PMU structure so that an easier comparison can be performed in Matlab.

The code corresponding to the described procedure is shown in figures 95 and 96.

```

Final_Elab.m x  Datim x  Accuracy.m x  +
50 - PMU(3).TVE_approx = TVE_I_approx_data(:,5);
51
52   %%% Loading OPAL Data %%%
53 - load('PMU_V.mat');
54 - load('PMU_C.mat');
55 - load('SimData.mat');
56
57   %%% PMU Voltage %%%
58   %Extracting only the first 5 seconds of the RT simulation
59 - FE_V_O = PMU_Voltage(5, 2:51);
60 - RAE_V_O = PMU_Voltage(6, 2:51);
61 - PhE_V_O = PMU_Voltage(7, 2:51);
62 - TVE_V_O = PMU_Voltage(8, 2:51);
63 - TVE_V_approx_O = PMU_Voltage(9, 2:51);
64
65   %Given a reporting rate equal to 10 frames/s, every 1 second a mean + std
66   %value is provided in output
67 - FE_V_O_data = [mean(reshape(FE_V_O,10,[])); std(reshape(FE_V_O, 10,[]))]; %1st row => average, 2nd row => stdev (the first sample must be discarded -> still PLL locking)
68 - RAE_V_O_data = [mean(reshape(RAE_V_O,10,[])); std(reshape(RAE_V_O, 10,[]))];
69 - PhE_V_O_data = [mean(reshape(PhE_V_O,10,[])); std(reshape(PhE_V_O, 10,[]))];
70 - TVE_V_O_data = [mean(reshape(TVE_V_O,10,[])); std(reshape(TVE_V_O, 10,[]))];
71 - TVE_V_approx_O_data = [mean(reshape(TVE_V_approx_O,10,[])); std(reshape(TVE_V_approx_O, 10,[]))];
72
73 - PMU(2).Entity = 'Voltage Indices - OPAL';
74 - PMU(2).FE = FE_V_O_data(:,5);
75 - PMU(2).RAE = RAE_V_O_data(:,5);
76 - PMU(2).PhE = PhE_V_O_data(:,5);
77 - PMU(2).TVE = TVE_V_O_data(:,5);
78 - PMU(2).TVE_approx = TVE_V_approx_O_data(:,5);
79

```

Figure 95: Final elaboration of OPAL-RT voltage data

```

Final_Elab.m x  Datim x  Accuracy.m x  +
77 - PMU(2).TVE = TVE_V_O_data(:,5);
78 - PMU(2).TVE_approx = TVE_V_approx_O_data(:,5);
79
80   %%% PMU Current %%%
81   %Extracting only the first 5 seconds of the RT simulation
82 - FE_I_O = PMU_Current(5, 2:51);
83 - RAE_I_O = PMU_Current(6, 2:51);
84 - PhE_I_O = PMU_Current(7, 2:51);
85 - TVE_I_O = PMU_Current(8, 2:51);
86 - TVE_I_approx_O = PMU_Current(9, 2:51);
87
88   %Given a reporting rate equal to 10 frames/s, every 1 second a mean + std
89   %value is provided in output
90 - FE_I_O_data = [mean(reshape(FE_I_O,10,[])); std(reshape(FE_I_O, 10,[]))]; %1st row => average, 2nd row => stdev (the first sample must be discarded -> still PLL locking)
91 - RAE_I_O_data = [mean(reshape(RAE_I_O,10,[])); std(reshape(RAE_I_O, 10,[]))];
92 - PhE_I_O_data = [mean(reshape(PhE_I_O,10,[])); std(reshape(PhE_I_O, 10,[]))];
93 - TVE_I_O_data = [mean(reshape(TVE_I_O,10,[])); std(reshape(TVE_I_O, 10,[]))];
94 - TVE_I_approx_O_data = [mean(reshape(TVE_I_approx_O,10,[])); std(reshape(TVE_I_approx_O, 10,[]))];
95
96 - PMU(4).Entity = 'Current Indices - OPAL';
97 - PMU(4).FE = FE_I_O_data(:,5);
98 - PMU(4).RAE = RAE_I_O_data(:,5);
99 - PMU(4).PhE = PhE_I_O_data(:,5);
100 - PMU(4).TVE = TVE_I_O_data(:,5);
101 - PMU(4).TVE_approx = TVE_I_approx_O_data(:,5);

```

Figure 96: Final elaboration of OPAL-RT current data

The final PMU structure summarizing all the results is depicted in figure 97.

Fields	Entity	FE	RAE	PhE	TVE	TVE_approx
1	'Voltage Indices - Simulink'	[3.1619e-12;0]	[-1.5966e-05;4.8953e-12]	[3.3456e-04;3.5071e-10]	[1.7000e-05;2.9989e-12]	[3.3494e-04;3.5011e-10]
2	'Voltage Indices - OPAL'	[3.1949e-10;0]	[-1.5966e-05;4.8947e-12]	[3.3508e-04;3.4878e-08]	[1.7003e-05;2.1331e-10]	[3.3546e-04;3.4839e-08]
3	'Current Indices - Simulink'	[3.1193e-12;0]	[-1.5192e-05;1.1069e-16]	[3.7295e-04;3.5071e-10]	[1.6528e-05;2.4107e-12]	[3.7326e-04;3.5042e-10]
4	'Current Indices - OPAL'	[3.1967e-10;0]	[-1.5192e-05;8.5702e-16]	[3.7347e-04;3.4879e-08]	[1.6531e-05;2.4002e-10]	[3.7378e-04;3.4850e-08]

Figure 97: PMU structure containing Simulink and OPAL-RT accuracy indices results

As clearly noticeable in figure 97, both the voltage and current indices simulated in Simulink are validated in the real-time simulation performed using OPAL RT-4510.

The attained results are visibly comparable and, it is worth noting, that the frequency error has a such small standard deviation that it is smaller than the machine epsilon of Matlab.

These results are sufficient to affirm that the synchrophasors simulated in Simulink are equal to the ones acquired in real-time. Anyhow, for the sake of completeness, the code has been completed including also the synchrophasors analysis.

The exact same procedure described for the accuracy indices is performed again, for both voltage and current data, and repeated for both Simulink and OPAL-RT results (figures 98 and 99).

```

Final_Elab.m x Datim x Accuracy.m x +
101 - PMU(4).TVE_approx = TVE_I_approx_O_data(:,5);
102
103
104   %%% Voltage and Current Phasors from Simulink %%%
105 - Mag_V(1) = [];
106 - Ph_V(1) = [];
107
108 - Mag_V_data = [mean(reshape(Mag_V,10,[])); std(reshape(Mag_V, 10,[]))];
109 - Ph_V_data = [mean(reshape(Ph_V,10,[])); std(reshape(Ph_V, 10,[]))];
110
111 - Phasors(1).Entity = 'Voltage Phasor - Simulink';
112 - Phasors(1).Magnitude = Mag_V_data(:,5);
113 - Phasors(1).Phase = Ph_V_data(:,5);
114
115   % Current
116 - Mag_I(1) = [];
117 - Ph_I(1) = [];
118
119 - Mag_I_data = [mean(reshape(Mag_I,10,[])); std(reshape(Mag_I, 10,[]))];
120 - Ph_I_data = [mean(reshape(Ph_I,10,[])); std(reshape(Ph_I, 10,[]))];
121
122 - Phasors(3).Entity = 'Current Phasor - Simulink';
123 - Phasors(3).Magnitude = Mag_I_data(:,5);
124 - Phasors(3).Phase = Ph_I_data(:,5);
125
126   %%% Voltage and Current Phasors from OPAL %%%
127 - Mag_V_O = PMU_Voltage(2, 2:51);
128 - Ph_V_O = PMU_Voltage(3, 2:51);
129
130 - Mag_V_O_data = [mean(reshape(Mag_V_O,10,[])); std(reshape(Mag_V_O, 10,[]))];
131 - Ph_V_O_data = [mean(reshape(Ph_V_O,10,[])); std(reshape(Ph_V_O, 10,[]))];
132
133 - Phasors(2).Entity = 'Voltage Phasor - OPAL';
134 - Phasors(2).Magnitude = Mag_V_O_data(:,5);
135 - Phasors(2).Phase = Ph_V_O_data(:,5);
136
137
138
139
140
141
142
143
144
145
146

```

Figure 98: Final elaboration of Simulink synchrophasors

```

Final_Elab.m x Datim x Accuracy.m x +
112 - Phasors(1).Magnitude = Mag_V_data(:,5);
113 - Phasors(1).Phase = Ph_V_data(:,5);
114
115   % Current
116 - Mag_I(1) = [];
117 - Ph_I(1) = [];
118
119 - Mag_I_data = [mean(reshape(Mag_I,10,[])); std(reshape(Mag_I, 10,[]))];
120 - Ph_I_data = [mean(reshape(Ph_I,10,[])); std(reshape(Ph_I, 10,[]))];
121
122 - Phasors(3).Entity = 'Current Phasor - Simulink';
123 - Phasors(3).Magnitude = Mag_I_data(:,5);
124 - Phasors(3).Phase = Ph_I_data(:,5);
125
126   %%% Voltage and Current Phasors from OPAL %%%
127 - Mag_V_O = PMU_Voltage(2, 2:51);
128 - Ph_V_O = PMU_Voltage(3, 2:51);
129
130 - Mag_V_O_data = [mean(reshape(Mag_V_O,10,[])); std(reshape(Mag_V_O, 10,[]))];
131 - Ph_V_O_data = [mean(reshape(Ph_V_O,10,[])); std(reshape(Ph_V_O, 10,[]))];
132
133 - Phasors(2).Entity = 'Voltage Phasor - OPAL';
134 - Phasors(2).Magnitude = Mag_V_O_data(:,5);
135 - Phasors(2).Phase = Ph_V_O_data(:,5);
136
137   % Current from OPAL
138 - Mag_I_O = PMU_Current(2, 2:51);
139 - Ph_I_O = PMU_Current(3, 2:51);
140
141 - Mag_I_O_data = [mean(reshape(Mag_I_O,10,[])); std(reshape(Mag_I_O, 10,[]))];
142 - Ph_I_O_data = [mean(reshape(Ph_I_O,10,[])); std(reshape(Ph_I_O, 10,[]))];
143
144 - Phasors(4).Entity = 'Current Phasor - OPAL';
145 - Phasors(4).Magnitude = Mag_I_O_data(:,5);
146 - Phasors(4).Phase = Ph_I_O_data(:,5);

```

Figure 99: Final elaboration of OPAL-RT synchrophasors

The achieved results are again stored in Matlab structure for a better comprehensibility and an easier comparison, as in figure 100.

Fields	Entity	Magnitude	Phase
1	'Voltage Phasor - Simulink'	[1.4813e+04;7.2514e-08]	[27.1933;3.5071e-10]
2	'Voltage Phasor - OPAL'	[1.4813e+04;7.2505e-08]	[27.1933;3.4878e-08]
3	'Current Phasor - Simulink'	[3.8975e+03;9.2204e-13]	[140.3236;3.5071e-10]
4	'Current Phasor - OPAL'	[3.8975e+03;3.3417e-12]	[140.3236;3.4879e-08]

Figure 100: Phasors structure containing Simulink and OPAL-RT synchrophasor results

As previously anticipated, these calculations have just been done for the sake of completeness of the code since the validity of the simulation results has already been proven.

Lastly, the simulation data acquired by the OpMonitor block in OPAL-RT is processed as well. Since no overruns are detected, only the first three rows of the SimData.mat file have been acquired and analysed.

The analysis of the results has been limited again to the first 5 seconds of simulation, in order to refer all the outcomes to the same time interval. In this case, the mean and standard deviation have been calculated over the samples acquired in the whole 5-second interval, as shown in figure 101.

```

148     %% Simulation Data Analysis (for just 5 s sim)
149 -   Comp_Time = SimData(2, 2:51);
150 -   Step_Size = SimData(3, 2:51);
151 -   Idle_Time = SimData(4, 2:51);
152
153 -   C_Time_data = [mean(reshape(Comp_Time,50,[])); std(reshape(Comp_Time, 50,[]))];
154 -   Step_s_data = [mean(reshape(Step_Size,50,[])); std(reshape(Step_Size, 50,[]))];
155 -   I_Time_data = [mean(reshape(Idle_Time,50,[])); std(reshape(Idle_Time, 50,[]))];
156
157 -   SimulationInfo(1).Entity = 'Computation Time';
158 -   SimulationInfo(1).MeanStDev = C_Time_data;
159
160 -   SimulationInfo(2).Entity = 'Real Step Size';
161 -   SimulationInfo(2).MeanStDev = Step_s_data;
162
163 -   SimulationInfo(3).Entity = 'Idle time';
164 -   SimulationInfo(3).MeanStDev = I_Time_data;
165

```

Figure 101: Analysis of simulation data from OPAL-RT

The results are stored in a Matlab structure for better comprehensibility. These results are proven to be comparable with the ones visualized in real-time in the GUI and in the Monitoring tab of RT-LAB while the simulation was running (figure 84).

Fields	Entity	Value
1	'Computation Time'	[3.3273;0.0267]
2	'Real Step Size'	[53.1275;6.2717]
3	'Idle time'	[48.1698;6.2681]
4		

Figure 102: SimulationInfo structure containing OPAL-RT simulation data

It is worth underlying that these values are specified in the same unit as the chosen time steps for the discretized real-time simulation, therefore they are all expressed in [μ s].

The small discrepancy between the time-step of the model and the real value of the time-step is always present and it is due to the clock oscillator.

Chapter 5 – Future perspective on PMUs and DT

5.1 Digital Twin concept

In this last chapter of the thesis, a brief overview over the future perspective on PMUs and Digital Twin (DT) is presented.

This is meant to provide an overall idea of a DT as well as the potentialities of PMUs in the future of the power systems. Indeed, in this work only steady-state PMUs performances have been analysed in real-time simulations using OPAL-RT. As one could clearly imagine, their capabilities do not stop there, which is what this chapter intends to present.

The so-called Digital Twin concept has become a more and more used term in the last years as it is assumed to be the next step for the future generation of control centre applications.

A DT is a software-based abstraction of a complex physical system, for instance an electric network, which is connected via a communication link to the real object through a continuous data flow from the real world. DT provides advanced analytics as it can identify problems, such as faults or overloads, before they actually occur [27].

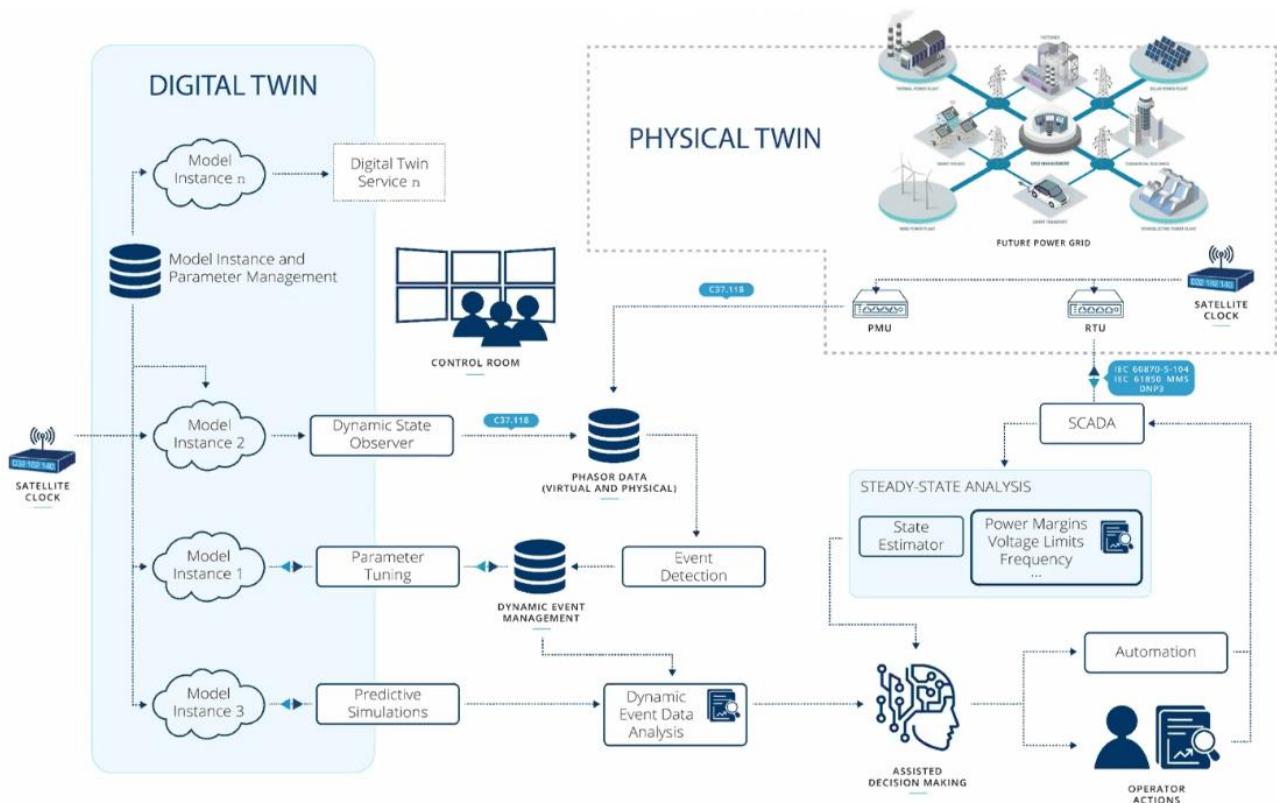


Figure 103: Digital Twin. Adapted from [53]

In figure 103 a clear image of a DT is depicted. More specifically, among the functionalities of a DT one can identify the following:

1. The DT works as a bridge with the real world because it can read data to/from its physically operating counterpart and give an account on an overseeing entity via hardware/software for maintenance, logging, reporting or control.
2. The DT is a self-adaptive model since it is able of adjusting itself in comparison to its physical counterpart in RT or near RT.
3. The DT works as an in-depth observer, owing to its great detail in the description of its components' interactions, unlike physical systems which have a limited number of sensors.
4. The DT can be a powerful predictive tool as it can respond to synthetic stimuli and help in obtaining what-if scenario analysis that cannot be analysed through its physical counterpart due to practical concerns, e.g. fault simulations.

DTs show many fields of implementation among which, in the electric utilities field, one can identify:

1. The possibility to learn about usage/consumption pattern through a combination of logging/reporting and vast amounts of data with the ultimate target of automating Demand Response [53].
2. The opportunity of prediction of a potential future breakdown, enabling the chance of acting before the fault actually occurs and the users become aware [53].

Nevertheless, DTs are not used in the control and monitoring for power systems yet, but the concept of a Dynamic Digital Mirror (DDM) has already been introduced in literature.

A DDM is a dynamic model reflecting the system state in real-time and it is created from the data pool of the asset operator, including a grid model and the current operating data and conditions. The idea of mirroring the network and minimize model errors is to be aware of the latest system state, even in case of non-availability of the actual measurements [27].

In figure 104, one can see the three different parts constituting a DT: the data pool, the RT-Running Instances and the Applications. The tuning of the model parameters is realized via a continuous data exchange from Remote Terminal Units (RTUs) and PMUs installed in field, representing a process layer.

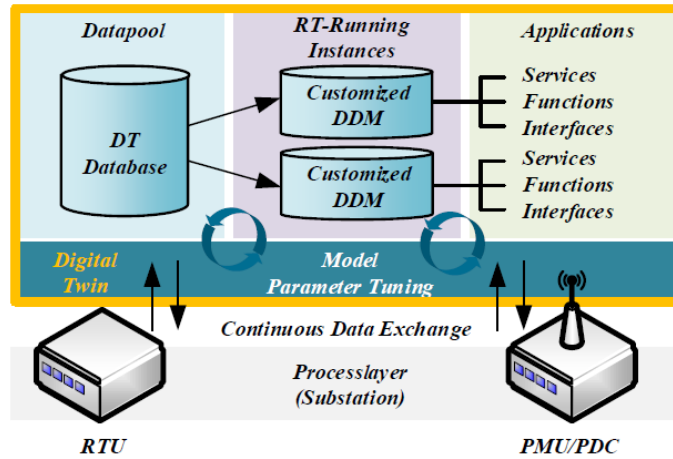


Figure 104: Digital Twin concept for Power System Control Centres. Adapted from [27]

5.2 Future trends for PMUs and DT implementation

Considering the DT concept and the state-of-the-art of PMUs and PQ measurements previously described, there is a number of research projects proposing to investigate on the further developments of these devices by improving both their flexibility/scalability and performances in the grid for the utilities.

As far as flexibility is concerned, the aim of many projects is oriented towards a realization of a faster exchange of data among the devices and a global gathering of information; meaning that starting from in-field sensors which will acquire information that will be collected and transferred via fast communication links (such as fibre optics) to a cloud-level server that will be dedicated for the storage and post-processing of data.

The processing of data, also known as *edge computing*, is focused on bringing the computational parts as close as possible to the data source, proposing to reduce latency and bandwidth usage.

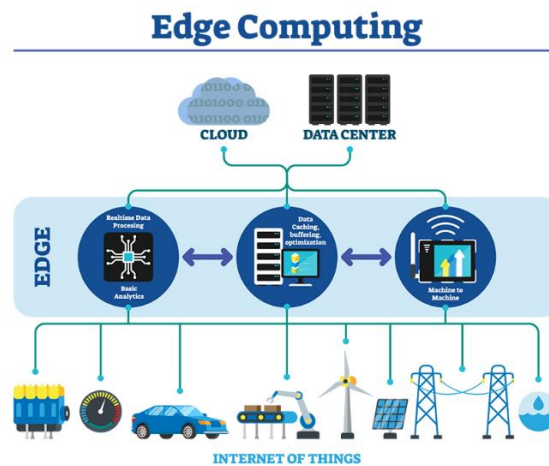


Figure 105: Layered structured of the data processing. Adapted from [54]

New PMUs and PQ devices will perform the requested measurements necessary for the utilities, using the gathered information in the cloud and, possibly, exchange common needed data among each other via the newly developed Process-Bus protocol IEC 61850-9-2 [15] in order to maximize the efficiency of the system.

Whereas concerning performances, the conduct of these measurements simultaneously with its implementation in a DT model proposes to perform both ordinary and extraordinary assessments.

Ordinary assessments are daily required by utilities to ensure the correct operation of the electric network as they involve, for instance, the identification of fault branches, billing for clients and the prevention of islanded operation if the penetration of DERs is high.

On the other hand, the peculiar target of this kind of project is to accomplish also a predictive identification of possible fault locations in electrical assets, as it commonly occurs in cable joints or terminations, and an overall predictive maintenance of the infrastructure which would prevent the utility from additional costs.

Furthermore, among the many goals, also an analysis of the collected data is proposed. This is such that other useful functionalities will be realized: for example, examining the trend of data over time, overloads could be predicted in portions of the network so that countermeasures might be taken in order to avoid interruptions.

The application of DT through the usage of Synchrophasors as well as of PQ data enable utilities to effectively to improve the load forecast analysis and to apply specific algorithms for assets aging evaluation and to perform accurate cost analysis.

All of this will be possible thanks to the double “picture” of the network and the analysis of the processed information.

5.3 Expected outcomes

The main expected results of these projects should be the capability of reducing the present number and minutes of interruptions per client and an overall improving of the Key Performance Indicators (KPIs) of the DSO, thus lower penalties and higher benefits for the utility.

Moreover, thanks to predictive maintenance and anomaly detections, an Operating Expenditure (OPEX) reduction should be reached, in addition to the detection of technical and non-technical losses and safety, such as on-line detection of anomalies inside primary and secondary substations alerting for unsafe internal conditions [55].

The evaluation of the results will be based on the current state-of-the-art. Precisely, the attained outcomes of the new instrumentation, after a calibration in a laboratory site, will be compared with the existing one in terms of performances and accuracy of the accomplished results.

To serve this purpose of evaluating the metrological performances of the realized distribution system of measurements, LabVIEW environment will be used, and the evaluation of uncertainty will be assessed according to the “Guide to the Expression of Uncertainty in Measurements” (GUM).

A further indicator of the achieved aims will be given by the actual reduction of minutes of interruption (e.g. well-known KPIs, like SAIDI and SAIFI) and requests for maintenance when the in-field applicability will be accomplished.

Conclusions

In this thesis, the realization of a PMU model installed in a basic IEEE network is realized and its validation has been performed by means of a real-time simulator.

A preliminary discussion was carried out in the first chapters concerning what are PMUs, the reasons why they are so much needed and the IEEE Standards regulating them; after that, the focus of the thesis was moved to the importance of real-time applications.

This kind of applications are more and more used in industries in order to validate results as well as in Model-Based Design approach.

Afterwards, the model of the network, together with the PMU one, was implemented in Simulink to get a preliminary idea of the goodness of the results of the simulated sensors, compared to the load-flow steady-state data.

Eventually, the same model was implemented in OPAL RT-4510 simulator in order to perform a real-time validation of the results. A final Matlab script was designed in such a way that a numerical comparison of the attained outcomes was performed over a 5-second long simulation, which was regarded to be more than reasonable for a steady-state evaluation.

Finally, in the last chapter, a brief overview of the capabilities of PMUs together with their implementation in a more complex system, which is the DT, was provided.

It is still worth underlying that DTs are not implemented yet in the monitoring of the electrical networks, but they are assumed to be the next step in the control and operation of the grid by utilities, together with PMUs and power quality meters, for the some of the reasons stated in the previous chapters.

The importance of having a dynamic model which is connected to the real one will allow to compare the real-time data coming from the field to the model, with the intention of performing ordinary and extraordinary measurements which will make available the latest state of the grid.

The outcomes of this study represent only a first step towards the real-time validation of PMUs performances in the network as well as the authentication of their results.

Further researches will allow one to eventually realize a system of distributed and synchronized measurements for the implementation of DT techniques to enhance the reliability of the power systems.

List of Abbreviations and Acronyms

AC	Alternating Current
ADC	Analog to Digital Converter
BMS	Battery Management System
CAN	Controller Area Network
CPU	Central Processing Unit
CT	Current Transformer
DC	Direct Current
DDM	Dynamic Digital Mirror
DDoS	Distributed Denial of Service
DER	Distributed Energy Resources
DFT	Discrete Fourier Transform
DG	Distributed Generation
DoS	Denial of Service
DSO	Distribution System Operator
DT	Digital Twin
DUT	Device Under Test
EHV	Extra High Voltage
EMS	Energy Management System
EMT	Electromagnetic Transient
FACTS	Flexible AC Transmission System
FE	Frequency Measurement Error
FPGA	Field Programmable Gate Array
FRACSEC	Fraction Of Second
GNSS	Global Navigation Satellite System
GOOSE	Generic Object Oriented System Event
GPIB	General Purpose Interface Bus
GPS	Global Positioning System
GUI	Graphical User Interface
GUM	Guide to the expression of Uncertainty in Measurements
HIL	Hardware-In-the-Loop
HMI	Human-Machine Interface
HV	High Voltage

HVDC	High Voltage Direct Current
ICT	Information and Communications Technology
IED	Intelligent Electronic Device
IEEE	Institute of Electrical and Electronics Engineers
ILP	Integer Linear Programming
I/O	Inputs and Outputs
IRIG-B	Inter-Range Instrumentation Group code-B
IT	Instrument Transformer
KPI	Key Performance Indicator
LPIT	Low Power Instrument Transformer
MBD	Model-Based Design
MIL	Model-In-the-Loop
MMS	Manufacturing Messaging Specification
MU	Merging Unit
MV	Medium Voltage
OF	Optical Fibre
OPEX	Operating Expenditure
OPP	Optimal PMU Placement
OSI	Open System Interconnection model
PAM	Phase Angle Monitoring
PDC	Phasor Data Concentrator
PEV	Plug-in Electric Vehicle
PhE	Phase Angle Error
PHIL	Power Hardware-In-the-Loop
PLL	Phase-Locked Loop
PMU	Phasor Measurement Unit
PPS	Pulse Per Second
PQ	Power Quality
PTP	Precision Time Protocol
RAE	Relative Amplitude Error
RCP	Rapid Control Prototyping
RFE	ROCOF Measurement Error
RMS	Root Mean Square
ROCOF	Rate Of Change Of Frequency

RT	Real-Time
RTS	Real-Time Simulator
RTU	Remote Terminal Unit
SCADA	Supervisory Control And Data Acquisition
SE	State Estimation
SIL	Software-In-the-Loop
SNTP	Simple Network Time Protocol
SOC	Second Of Century
SV	Sampled Value
THD	Total Harmonic Distortion
TOV	Temporary Overvoltages
TSO	Transmission System Operator
TSSA	Time Synchronization Spoofing Attacks
TVE	Total Vector Error
UTC	Universal Time Coordinated
VT	Voltage Transformer
WAMPAC	Wide Area Monitoring, Protection, Automation and Control
WAN	Wide Area Network
ZIB	Zero-Injection Bus

List of Figures

Chapter 1

Figure 1: Basic PMU structure. Adapted from [6].....	8
Figure 2: Convention for synchrophasor representation. Adapted from [12].....	12
Figure 3: Synchrophasor Network. Adapted from [7]	18
Figure 4: IEC 61850 mapping into ISO/OSI layers. Adapted from [18].....	20
Figure 5: Station and process buses. Adapted from [18]	21
Figure 6: Cyberattack paths and their relationship with information security. Adapted from [21]...	22

Chapter 2

Figure 7: Average TVE for PMU phase phasors and positive sequence (V1) phasors. Adapted from [31].....	26
Figure 8: THD of voltage and current harmonics. THD V/I total harmonic distortion for voltage and current signals. THD V/I error is calculated as the difference of the computed and reference THD. Adapted from [31].....	26

Chapter 3

Figure 9: Real-Time simulation techniques. Adapted from [32]	28
Figure 10: Simulation time-step by application. Adapted from [32].....	29
Figure 11: Suggested EMT simulation time-steps. Adapted from [33].....	30
Figure 12: Application categories. Adapted from [32]	31
Figure 13: PHIL application. Adapted from [37]	32
Figure 14: MBD workflow. Adapted from [32]	33
Figure 15: PMU unit tests. Adapted from [33]	35
Figure 16: Virtual and PMU under test. Adapted from [33].....	35
Figure 17: EMT simulations in RT. Adapted from [33].....	35
Figure 18: RCP of PMUs. Adapted from [33].....	36
Figure 19: Max TVE results of M-class PMUs. Adapted from [33]	36
Figure 20: TVE in phase angle step results. Adapted from [33].....	37
Figure 21: Harmonic distortion tests with (blue) and without (grey) amplifier in the loop. Adapted from [33]	37
Figure 22: TSSA block diagram test setup. Adapted from [31]	38

Chapter 4

Figure 23: IEEE 5-bus network	41
Figure 24: Simulink implementation. Adapted from [43]	41
Figure 25: Results of the load flow calculation	42
Figure 26: Network Topology (nodes in black, node with PMU installed in red)	43
Figure 27: IEEE 5-bus with a PMU installed on bus 2	45
Figure 28: Scopes for the current sensors installed on the lines connected to bus 2	45
Figure 29: PMU installed at bus 2	46
Figure 30: Steady-state measurements in the network.....	47
Figure 31: Import of steady-state data and conversions	47
Figure 32: Simulink code inside PMU Voltage block	48
Figure 33: PMU PLL-based. Adapted from [46]	48
Figure 34: PMU Voltage mask	49
Figure 35: Sampling time of the PMU in Simulink	51
Figure 36: PMU PLL-based blocks	52
Figure 37: Voltage sinusoids comparison block	52
Figure 38: Sinusoid comparison code and data conversion	53
Figure 39: Radians to degrees conversion	53
Figure 40: Reported synchrophasors information.....	54
Figure 41: Indices calculation and reporting rate generators	54
Figure 42: Matlab script for PMU parameters	55
Figure 43: Line-to-line voltages and currents in peak values measured at bus 2	56
Figure 44: Currents measured in the lines incident to bus 2	57
Figure 45: Module and phase of PMU Voltage	58
Figure 46: Detected frequency for voltage and current phasors	59
Figure 47: PLL circuit oscillations in the frequency detection	59
Figure 48: Comparison between ideal and measured voltages inside the PMU	60
Figure 49: Module and phase of PMU Current.....	61
Figure 50: Comparison between ideal and measured currents inside the PMU	61
Figure 51: Indices Calculation block	62
Figure 52: Matlab function in the Indices Calculation block.....	63
Figure 53: Matlab script required to compute accuracy indices for the PMU Voltage block	63
Figure 54: Matlab script required to compute accuracy indices for the PMU Current block.....	64
Figure 55: Voltage FE.....	65

Figure 56: Voltage RAE	65
Figure 57: Voltage PhE.....	66
Figure 58: Voltage TVE.....	66
Figure 59: Approximate Voltage TVE.....	67
Figure 60: Comparison between voltage TVE and approximate TVE	67
Figure 61: Current FE	68
Figure 62: Current RAE.....	68
Figure 63: Current PhE	69
Figure 64: Current TVE	69
Figure 65: Approximate Current TVE.....	70
Figure 66: Comparison between current TVE and approximate TVE.....	70
Figure 67: Typical development process. Adapted from [51]	72
Figure 68: New approach involving simulations during the concept and design phases. Adapted from [51]	72
Figure 69: OPAL RT-4510 I/O channels. Adapted from [53].....	73
Figure 70: OPAL RT-4510 internal structure. Adapted from [53].....	73
Figure 71: Typical OPAL RT simulator setup. Adapted from [55].....	74
Figure 72: Communication between Subsystems. Adapted from [55].....	75
Figure 73: OpComm blocks placement. Adapted from [55]	76
Figure 74: Re-arranged code for implementation on OPAL-RT	76
Figure 75: Matlab script for data initialization in RT-LAB.....	77
Figure 76: SM_NET subsystem.....	78
Figure 77: IEEE 5-bus network in SM_NET	78
Figure 78: PMUs blocks in the SM_NET subsystem	79
Figure 79: OpWriteFile blocks for voltage and current PMUs in SM_NET	80
Figure 80: OpWriteFile parameters mask in SM_NET	80
Figure 81: Model pause and simulation data in SM_NET.....	82
Figure 82: SC_SCOPE subsystem	83
Figure 83: Programme running in RT on OPAL simulator	84
Figure 84: Real-time overruns monitoring.....	84
Figure 85: Currents measured in the lines incident to bus 2 in OPAL-RT.....	85
Figure 86: Voltage and current at bus 2 in OPAL-RT	86
Figure 87: Amplitude (blue), phase (red) and frequency (yellow) of voltage synchrophasor in OPAL-RT.....	87

Figure 88: Accuracy indices of voltage PMU in OPAL-RT.....	87
Figure 89: Comparison between ideal and measured voltage sinusoid in OPAL-RT	88
Figure 90: Amplitude (blue), phase (red) and frequency (yellow) of current synchrophasor in OPAL-RT.....	88
Figure 91: Accuracy indices of current PMU in OPAL-RT	89
Figure 92: Comparison between ideal and measured current sinusoid in OPAL-RT.....	89
Figure 93: Final elaboration of Simulink for voltage data.....	91
Figure 94: Final elaboration of Simulink for current data	91
Figure 95: Final elaboration of OPAL-RT voltage data	92
Figure 96: Final elaboration of OPAL-RT current data.....	92
Figure 97: PMU structure containing Simulink and OPAL-RT accuracy indices results	92
Figure 98: Final elaboration of Simulink synchrophasors	93
Figure 99: Final elaboration of OPAL-RT synchrophasors.....	93
Figure 100: Phasors structure containing Simulink and OPAL-RT synchrophasor results.....	94
Figure 101: Analysis of simulation data from OPAL-RT.....	94
Figure 102: SimulationInfo structure containing OPAL-RT simulation data.....	95

Chapter 5

Figure 103: Digital Twin. Adapted from [56].....	96
Figure 104: Digital Twin concept for Power System Control Centres. Adapted from [30]	98
Figure 105: Layered structured of the data processing. Adapted from [57]	98

List of Tables

Chapter 1

Table 1: Reporting rates. Adapted from [12].	15
Table 2: Steady-state measurement requirements. Adapted from [12].	16
Table 3: Measurement reporting latency. Adapted from [12]	17

Chapter 4

Table 4: Bus data for IEEE 5-bus. Adapted from [43]	42
Table 5: Line data for IEEE 5-bus. Adapted from [43]	42
Table 6: Load flow solution computed in Simulink	43
Table 7: Selected sampling times	56
Table 8: Selected simulation parameters	56
Table 9: Measured voltages and currents at bus 2	57
Table 10: Current sensors data in Simulink	57
Table 11: Current sensors data in OPAL-RT	85
Table 12: Voltage and current at bus 2 in OPAL-RT	86

Bibliography

- [1] R. Kuffel, P. Forsyth, and C. Peters, “The Role and Importance of Real Time Digital Simulation in the Development and Testing of Power System Control and Protection Equipment,” *IFAC-PapersOnLine*, vol. 49, no. 27, pp. 178–182, 2016, doi: 10.1016/j.ifacol.2016.10.739.
- [2] A. Gómez-Expósito, A. Abur, P. Rousseaux, A. De La Villa Jaén, and C. Gómez-Quiles, “On the use of PMUs in power system state estimation,” *17th Power Syst. Comput. Conf. PSCC 2011*, no. January, 2011.
- [3] A. Monti, C. Muscas, and F. Ponci, *Phasor Measurement Units and Wide Area Monitoring Systems: From the Sensors to the System*. 2016.
- [4] IEEE Power & Energy Society, *IEEE Std C37.118.2TM-2011 - IEEE Standard for Synchrophasor Data Transfer for Power Systems*, vol. 2011, no. December. 2011.
- [5] IEEE Power & Energy Society, *IEEE Std C37.242TM-2013 - IEEE Guide for Synchronization, Calibration, Testing and Installation of Phasor Measurement Units (PMUs) for Power System Protection and Control*, no. March. 2013.
- [6] IEEE Power & Energy Society, *IEEE Std C37.244TM-2013 Authorized - IEEE Guide for Phasor Data Concentrator Requirements for Power System Protection, Control, and Monitoring*, no. May. 2013.
- [7] F. Aminifar, M. Fotuhi-Firuzabad, A. Safdarian, A. Davoudi, and M. Shahidehpour, “Synchrophasor Measurement Technology in Power Systems: Panorama and State-of-the-Art,” *IEEE Access*, vol. 2, pp. 1607–1628, 2014, doi: 10.1109/ACCESS.2015.2389659.
- [8] IEEE Instrumentation and Measurement Society, “IEEE Std 1588TM-2019 - IEEE Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems,” 2019. <https://ieeexplore-ieee-org.ezproxy.unibo.it/stamp/stamp.jsp?tp=&arnumber=9120376> (accessed Aug. 12, 2020).
- [9] IEEE Power & Energy Society, *IEEE Std C37.118.1TM-2011 - IEEE Standard for Synchrophasor Measurements for Power Systems*, vol. 2011, no. December. 2012.
- [10] A. K. Srivastava, “Phasor Measurement (Estimation) Units.”
- [11] G. Sanchez-Ayala, J. R. Aguerce, D. Elizondo, and M. Dino Lelic, “Current trends on applications of PMUs in distribution systems,” *2013 IEEE PES Innov. Smart Grid Technol. Conf. ISGT 2013*, pp. 1–6, 2013, doi: 10.1109/ISGT.2013.6497923.
- [12] M. Ghalei, M. Zanjani, H. K. Karegar, H. A. Niaki, and M. Ghalei, “High Impedance Fault Detection of Distribution Network by Phasor Measurement Units,” vol. 4, pp. 297–305,

2013, doi: 10.4236/sgre.2013.43036.

- [13] A. Mingotti, L. Peretto, R. Tinarelli, A. Angioni, A. Monti, and F. Ponci, “A simple calibration procedure for an LPIT plus PMU system under off-nominal conditions,” *Energies*, vol. 12, no. 24, pp. 1–19, 2019, doi: 10.3390/en12244645.
- [14] A. Mingotti, L. Peretto, R. Tinarelli, A. Angioni, A. Monti, and F. Ponci, “Calibration of Synchronized Measurement System: From the Instrument Transformer to the PMU,” *9th IEEE Int. Work. Appl. Meas. Power Syst. AMPS 2018 - Proc.*, 2018, doi: 10.1109/AMPS.2018.8494887.
- [15] ABB, “Technical Application Papers No. 19 Smart grids ‘3. Standard IEC 61850,’” 2017.
- [16] D. Rösch et al., “Local anomaly detection analysis in distribution grid based on IEC 61850-9-2 LE SV voltage signals,” *IEEE*, 2019. <https://ieeexplore-ieee.org.ezproxy.unibo.it/stamp/stamp.jsp?tp=&arnumber=8849139> (accessed Aug. 12, 2020).
- [17] A. Kummerow et al., “Challenges and opportunities for phasor data based even detection in transmission control centers under cyber security constraints,” *IEEE*, pp. 1–6, 2019.
- [18] D. J. Kang, H. T. Kim, and S. Choi, “Methodology for quantifying the economic impact of cyberattacks on bulk electric systems,” *Conf. Rec. - Ind. Commer. Power Syst. Tech. Conf.*, vol. 2019-May, pp. 1–5, 2019, doi: 10.1109/ICPS.2019.8733322.
- [19] C. Riedel, G. Fu, D. Beyette, and J. C. Liu, “Measurement system timing integrity in the presence of faults and malicious attacks,” *2019 Int. Conf. Smart Grid Synchronized Meas. Anal. SGSMA 2019*, 2019, doi: 10.1109/SGSMA.2019.8784502.
- [20] IEEE Power and Energy Society, *IEEE Std C37.240TM-2014 Authorized - IEEE Standard Cybersecurity Requirements for Substation Automation , Protection , and Control Systems*. 2014.
- [21] IEEE Std 1159-2019, *IEEE Recommended Practice for Monitoring Electric Power Quality*, vol. 2019. 2019.
- [22] J. Kilter *et al.*, “Current Practice and Future Challenges for Power Quality Monitoring - CIGRE WG C4.112 Perspective,” *2012 IEEE 15th Int. Conf. Harmon. Qual. Power*, vol. 00, pp. 390–397, 2011, doi: 10.1109/ICHQP.2012.6381311.
- [23] H. Markiewicz, A. Klajn, and W. University of Technology, “Standard EN 50160 Voltage Characteristics in Public Distribution Systems 5.4.2,” 2004. [Online]. Available: www.lpqi.org.
- [24] IEC, “IEC 61000-2-2 - Electromagnetic compatibility (EMC) - Part 2-2: Environment - Compatibility levels for low-frequency conducted disturbances and signalling in public low-voltage power supply systems.” <https://webstore.iec.ch/publication/63116> (accessed Aug. 14,

2020).

- [25] IEC, “IEC 61000-4-30 - Electromagnetic compatibility (EMC) - Part 4-30: Testing and measurement techniques - Power quality measurement methods,” 2003. [Online]. Available: www.iec.ch.
- [26] MIGRATE - Massive InteGRATion of power Electronic Devices, “Critical PQ phenomena and sources of PQ disturbances in PE rich power systems - Deliverable 5.1,” no. 691800, pp. 1–158, 2016.
- [27] C. Brosinsky, D. Westermann, and R. Krebs, “Recent and prospective developments in power system control centers: Adapting the digital twin technology for application in power system control centers,” in *2018 IEEE International Energy Conference, ENERGYCON 2018*, Jun. 2018, pp. 1–6, doi: 10.1109/ENERGYCON.2018.8398846.
- [28] M. Loper, T. Trummal, and J. Kilter, “Analysis of the Applicability of PMU Measurements for Power Quality Assessment,” *Proc. - 2018 IEEE PES Innov. Smart Grid Technol. Conf. Eur. ISGT-Europe 2018*, vol. 691800, no. 691800, pp. 1–6, 2018, doi: 10.1109/ISGTEurope.2018.8571618.
- [29] J. Bélanger, P. Venne, and J. N. Paquin, “The What, Where and Why of Real-Time Simulation,” pp. 37–39.
- [30] S. Li, L. Zhang, J. N. Paquin, J. Belanger, and L. Vanfretti, “Hardware-in-the-loop use cases for synchrophasor applications,” *2019 Int. Conf. Smart Grid Synchronized Meas. Anal. SGSMA 2019*, 2019, doi: 10.1109/SGSMA.2019.8784526.
- [31] Y. Shu, H. Li, and Q. Wu, “Expansion Application of dSPACE for HILS,” pp. 2231–2235, 2008.
- [32] C. A. Rabbath, M. Abdoune, and J. Belanger, “Effective Real-Time Simulations of Event-Based Systems,” in *Proceedings of the 2000 Winter Simulation Conference J. A. Joines, R. R. Barton, K. Kang, and P. A. Fishwick, eds*, 2000, pp. 232–238, Accessed: Aug. 18, 2020. [Online]. Available: <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=899723>.
- [33] R. Bednar and R. Crosbie, “Stability of multi-rate simulation algorithms,” 2007, Accessed: Aug. 18, 2020. [Online]. Available: https://www.researchgate.net/publication/221113046_Stability_of_multi-rate_simulation_algorithms.
- [34] T. Kirk, “Real-Time Simulation for Energy Storage Applications,” 2019.
- [35] M. Matar and R. Iravani, “FPGA Implementation of the Power Electronic Converter Model for Real-Time Simulation of Electromagnetic Transients - IEEE Journals & Magazine,” *IEEE Transactions on Power Delivery*, 2009.

- <https://ieeexplore.ieee.org/document/5345690/references#references> (accessed Aug. 18, 2020).
- [36] OPAL-RT, “Real-Time Transient Stability Simulator The Ultimate Solution for Large-Scale Power System Simulations.”
- [37] OPAL-RT, “Power system simulation | Power system Analysis | HYPERSIM.” <https://www.opal-rt.com/systems-hypersim/> (accessed Aug. 20, 2020).
- [38] M. S. Almas, L. Vanfretti, R. S. Singh, and G. M. Jonsdottir, “Vulnerability of synchrophasor-based WAMPAC applications’ to time synchronization spoofing,” *IEEE Trans. Smart Grid*, vol. 9, no. 5, pp. 4601–4612, 2018, doi: 10.1109/TSG.2017.2665461.
- [39] I. Abdulrahman and G. Radman, “ILP-Based Optimal PMU Placement with the Inclusion of the Effect of a Group of Zero-Injection Buses,” *J. Control. Autom. Electr. Syst.*, vol. 29, no. 4, pp. 512–524, 2018, doi: 10.1007/s40313-018-0389-4.
- [40] T. Rodney, “IEEE 5-Bus System Model - File Exchange - MATLAB Central,” 2018. <https://it.mathworks.com/matlabcentral/fileexchange/66555-ieee-5-bus-system-model> (accessed Aug. 26, 2020).
- [41] V. Jaiswal, S. S. Thakur, and B. Mishra, “Optimal placement of PMUs using Greedy Algorithm and state estimation,” *1st IEEE Int. Conf. Power Electron. Intell. Control Energy Syst. ICPEICES 2016*, no. 3, pp. 3–7, 2017, doi: 10.1109/ICPEICES.2016.7853556.
- [42] B. Mishra, S. S. Thakur, S. Mallick, and C. K. Panigrahi, “Optimal Placement of PMUs for Power System State Estimation,” *1st IEEE Int. Conf. Sustain. Energy Technol. Syst. ICSETS 2019*, pp. 188–193, 2019, doi: 10.1109/ICSETS.2019.8745327.
- [43] Simulink - MathWorks, “PMU (PLL-Based, Positive-Sequence),” 2020. <https://it.mathworks.com/help/releases/R2020a/phymod/sps/powersys/ref/pmupllbasedpositivevesequence.html> (accessed Sep. 19, 2020).
- [44] I. Krish Narendra, Senior Member, IEEE, Dinesh Rangana Gurusinghe, Graduate Student Member, IEEE, and Athula D. Rajapakse, Senior Member, “Dynamic Performance Evaluation and Testing of Phasor Measurement Unit (PMU) as per,” *IEEE Trans. Power Appar. Syst.*, vol. 01, no. October, pp. 270–279, 2012.
- [45] A. Mingotti, L. Peretto, R. Tinarelli, A. Angioni, A. Monti, and F. Ponci, “Calibration of Synchronized Measurement System: From the Instrument Transformer to the PMU,” *9th IEEE Int. Work. Appl. Meas. Power Syst. AMPS 2018 - Proc.*, pp. 1–5, 2018, doi: 10.1109/AMPS.2018.8494887.
- [46] “Real-time simulation.” https://en.wikipedia.org/wiki/Real-time_simulation (accessed Sep. 23, 2020).

- [47] H. Chang, “Power Electronics Control Design & Testing in the 21st century,” 2016. <https://info.typhoon-hil.com/blog/power-electronics-control-design-and-testing-in-the-21st-century> (accessed Sep. 23, 2020).
- [48] C. McClintock, “Real-Time Simulation Explained,” 2019. <https://www.ptc.com/en/cad-software-blog/real-time-simulation-explained> (accessed Sep. 23, 2020).
- [49] OPAL-RT, “Resource Center : Document - OPAL-RT4510.” https://www.opal-rt.com/resource-center/document/?resource=L00161_0124 (accessed Sep. 23, 2020).
- [50] OPAL-RT, “OP4510 Simulator.” <https://www.opal-rt.com/simulator-platform-op4510/> (accessed Sep. 23, 2020).
- [51] OPAL-RT, “OP4510 V2 - Hardware Products Documentation - Wiki OPAL-RT.” <https://wiki.opal-rt.com/display/HDGD/OP4510+V2> (accessed Sep. 23, 2020).
- [52] OPAL-RT, “Preparing a Simulink model for Real Time Execution (Lesson) - OPAL-RT.” https://www.opal-rt.com/opal_tutorial/preparing-simulink-model-real-time-execution/ (accessed Sep. 23, 2020).
- [53] OPAL-RT, “The ‘Digital Twin’ in Hardware in the Loop (HiL) Simulation: A Conceptual Primer,” 2020. <https://www.opal-rt.com/the-digital-twin-in-hardware-in-the-loop-hil-simulation-a-conceptual-primer/> (accessed Aug. 30, 2020).
- [54] IEEE Innovation at Work, “Real-Life Use Cases for Edge Computing,” 2020. <https://innovationatwork.ieee.org/real-life-edge-computing-use-cases/> (accessed Aug. 31, 2020).
- [55] M. Lombardi, A. Cammarota, and J. Refoyo, “Development, applications and benefits of the network digital twin,” *25th Int. Conf. Electr. Distrib. Int. Conf. Electr. Distrib.*, no. June, pp. 1–3, 2019, doi: 20.500.12455/751.