Linear state estimation for distribution grids

Sotirios Dimitrakopoulos

using measurements provided by

DEPsys S.A.,
Switzerland

TU Delft

Linear state estimation for distribution grids

by

Sotirios
Dimitrakopoulos

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Thursday, 18 July of 2019.

Student number: 4736397
Project duration: December 13, 2018 – July 18, 2019
Thesis committee:

Prof. dr. P. (Peter) Palensky,	TU Delft, Supervising Professor
Dr. S. H. (Simon) Tindemans,	TU Delft, Daily Supervisor
Dr. D. (Domenico) Lahaye,	TU Delft, Applied Mathematics
Dr. O. (Omid) Alizadeh-Mousavi,	DEPsys. S.A., Daily Supervisor

TU Delft

Preface

The recent trend for monitoring and control of the distribution networks requires methods capable to calculate the network's operating conditions. One way to monitor the distribution network is by state estimation. Although previously this method was established only for the transmission networks, its use has begun to also make sense for the distribution networks, due to the increased availability of smart meters.

Typically, state estimation is a non-linear iterative procedure. Research aims to overcome this problem, by creating a non-iterative procedure. To achieve this, linearity of measurement function is a necessity. Although linearity is always desired, the measurement inputs or the procedure renders the state estimation problem difficult to formulate it in a linear fashion. The scope of this thesis is to create a linear state estimation algorithm for the distribution network.

To achieve linearity, a reformulation of the states for the state estimation was used at this thesis. This reformulation created a linear optimization problem, which is easier and faster to be solved (Weighted Least Squared method). Since the measurements are not enough to provide full observability, based on this methods, additional assumptions were made.

As an alternative to the primary method used for the state estimation, another optimization method was used, which is called Least Absolute Value. This method will be based on the same inputs as the previous method, but different optimization goal. The goal was to compare the two methods and find each one's limitations.

Measurements used for this thesis were provided tracked and provided by DEPsys S.A., by their main module, called GridEye. GridEye devices are synchronized SCADA-type devices, installed on cabinets of the distribution grid, based on the DSO needs. All measurements provided were from real networks within Switzerland. The smart meter device data were provided by collaborative projects of DEPsys S.A.. Smart meter devices were installed in different low-voltage consumers.

Both methods were implemented correctly and were compared. Based on the results, it was obvious that the Weighted Least Square outperformed the Least Absolute Value method, in the absence of bad data.

Moreover, the effect of the number of GridEye metering devices on the State Estimation was compared. While numerically and visually the outputs are comparable, the effect of having multiple devices is still important for different scenarios.

Finally, bad data cases were examined for both methods under different types of errors for the smart meter devices. The smart meter devices were manually introduced to bad data, to understand the algorithm's capabilities and limitations. An additional algorithm, in addition to the state estimation one, was implemented to not only detect and identify the location of the bad data occurrence, but to correct the bad data inserted.

Sotirios
Dimitrakopoulos
Delft, July 2019

Acknowledgments

To my family (incl. Frankie)
and everyone who supported me

Contents

List of Abbreviations

GE GridEye device.

LAV Least Absolute Value.

PF Power Flow.

RMSE Root mean square error.

SCADA Supervisory Control and Data Acquisition.

SE State Estimation.

SM Smart meter device.

WLS Weighted Least Square.

List of Figures

List of Tables

1 Introduction

Power system networks are quite complex by nature. This complexity tends to increase in recent years, as the energy demands are shifting and local power production is generating greater traction. These changes require adaptations in the monitoring and control of the power systems. To validate the secure operation of such a network, monitoring of operating states of the network are required. Typically, control methods are implemented via evaluating measurements of voltage magnitudes, active and reactive power flows. These measurements are then used to provide a reliable estimate, to access the degree of trust of the validity of the measurements and the network's operating conditions.

The need of control and better understanding of the network led to research for methods of acquiring network information out of the network's measurements. Using network measurements, the aim is to acquire complete network knowledge, or at least an adequate overview of strategic points within the network. Acquiring overview of the network's operating conditions provides outcomes, which are the basis for the development of further actions of the Energy Management System (EMS) within the network, like for example Demand Response (DR) techniques.

The transmission systems have been traditionally more important to measure and control, which lead to many measurement devices in such a network. This created a high redundancy of measurements availability. While measurement and control can still be improved for transmission networks, the challenge lies in the distribution systems, which up until recently were totally unmonitored. The changes of the distribution network, where local production is gathering greater interest in the recent years, led to high research in the control and monitor of the distribution systems. This challenge still remains difficult to overcome due to the distribution's network complexity and lack of measurements availability.

The correct estimation of the network operating conditions is crucial. The first steps to acquiring an overview of the transmission's network state was by solving load-flow calculations, providing the raw measurements directly acquired from the measurement units. A significant disadvantage of this method was its inability to cope with missing and bad measurements, as well as lack of flexibility, since specific measurements at specific points we required.

1.1. Introduction to State Estimation

To tackle the inflexibility of power flow, Schweppe introduced an alternate method, called State Estimation [1], which was based on solving an optimization problem to *estimate* mathematically the state values. This measurement processing algorithm for converting redundant measurements into states revolutionized the monitoring of transmission systems. States were defined as the minimum set of independent variables, from which all the power-system quantities can be computed [2]. Full network knowledge can be acquired by successfully calculating the system states.

The scope of this mathematical approach was to use measurements and topology knowl-

edge to reach a state estimate. What this approach ensured was overcoming the inflexibility and error dependency of load flow by taking advantage of the network measurement's redundancy. This has led to the development of Energy Management System, which helps the grid operator to manage the energy assets within a network [3]. The EMS system includes the Supervisory Control and Data Acquisition (SCADA) system and the Remote Terminal Units (RTU). The measurement quantity was never an issue in the transmission network, where the State Estimation algorithm was initially developed. The estimator's goal was to produce the best states by minimizing the cost function, or the difference between estimated and actual measurements.

The transmission network, usually equipped with a SCADA system, collects measurements from various devices, which are the necessary inputs for the state estimator. The transmission network is equipped with multiple devices and acquires measurements. The recent shift towards a smarter grid, one that encompasses bidirectional flow and local production, made monitoring a necessity for the distribution grid.

1.2. State Estimation's limitations for Distribution Grids

Much research focused on evolving the state estimation method for the distribution network. Nevertheless, there are major differences of the distribution network and the transmission one [4]:

- The distribution grid is mostly radial, compared to the the meshed one of the transmission one. There is a shift towards some meshed distribution networks, but still the majority of them exist in a radial way.
- A distribution grid is mostly unbalanced between the three phases, while the transmission network is under 3-phase balanced conditions. This is because the lack of knowledge in the load allocation in the distribution network, as well as the unpredictability of the load consumption within the distribution network.
- In the distribution grid there are higher R/X ratios, meaning that lower impedance between the nodes. The lower impedance creates in turn small phase angle differences between nodes.
- Last and foremost, the distribution grid is mostly unobservable, meaning that there is a significant lack of measurement, compared to the transmission network. In the transmission network, it is safe to assume that there is a measurement redundancy to calculate the state estimation, whereas in the distribution grid the lack of measurement is a contributing factor to the lack of development of state estimating techniques.

Initially, State Estimation algorithm was developed by using only conventional measurements provided by the RTU, which consisted of voltage magnitudes and power flows. These measurements are typically called conventional measurements or SCADA type measurements. Devices provide voltage magnitude and power, reactive power flows. The overcome the network's lack of observability, in most recently developed state estimators, pseudomeasurements based on historical data or assumptions are mostly used. This way, formulated measurements play a role in acquiring sufficient measurements for the state estimation. To validate the state estimator's correct initialization, observability analysis and topology processing are required.

1.3. Measurements provided by DEPsys S.A.

Currently, metering devices in the distribution grid are installed in transformer's cabinets, providing SCADA-type measurements at the installed location. SCADA main problems still remain the low-refresh rate and the lack of measurement time-stamp. Sometimes time-stamps are provided, but with poor accuracy. This makes SCADA system highly uncooperative with other measurement type of data. To solve the problem of having to deal with unsynchronized and low-resolution measurements in the distribution system, a typical measurement problem that the DSOs have to face, DEPsys S.A. was created.

DEPsys S.A. is a Swiss-based company with the aim to revolutionize the active distribution grid. The continuous increase of distributed generation in combination with the lack of active control of the grid led to the idea of creating a device capable of calculating complicated measurements. Although a start-up, the company grew from 2012 to its current size, and has expanded to the worldwide market (Germany, Cyprus, Singapor, Austria, Ireland), aiming to expand its current portfolio of clients even more. Today, DEPsys doesn't just focus on providing measurements for the low and medium voltage grid, but the company aims to provide solutions to enable the current electricity networks to cope with the new constraints imposed by the ever-growing decentralized production from renewable energy sources. To solve these problems, a device called GridEye (GE) [5] was created. A typical GridEye device is shown in figure 1.1.

Figure 1.1: GridEye device.

GridEye is a complex architecture, containing three parts, the hardware, software and server. As a hardware, the GridEye provides the possibility of measuring and controlling the active elements and assuring communication with the server. The installation point is usually at transformer or cabinets within the network, as the DSOs see fit. The software ensures that grid intelligence is acquired without the need to measure each individual point of the network. The software contains multiple processes that achieve high operation knowledge within the network. These processes contain power quality measurement, optimal control, network planning and many more. Finally, all these are communicated to a server, which in turn provides visualization, storage and data configuration. More information about the product and everything that it can achieve can be found in [5]

Since the first step towards a distribution system state estimator is sufficient knowledge in the MV/LV transformers, GridEye data provide a great tool to acquire better knowledge on the operating conditions of transformers and cabinets within the network. The measurements

include voltage, power, current and power quality. If the voltage phase and magnitude are uncorrelated, the measurements are considered as SCADA-type measurements. Thus, the measurements provided are still considered as **synchronized SCADA-type measurements**, since the measurements contain provided voltage only in the form of magnitude, without in any way providing sufficient information to extract the voltage phase angle. Typical accuracy level of GridEye devices are 0.1% possible error for voltage and 1% error for current.

Despite GridEye's accuracy in the installed location, its use makes only sense at transformers and other strategic position of high significance (i.e. cabinets of high consumption), where the DSOs prefer having better operating knowledge of those points. For network observability condition to hold, including the end-customer level, additional measurements are required.

The rise of the smart metering devices (SM) led in research for their possible use in roles other than visualizing the consumption over specific time. Their implementation was suggested for state estimators, to presumably solve the observability issue within such networks. Their disadvantages, however, of high-communication costs, low implementation number within the network, intrinsic measurement error of the device and lack of sufficient information, since most of currently installed smart meters provide only energy consumed through a specified time period, render them inadequate to be used in their own as stand-alone measurements for network analysis purposes. It must be mentioned, that although DEPsys S.A. does not own the Smart meter devices, the measurements were provided by agreement between DEPsys and companies providing smart meters.

To this end, focus of state estimation using multiple measurement inputs is considered. The problem today still lies in the difficulty of synchronization and availability between different measurements types. This can be tackled by using GridEye devices and smart meter devices within the same network, which can complement the previously laid disadvantages. The goal of this thesis is to provide an alternative way of the typical state estimation algorithm, using measurements of the GridEye devices (high-resolution measurements) and advanced metering infrastructure devices (low-resolution). This way, higher network knowledge can be achieved.

1.4. Research questions

The main objective of this thesis is to create a **linear state estimation algorithm for the low-voltage grid using multiple measurement types as inputs**. This algorithm will be created and tested based on a single network with multiple smart meters and GridEye devices.

The main research question is **whether such an algorithm is applicable for the low-voltage network and what are the errors produced for the linear state estimation algorithm under assumptions for the smart meters**. The lack of redundancy in the distribution grid, is a setback for the state estimation. Thus, this thesis project aims to analyze the results and outputs of such an estimator. The rest of the research questions are summarized below:

- **Which state estimation algorithms can be considered, that satisfy both high precision and computation time?**
  There are many different methods to implement a state estimation algorithm. The main difference between each method lies in the objective solver and the associated solver. Due to the difference in each optimization, different results and computation times occur. Thus, only the most promising ones can be considered.
- **How can the selected state estimation algorithm be adapted for use in distribution grids?**
  All state estimation methods for power system methods were initially considered and tested for the transmission network. The transmission network contains high measurement redundancy. What are the steps to adapt an algorithm developed for the transmission network towards the distribution network?
- **How does the algorithm perform with different redundancy levels?**
  Generally, the main issue of power flow analysis is the iterative procedure and the lack

of use of redundancy. The aimed algorithm of the state estimation aims to tackle both of these problems. Since the smart meters are mostly providing active power outputs every 30 minutes, and these data are only available at the end of the day, assumptions must be made to create such an algorithm. The question lies in the validation of the degree of the errors between the power flow and the state estimation outputs.

- **How does the objective function of the state estimation influence the outputs of the estimator?**
  State estimation by nature requires an optimization goal. Thus, the method used, defines by a large margin the outputs of the estimator. To this end, two methods will be used (WLS and LAV). The considered methods are only compared for output accuracy and how they deal with bad data, since LAV optimization requires certainly higher computation costs.

- **How does redundancy in the higher-resolution (GridEye) devices affect the state estimation?**
  As mentioned, state estimation was originally created to take advantage of measurement redundancy. Nevertheless, the low-voltage network operation is highly unobservable. Most times only the transformers are measured, due to their high value to the network's understanding. But a question lies in what happens if more high-resolution measurements are available and could be used for the state estimation. The question that arises is, does higher network knowledge create a higher algorithmic performance?

- **How does the algorithm deal with bad data?**
  A major problem observed is the lack of trust in the measurement outputs, especially the smart meter provided data. Lost data and failing communications are the major problems for bad data in smart meter measurements. In the transmission grids, the redundancy offers some methods that can tackle this problem? This thesis aims to answer the question, if the low-voltage network bad data identification is possible.

1.5. Thesis outline

The work-flow of the thesis is divided into five chapters. In chapter 2, mathematical background for state estimators is provided and typical methods of state estimation are briefly explained. In chapter 3, the implemented state estimators used for this thesis will be discussed, mathematically analyzed and explained. In chapter 4, the test-cases of the methods will be provided and comparisons will be made. In chapter 5 the conclusions will be discussed, as well as future possible ideas for expansion of the state estimation algorithm. Chapter 6 will be the bibliography.

2

Introduction to State Estimation

This chapter provides the necessary mathematical background for State Estimation (SE), the general methods and the limitations. The thorough analysis of the methods used will be in chapter 3.

2.1. Basic introduction

Contrary to the power flow analysis (PF), state estimation is a mathematical method of correlating measurements and states, which as previously mentioned, are the minimum set of independent variables from which all the power-system quantities can be computed. State estimation evaluates a set of provided measurements to establish the operating conditions of the network. The evaluation is done by an optimization, which aims to minimize the error between measured and estimated values. Estimated values are established via a measurement function, which contains the correlation between states and provided measurements. This measurement function aims create estimated measurements, by minimizing the errors that exists between estimated and metered measurements.

The main advantage of the state estimation is the flexibility of the measurements. Obviously, the quantity of measurements must suffice to perform such an optimization. Adequate measurement availability is called observability analysis and evaluates whether the measurements, states and measurement function can provide state estimation outputs. Thus, it assures mathematical feasibility. Every state estimation formulation requires definition of the following:

- States definition
- Method
- Measurement function

Following the observability analysis, states must be defined, in a way that they can completely characterize network's operating conditions.

2.2. States definition

The two common types of states are node-voltage-based and a branch-current-based. While both are used, depending on the advantages of each [6], the node voltage is the typical one used. Examples of branch-current-based state estimations are in [7] [8]. This is because voltage validation is easier compared to branch-current validation. Nevertheless, states are adaptable to the needs of every method.

Since measurement devices typically provide voltage magnitude, phase angle knowledge is a necessity for the complete network knowledge. Thus, for a system with N buses, the number of state variables must be $2N - 1$, since the slack voltage phase angle is always zero

as the reference phase angle. A typical example of states in the form of voltage magnitude and phase angle in a network of N number of buses is:

$$x = [V_1 \quad V_2 \quad \dots \quad V_N \quad \delta_2 \quad \delta_3 \quad \dots \quad \delta_N]^T \tag{2.1}$$

2.3. Measurement function

Following the definition of the system states, the correlation between the states and estimated outputs must be formed, to calculate the estimated outputs. These outputs will be compared to the measured ones. Equations are formed within the measurement function $h(x)$, which is completed by establishing the relationship between states and outputs. Usually in power systems, these equations are Kirchhoff voltage or current laws and their expansions. These equations produce "expected" results, which aim to match the provided measurements. Since the perfect match is impossible in real scenarios, the goal is to minimize the error between expected outputs and measurements. For m measurements and for N buses, the equations for minimizing the error are:

$$z = \begin{pmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{pmatrix} = \begin{pmatrix} h_1(x) \\ h_2(x) \\ \vdots \\ h_m(x) \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{pmatrix} = h(x) + e \tag{2.2}$$

where: z is the measurement vector of m number of measurements, $h_i(x)$ is a specific measurement function based for the measurement m_i and e_i is the error between the estimated and the measured value. Since the definition of states is flexible, the measurement function is flexible as well. The measurement function contains topological and measurement information with respect to the state variables. Essentially, state variables are correlated with the measurement function based on the equations provided in the measurement function, thus creating estimated outputs. These estimated outputs aim to match the measurements provided. Following the equations formulation, the estimation problem is formulated as an optimization of solving equation 2.3 :

$$\begin{aligned} min \quad & J(z - h(x)) \\ s.t. \quad & g(x) = 0 \end{aligned} \tag{2.3}$$

where J is the objective function, $x \in R^{2n-1}$ is the state vector, $z \in R^m$ is the measurement vector and $h :\in R^{2n-1} \rightarrow R^m$ is the measurement function. Constraints can be provided in the form of equality or inequality constraints within the constraint function $g(x)$. All optimization methods aim to minimize the objective function J, but the means of each optimization is different. While the optimization always aims to minimize 2.3, the minimization goal of J is different for each method. For example, some methods try to minimize the square root of errors or the absolute of the errors, while others can also take weights into consideration. Thus J depicts optimization goal of each method.

2.3.1. Linearity and non-linearity within the State estimation

Linearity in the State Estimation is established by linear correlation between the states and the measurements function. State estimation was initially an approach to solve a non-linear optimization problem. The non-linearity mentioned, in the power system, is the result of the function that links power or energy to voltage and current, as shown in 2.4. To be precise, to calculate power and its flow direction, voltage and current as complex numbers are required. Since the standard state definition of 2.1 has separate states for voltage magnitude and phase angle, a multiplication between states creates a non-linear problem. This, in turn, creates an iterative procedure for the optimization calculation, which affects mainly the computational costs and efficiency of the algorithm.

$$\begin{aligned} S_j &= \boldsymbol{V}_i I_j^* \\ &= |V_i| e^{\delta_i} I_j^* \end{aligned} \tag{2.4}$$

where $\boldsymbol{V}_i$ is nodal voltage of node i, I_j^* is the conjugate of the line current of line j and S_j is the apparent power flowing through line j.

In recent years, to tackle this issue of non-linearity, PMUs have provided a good alternative in the formulation of the State Estimation problem in transmission systems, since PMUs are able to provide voltage and phase angles separately [9]. In the absence of PMUs, different approaches have been attempted to tackle the non-linearity. Most ideas, which aim to achieve linearization, are focused on convexification of power flow equations through semidefinite and conic programming [10]. To this point, no known output was successful in implementing linear solution for the measurement function with multiple measurement inputs. The minimization process is what distinguishes the different optimization methods. Objective function J depends on **how** to minimize the error, thus every methods has a separate optimization goal.

2.4. State estimation methods

Generally, there are two types of State Estimations approaches. Static state estimation assumes no correlation and memory between individual time measurements. Typical static estimators are the weighted-least square method, least absolute value, maximized likelihood method etc. On the other hand, dynamic state estimators recursively track systems changes at any time instant to relate them with the next one in the future. Typical dynamic state estimators are Kalman filters and further extensions [11]. Generally, state estimation approaches depend on the network knowledge, measurement availability, device used and desired output.

The original State Estimation algorithm was developed as a static estimator by a weighted least square (WLS) minimization, disregarding correlation between different time stamps. Power systems have two operating conditions, quasi-steady state and transient state. The former, which is an approximation, assumes a non-dynamic character in the operating point, meaning that network changes are suggested as negligible to be taken into consideration. The latter, considers the sudden disturbances in the system, due to a fault or due to change in the operating point and capture the network's operating condition better. Implementing a state estimator based on transient state is impossible in cases of no real-time measurement availability. Generally, state estimators that operate on non-real-time measurements are based on the assumption of the quasi-steady state [12], which disregards sudden abrupt changes and system dynamics within the network. This is a good approximation for SCADA type of measurements. Essentially, every time-stamp is treated independently of other measurements within the same network and also independent of the previous or future ones for the same metering device.

The inherent flaws of static state estimation is the lack of dependence between two time-stamps, meaning that it's more prone to bad data, while the dynamic state estimation provides an acceptable value for the next time-stamp, which is a good way to filter bad data. The measurement availability provided for this thesis, which are SCADA type measurements, make static estimators the appropriate methods for the State Estimation formulation. As previously mentioned, the typical methods for State Estimation are the previously mentioned WLS, Least Absolute Value method (LAV) and Generalized Maximum likelihood method (GM), Schweppe Huber generalised M (SHGM) [4]. The most optimal method in terms of three statistical measures, namely bias, consistency and quality, was deemed the WLS, then WLAV and then SHGM [13]. The measurements provided in the state estimator were SCADA-type measurements. This means, that for a non-linear problem, the WLS method outperforms other methods, for SCADA type of measurements.

2.4.1. Non-linear WLS method

In the typical implementation of WLS for a power system introduced by Schweppe in 1970 [1]. The WLS estimator is a static estimator, since as previously mentioned, it does not consider previous or future dependency between either the measurements or the state variables. As such, it is formulated by solving an adjusted optimization problem of 2.3 in the following

manner:

$$min \quad (z - h(x))^T W (z - h(x)) \tag{2.5}$$

where the $W \in R^{m \times m}$ is the diagonal weight matrix, which is formed by the relative accuracy and trust levels of the measurements. Usually the weight matrix is based on the variance of the errors of the measurements. Three assumptions are needed, to successfully implement a WLS optimization:

- The noise of measurements must be a normal distribution and the error mean value must be zero.
- The correlation between measurements is zero, which means that no measurement has an impact on another measurement device outputs.
- Observability must be ensured, which typically means that the number of measurements at least exceeds the number of states. The mathematical representation of observability is $m \geq n$, only if there are sufficient independent equations for the measurement function h, thus $rank(h) = n$.

To solve the minimization problem stated, the first optimality condition must be satisfied [3], as shown in 2.6.

$$g(x) = \frac{\mathrm{d}J(x)}{\mathrm{d}x} = -H(x)^T \times W(z - h(x)) = 0 \tag{2.6}$$

In that sense, the objective function is derived into the Jacobian of the measurement function $h(x)$. As mentioned, power, voltage and current correlation is a non-linear relationship, as voltage and current in their complex forms must be used. This also means that the solving problem is non-linear, as this non-linearity is transferred within the measurement function.

The most common way to solve a non-linear optimization in an iterative procedure is the Gauss-Newton method. There is a lot of research and years of implementation of a non-linear WLS State Estimator. Since this is not a part of the thesis project, the procedure will be explained only briefly. Nevertheless, more details of the solving procedure are available in [1].

To find a solution, an iterative procedure begins, where the constraint function $g(x)$ is linearized around a specific points by Taylor expansion. This achieves a local linearization for every point. For every linearized point, a matrix called Gain matrix is calculated, $G(x)$, where

$$G(x) = \frac{du}{dt} = H(x)^T W H(x) \tag{2.7}$$

which is necessary to for the step calculation of the iterative procedure of

$$G(x_i)(x_{i+1} - x_i) = H(x)^T W \times (z - h(x)) \tag{2.8}$$

where x_i is defined as the i^{th} iteration of the Gauss-Newton algorithm, $H(x)$ is the linearized measurement function.

This procedure occurs until convergence is reached or number of iterations exceed the limits of the algorithm. Convergence is reached when the $x_{i+1} - x_i$ is less than the tolerance limit. The iterative procedure is computationally costly, but matrix decomposition methods like Cholesky or LU decomposition are helpful due to sparsity of the function matrix. The decomposition can be made through a decoupled approach for solving a state estimation problem [15]. Still, like every iterative procedure, speed of convergence and failure to converge are the main issues faced in every non-linear method.

2.4.2. Linear Weighted Least Square method

When solving a linear problem, the optimization requires no iterations to reach a solution. The measurement functions is transformed into a linear matrix and the equations of WLS method change to

$$Hx + e = z \tag{2.9}$$

where the size of H is still $m \times (2n-1)$. Therefore, unconstrained optimization problem aims to minimize the objective function of equation 2.10 :

$$min \quad J(x) = \frac{1}{2} r^T R^{-1} r \tag{2.10}$$

The solution of the aforementioned problem is transformed into solving the following problem:

$$G\hat{x} = H^T R^{-1} z \tag{2.11}$$

where G is the gain matrix

$$G = H^T R^{-1} H \tag{2.12}$$

Thus, the problem is transformed into the typical form of linear equations, where $A\hat{x} = b$. A solution occurs by matrix multiplication as shown in 2.13 :

$$\hat{x} = (H^* R^{-1} H)^{-1} H^* R^{-1} z \tag{2.13}$$

Therefore, the solution is reached by solving 2.13, disregarding the greatest obstacle of non-linear problems, which are iterative procedures and inability to converge within a pre-defined converge time. The goal is always to have a linear system, but the main issue thus far for the power systems was the inability to create linear equations that correlate states and active/reactive powers.

2.4.3. Least Absolute Value method

The LAV method is an optimization technique that, instead of minimizing the square of the errors, aims to minimize the absolute value of errors [16]. This difference puts the LAV into the robust estimators, which are estimators with high breakdown-point [17]. Breakdown-point is the minimum number of measurements that will lead to an incorrect SE solution. Essentially, LAV method does have an intrinsic bad data rejection ability, contrary to the weighted-least square method, which is more prone to bad data errors. Nevertheless, the disadvantages of this method lie in the much higher computation time and the vulnerability towards leverage measurements [18]. The LAV state estimation problem is formulated in equation 2.14:

$$\begin{aligned} &min \quad c^T |r| \\ s.t. \quad &z - h(x) = r \end{aligned} \tag{2.14}$$

where c is a vector of ones and r is the vector of measurement residuals. Another advantage of this method is using the minimum number of data points required, which can be equal to the number of states), thus inherent filtering of measurements considered as outliers, which WLS does not. The LAV problem is usually formulated as a Linear programming (LP) problem, by linearizing the system around the operating point. The LAV state estimation is formulated thus as a series of LP problems, thus still it's an iterative process.

Nevertheless, LAV is prone to errors in leverage measurements [17]. Leverage measurements are measurement points that impact significantly more the output of the State Estimation compared to others. Typical leverage points within a network are usually flows within short lines and injections at buses with large number of connections [19]. An occurrence of an error in those measurements can disregard the intrinsic capability of bad data rejection of LAV, thus leading to a bad estimation. Currently, the implementation of LAV is only considered in cases with PMUs, which are capable of providing voltage measurements in real and imaginary part, thus tackling the issue of severe iterative procedures that cripple the LAV optimization. This linear answer to an otherwise non-linear problem, has forced many to reapproach the state estimation under LAV methods [20],[21].

3

Mathematical formulation of the Linear State Estimation algorithm for low-voltage network

This chapter introduces the mathematical formulation of algorithms that will be used in this thesis project. The analysis entails the idea and the establishment of the algorithm, which is based on the Weighted Least Square (WLS) method, but is an expansion of the typical approach. This method, called also as Ultra-fast WLS method, will be the base for the State Estimation in the test cases. Moreover, an LAV method based on the expanded case will also be developed.

3.1. Linear Weighted Least Square State Estimation method for transmission network with high-redundancy

In this section the methodology behind the linear State Estimation based on WLS will be briefly analyzed, as initially introduced in [22]. The general idea is reformulating the state estimation problem into a two-step procedure, by smart reformulation of the states. This procedure requires knowledge of voltage and current measurements. In the first step, without the use of weights, and taking advantage of mathematical properties, the phase angles are calculated. In the second step, the voltage in its complex form is calculated, based on the phase angles calculated in the first step. This two-step procedure is vital to avoid any non-linearized occurrences in the measurement function matrix. Thus, a whole linearized state estimation is achieved by matrix multiplications.

To begin with, a distinction between complex numbers and magnitudes is necessary. To distinguish between real and complex numbers, the notation of **bold** represents complex numbers, whereas the rest represent real numbers. Since the measurement function is built on both complex and real numbers, notations to differentiate these values are deemed necessary.

As previously mentioned, voltage and current, when provided, are given only in magnitudes, since the measurements are assumed to be SCADA type. Thus, the voltage is calculated as:

$$\boldsymbol{E_a} = E_a e^{j\delta_{E_a}} \tag{3.1}$$

where $\boldsymbol{E_a}$ is the voltage in its complex form, E_a is the voltage magnitude provided and δ_{E_a} is the phase angle with reference to the slack voltage. Similar notation is used for the current in its complex form. The notation of $\boldsymbol{I}^{local}$ signifies only the complex form of current irrespective of the voltage reference. Thus in 3.2, this number signifies the output of current with respect to the power factor, but disregarding voltage phase angle. Two ways of calculating current

in its complex form, depending on if the amplitude is provided, which are shown in 3.2.

$$\boldsymbol{I_{ab}}^{local} = \begin{cases} I_{ab}e^{j\theta_{ab}} & \text{if } I_{ab} \text{ provided} \\ \dfrac{P_{ab} - iQ_{ab}}{V_a} & \text{if } I_{ab} \text{ is not provided} \end{cases} \tag{3.2}$$

where $\theta_{ab} = tg^{-1}(-\frac{Q_{ab}}{P_{ab}})$ is the power factor at the branch a-b, P_{ab} is the active power of the branch and Q_{ab} is the reactive power of the branch and V_a is the bus voltage magnitude of the associated branch.

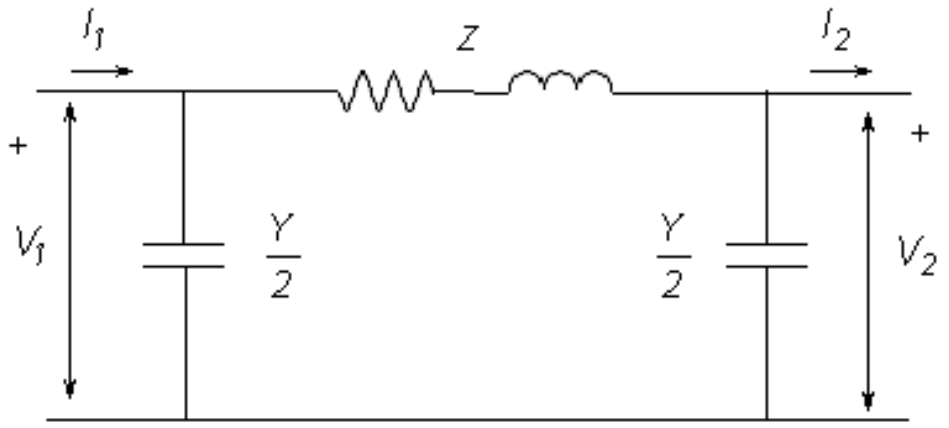


Figure 3.1: Single line pi-equivalent for distribution line [23]

Using the typical π equivalent shown in figure 3.1, the current flowing between two nodes can be calculated by using Kirchhoff law:

$$\boldsymbol{I_{ab}} = (\boldsymbol{Y_{ab}} + \frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_a} + (-\frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_b} \tag{3.3}$$

where $\boldsymbol{Y_{ab}}$ is the shunt admittance and $\boldsymbol{Z_{ab}}$ is the series admittance. This equation expresses the complex current at any bus a that goes through the branch $a-b$ in terms of state variables. To take into consideration the complex current with respect to the phase angle of the node, as shown in equation 3.2, the equation 3.3 is converted to the equation 3.4:

$$\boldsymbol{I_{ab}}^{local}e^{\delta_{E_a}} = (\boldsymbol{Y_{ab}} + \frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_a} + (-\frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_b} \tag{3.4}$$

Similar to the previous equation, the injected current between a nodes b, where $b \in L_a$. Injected current is calculated as:

$$\boldsymbol{I_{a,inj}}^{local}e^{\delta_{E_a}} = \sum_{b \in L_a} (\boldsymbol{Y_{ab}} + \frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_a} + (-\frac{1}{\boldsymbol{Z_{ab}}})\boldsymbol{E_b} \tag{3.5}$$

Taking advantage of the above formulations of equations only, a linearized system of equations occur in the form of

$$\boldsymbol{Hx} - e = \boldsymbol{z} \tag{3.6}$$

where:

$$\boldsymbol{H} = \begin{bmatrix} 1 & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & -E_2 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & -E_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ldots & \vdots & \vdots \\ \boldsymbol{Y_{12} + Z_{12}}^{-1} & \boldsymbol{-Z_{12}}^{-1} & \cdots & 0 & 0 & \cdots & 0 \\ 0 & \boldsymbol{Y_{2n} + Z_{2n}}^{-1} & \cdots & \boldsymbol{-Z_{2n}}^{-1} & \boldsymbol{-I_{2n}}^{local} & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & \boldsymbol{-Z_{n2}}^{-1} & \cdots & \boldsymbol{Y_{n2} + Z_{n2}}^{-1} & 0 & \cdots & \boldsymbol{-I_{n2}}^{local} \\ \boldsymbol{-Z_{n1}}^{-1} & 0 & \cdots & \boldsymbol{Y_{n1} + Z_{n1}}^{-1} & \boldsymbol{-I_{n1}}^{local} & \cdots & 0 \\ \boldsymbol{Y_{12} + Z_{12}}^{-1} + \boldsymbol{Y_{1n} + Z_{1n}}^{-1} & \boldsymbol{-Z_{12}}^{-1} & \cdots & \boldsymbol{-Z_{1n}}^{-1} & 0 & \cdots & 0 \\ \boldsymbol{-Z_{21}}^{-1} & \boldsymbol{Y_{12} + Z_{12}}^{-1} + \boldsymbol{Y_{2n} + Z_{2n}}^{-1} & \cdots & \boldsymbol{-Z_{12}}^{-1} & \boldsymbol{-I_{2,inj}}^{local} & \cdots & 0 \end{bmatrix}$$

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{E_1} \\ \boldsymbol{E_2} \\ \boldsymbol{E_3} \\ \vdots \\ \boldsymbol{E_n} \\ e^{\delta_2} \\ \vdots \\ e^{\delta_n} \end{bmatrix}$$

$$\boldsymbol{z} = \begin{bmatrix} E_1 \\ 0 \\ 0 \\ \vdots \\ \boldsymbol{I_{12}}^{local} \\ 0 \\ 0 \\ \vdots \\ 0 \\ 0 \\ \boldsymbol{I_{1,inj}}^{local} \\ 0 \end{bmatrix}$$

It's worth noting $\boldsymbol{H}$ is the size of $m \times (2N-1)$, thus $m \geq (2N-1)$ to avoid observability issues. Moreover, it can be obvious that H matrix could be partitioned in four different sub-matrices, as shown in equation 3.7.

$$\boldsymbol{H} = \begin{bmatrix} \boldsymbol{H_A} & \boldsymbol{H_B} \\ \boldsymbol{H_C} & \boldsymbol{H_D} \end{bmatrix} \tag{3.7}$$

where the size of H_A is $N \times N$, the size of H_B is $N \times N-1$, the size of H_C is $(m-N) \times N$, the size of H_D is $(m-N) \times N-1$.

This can help create some notations about the H matrix, where the **upper part of the matrix H** is in equation 3.8. Similarly, the **lower part of the matrix H** is in equation 3.9. This is done to differentiate that the upper part of the matrix H contains voltage-related measurements only, while the lower part of the H matrix contains current-related measurements.

$$H_{upper} = H_A H_B \tag{3.8}$$

$$H_{lower} = H_C H_D \tag{3.9}$$

Likewise, the **left part of the matrix H** is in equation 3.10 and the **right part of the matrix** is in equation 3.11. This differentiation is done to note that the left part of the matrix is topology related only, means that it doesn't change as long as the topology in the network doesn't change, while the right part of the matrix contains measurements, that of course change over time. Thus, an occurring advantage of this method, is that the left part of the

matrix for many time-stamps is calculated once at the beginning, and the is concatenated with the right part of the matrix, to avoid needless iterative procedures.

$$H_{left} = \begin{bmatrix} H_A \\ H_C \end{bmatrix} \tag{3.10}$$

$$H_{right} = \begin{bmatrix} H_B \\ H_D \end{bmatrix} \tag{3.11}$$

Effectively, what this H measurement function does, is change the way the states are represented in the network. The states still in a way represent the voltage magnitudes and phase angles, but the typical used states are altered from

$$x = [V_1 \quad V_2 \quad \dots \quad V_N \quad \delta_2 \quad \delta_3 \quad \dots \quad \delta_N]^T$$

to

$$x = [V_1 \quad \boldsymbol{V_2} \quad \dots \quad \boldsymbol{V_N} \quad e^{\delta_2} \quad e^{\delta_3} \quad \dots \quad e^{\delta_N}]^T \tag{3.12}$$

If the observability condition is true $m \geq (2N - 1)$, then the above set of equations has the solution of the closed form of

$$\hat{x} = (\boldsymbol{H}^*\boldsymbol{H})^{-1}\boldsymbol{H}^*\boldsymbol{z} \tag{3.13}$$

where the $\hat{x}$ is the least-squares estimate of x. To avoid direct matrix inversion of $m*m$ matrix, the paper suggests an alternative formulation. This suggestion simplifies the inversion by creating submatrices and thus inverting smaller matrices instead. The procedure is creating variables based on the two sets that constitute the equation 3.13, $\boldsymbol{H}^*\boldsymbol{H}$ and $\boldsymbol{H}^*\boldsymbol{z}$, which are

$$\boldsymbol{H}^*\boldsymbol{H} = \begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & D \end{bmatrix} \tag{3.14}$$

$$\boldsymbol{H}^*\boldsymbol{Z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \tag{3.15}$$

This separation's technique gain is threefold:

- The size of matrices for calculations is reduced. Despite the fact that more inversions and multiplications must take place, it is computationally more efficient to invert smaller matrices.
- The last N-1 columns (right side of H matrix, equation 3.11) of the matrix are highly sparse, having at most one non-zero element in every column in the upper and the lower part. Sparsity hastens matrix multiplication. Also, this will make D a diagonal matrix.
- Vector Z contains measurements associated with the slack bus only, since there are required no phase angle states to calculate currents associated with slack node. This means that, for every row, when a current appears on the last columns of the matrix H, there is a zero value in the same row of the Z vector. Therefore, all elements of z_2 submatrix in 3.15 are zero.

Taking advantage of the inversion formula from paper [22],

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{B} \\ \boldsymbol{C} & D \end{bmatrix}^{-1} = \begin{bmatrix} (\boldsymbol{A} - \boldsymbol{B}D^{-1}\boldsymbol{C})^{-1} & \boldsymbol{E} \\ -D^{-1}\boldsymbol{C}(\boldsymbol{A} - \boldsymbol{B}D^{-1}\boldsymbol{C})^{-1} & \boldsymbol{F} \end{bmatrix} \tag{3.16}$$

Here lies the importance of the third advantage of this separation method, since the z_2 is a row matrix of zeros, which means that the multiplication of the two created submatrices render E and F unimportant, since they are multiplied by zero elements of the same matrix. Thus, an unweighted solution occurs for

$$\hat{x} = \begin{bmatrix} (\boldsymbol{A} - \boldsymbol{B}D^{-1}\boldsymbol{C})^{-1}z_1 \\ -D^{-1}\boldsymbol{C}(\boldsymbol{A} - \boldsymbol{B}D^{-1}\boldsymbol{C})^{-1}z_1 \end{bmatrix} \tag{3.17}$$

An unweighted solution is provided by solving the above matrix, but since the wanted solution should be weighted, in this part only the phase angle is calculated, which is not affected by the weight values. So,

$$\begin{bmatrix} e^{j\hat{\delta_2}} \\ \vdots \\ e^{j\hat{\delta_n}} \end{bmatrix} = -D^{-1}\boldsymbol{C}(\boldsymbol{A} - \boldsymbol{B}\boldsymbol{D}^{-1}\boldsymbol{C})^{-1}z_1 \tag{3.18}$$

which is easily solved by either calculating the real or the imaginary part of the Euler identity

$$e^{j\delta} = \cos\delta + j\sin\delta$$

This helps calculate the bottom part of the state matrix, which includes the voltage phase angles. Following the calculation of voltage phase angles, voltage magnitudes are computed as a linear minimum-variance estimation, while taking into consideration the weights. Essentially, from the H matrix the only keeping the left side that are correlated with the voltages in their complex form and creating a matrix of the expected outcomes in the measurement matrix m, there is created a minimum variance of weighted least square method for m equations and N states, where $m \geq n$. Essentially the previous equation is transformed to this form:

$$\begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \cdots & \vdots \\ \boldsymbol{Y_{12}} + \boldsymbol{Z_{12}}^{-1} & -\boldsymbol{Z_{12}}^{-1} & \cdots & 0 \\ 0 & \boldsymbol{Y_{2n}} + \boldsymbol{Z_{2n}}^{-1} & \cdots & -\boldsymbol{Z_{2n}}^{-1} \\ \vdots & \vdots & \vdots & \vdots \\ 0 & -\boldsymbol{Z_{n2}}^{-1} & \cdots & \boldsymbol{Y_{n2}} + \boldsymbol{Z_{n2}}^{-1} \\ -\boldsymbol{Z_{n1}}^{-1} & 0 & \cdots & \boldsymbol{Y_{n1}} + \boldsymbol{Z_{n1}}^{-1} \\ \boldsymbol{Y_{12}} + \boldsymbol{Z_{12}}^{-1} + \boldsymbol{Y_{1n}} + \boldsymbol{Z_{1n}}^{-1} & -\boldsymbol{Z_{12}}^{-1} & \cdots & -\boldsymbol{Z_{1n}}^{-1} \\ -\boldsymbol{Z_{21}}^{-1} & \boldsymbol{Y_{12}} + \boldsymbol{Z_{12}}^{-1} + \boldsymbol{Y_{2n}} + \boldsymbol{Z_{2n}}^{-1} & \cdots & -\boldsymbol{Z_{2n}}^{-1} \end{pmatrix} * \begin{pmatrix} \boldsymbol{E_1} \\ \boldsymbol{E_2} \\ \boldsymbol{E_3} \\ \vdots \\ \boldsymbol{E_n} \end{pmatrix} = \begin{pmatrix} E_1 \\ E_2 e^{j\delta_2} \\ E_3 e^{j\delta_3} \\ \vdots \\ \boldsymbol{I_{12}}^{local} \\ -\boldsymbol{I_{2n}}^{local} e^{j\delta_2} \\ \vdots \\ \boldsymbol{I_{n2}}^{local} e^{j\delta_n} \\ \boldsymbol{I_{n1}}^{local} e^{j\delta_n} \\ -\boldsymbol{I_{1,inj}}^{local} \\ -\boldsymbol{I_{2,inj}}^{local} e^{j\delta_2} \end{pmatrix} \tag{3.19}$$

which is in matrix representation form as

$$\boldsymbol{T}\boldsymbol{y} + e = \boldsymbol{m} \tag{3.20}$$

thus the minimum variance of the above equation, following the similar procedure, but now inserting weights is

$$\hat{y} = (\boldsymbol{T}^* R^{-1} \boldsymbol{T})^{-1} \boldsymbol{T}^* R^{-1} \boldsymbol{m} \tag{3.21}$$

where R is the diagonal $m \times m$ covariance matrix. It the paper studied, the standard deviation of the measurement error for this specific measurement was considered for voltages, current and phase angles.

3.2. Limitations of adapting existing algorithm in the low-voltage grid

As mentioned, there are challenges of using an algorithm made for transmission networks to distribution ones. The main one is the unobservability of the network. This smart reformulation of the equations is able to be of great help reducing the computation costs for the calculation of State Estimation. This idea is really helpful to achieve linearity, but the problem still lies in using redundant and synchronized measurements, which are only available in the transmission network. The example provided at the studied paper [22] implements test cases for IEEE 118-bus Test System, 1354- and 9241-bus Pan-European High-Voltage Grids. In the first example, the number of state variables are approximately 235, while using over 600 measurements equations to achieve a result. Generally, all study cases examined are equipped with ratio of redundancy $\frac{measurements}{state}$ of 2.5 or higher.

Additionally, the measurements required for this linear approach are difficult to acquire in the low-voltage network. This method of linear state estimation is based on knowledge

of two network operating conditions in all points, voltage and current. For N nodes, the requirements are voltage knowledge of all N nodes, as well as knowledge *at least* about all active and reactive powers of lines or nodes (or a combination of them capable of creating sufficient information). Essentially, the observability condition is unaltered ($m \geq (2N-1)$), but now m is comprised of voltage and currents only. Due to the nature of low-voltage network, these requirements for such an algorithm are impossible, since for most cases only energy of smart meters at most could be provided. Overall, the list of limitations for adapting a linear state estimation in the low-voltage grid are:

- Multiple smart meters are used for the same nodes of the network. Thus, smart meter data can not be used directly.
- The energy of the smart meter devices is provided for a single phase and it is unknown which one, or for all three phases. Also, it must be converted to active power. Thus, reactive power can not be calculated.
- The examined algorithm relies on solving the equation stated in 3.6, which suggest knowledge of voltages and currents throughout the network. While current can be calculated, when voltage, active and reactive power are available, for low-voltage grids the assumption of voltage knowledge for every node is impossible.
- Different measurement types of different time-resolution of provided data must be used. This creates problems in the data coordination and cooperation.
- The distribution grid are highly complex networks. To add up, the cables used overall have low resistance and reactance values, due to shorter cable distance. Moreover, most DSOs provide network connections with fictitious nodes to represent fuses, breakers, switches and sectionalizers. These have minimal or zero values, which in turn cause issues in the matrix multiplications in the algorithm.
- The susceptibility to bad data is quite high, since the lack of concrete knowledge regarding the metering devices and their availability is questioned (since most metering devices are sent to their collector once per day).

3.3. Innovation required for distribution grid applications

This part contains the innovations and adjustments made to adapt the algorithm towards a distribution network. As previously established, there are many limitations to adapting the linear state estimation in distribution grid. To overcome these limitations, the following steps were taken.

- **Multiple smart meters are used for the same nodes of the network. Thus, smart meter data can not be used directly.** For this part, allocation of smart meter based on their node names is necessary. This is process which requires merging of smart meter data of within the same time-period. The merging requires aggregating the smart meter data for every node. This process also helps identifying which nodes are measurement nodes (GridEye or smart meter devices) and which are zero-injection nodes. The zero-injection nodes are potential nodes for removal in the network reduction process. Due to the complexity of the network, a two-step network reduction is implemented, where non-metered nodes and fictitious nodes are removed.
- **The energy of the smart meter devices is provided for a single phase and it is unknown which one, or for all three phases. Also, it must be converted to active power. Thus, reactive power can not be calculated.**
  The provided data for smart meters were three phase data, which is common in Switzerland. Every smart meter energy is transformed into power by using $P = \frac{E}{t}$, where P, E are the active power and energy of the smart meter respectively. The next step is allocation of smart meter per phase. In this part, the GridEye in the transformer, which contains all the network information aggregated, is of great use. Since GridEye device in

the transformer contains information about the active power for every phase, for every phase by the GridEye measurements we have a phase ratio that is calculated by:

$$r_p^{GE} = \frac{P_p^{GE}}{\sum_{p=1}^{3}(P_p^{GE})} \tag{3.22}$$

Assuming that the same ratio for phase p is kept for every smart meter device, the active power for every smart meter device i is calculated as:

$$P_p^i = P^i \times r_p^{GE} \tag{3.23}$$

To calculate the ratio for every smart meter device with respect to the total consumption of all smart meters for phase p and smart meter i, we use:

$$r_p^i = \frac{P_p^i}{\sum_{i=1}^{SM}(P_p^i)} \tag{3.24}$$

Following that, the smart meter power aggregation, based on the provided measurements, should ideally match the measurements provided by GridEye device in the transformer. Since this is never the case, the active power correction factor per phase is calculated:

$$E_p^{SM} = \frac{\sum_{i=1}^{SM}(P_p^i)}{P_p^{GE}} \tag{3.25}$$

Thus, for reactive power calculation for every phase p and smart meter i is:

$$Q_p^i = E_p^{SM} \times Q_p^{GE} \times r_p^i \tag{3.26}$$

Overall, based on the two measurements types provided, the reactive power is estimated for every smart meter device.

- **The examined algorithm relies on solving the equation stated in 3.6, which suggest knowledge of voltages and currents throughout the network. While current can be calculated, when voltage, active and reactive power are available, for low-voltage grids the assumption of voltage knowledge for every node is impossible.** Since voltage is mostly unavailable, the main contribution of this research is to make assumptions based on the provided voltage data, which are the GridEye devices. Nevertheless, the algorithm was implemented in a flexible way, where smart meter voltage could be provided and thus enhancing the outputs of the smart meters.

  In the case of lack of knowledge of smart meter's voltage, assumptions are made. The upper part of the H matrix, as shown in 3.8, contains information about voltages. The advantages of the WLS method is using weights, based on the degree of trust for every measurement. In the examined paper [22], the weights are used based on the measurement accuracy of the device used to provide that measurement. For this research, the upper part of the matrix will be assigned with such a weight value, that the algorithm will deem it totally unimportant. This weight value is 10^5 in order of magnitude less important than the other weight values. The values which were used, numerically, are provided in chapter 4.2, in table 4.1. What this achieves, is *faking* observability, without jeopardizing the outputs of the estimation.

  Moreover, the current calculation required is based from the equation:

$$\begin{aligned} \boldsymbol{I} &= \frac{P - iQ}{\boldsymbol{V}} \\ &= \frac{P - iQ}{|V|e^{j\delta}} \end{aligned} \tag{3.27}$$

  Based on the above equation, the current with respect to the reference, as it is required for the linear state estimation, can be calculated, if the voltage V is assigned. This part is

the major assumption of the algorithm, which, in the absence of information regarding the voltage of the smart meter, is using the voltage of the nearest GridEye device. What this means, is that the closest GridEye device is responsible for the value of current. Thus, a graph search algorithm was also created, identifying the closeness of smart meters and GridEye devices. Overall, the further a smart meter is from the GridEye device, the larger the error and the higher mismatch between the estimated and real current.

Moreover, since there are many zero-injection nodes in the network (all nodes not consuming or producing are zero-injection measurements), to reach observability, the zero-injection nodes are considered as measurements, where the output is 0.

- **Different measurement types of different time-resolution of provided data must be used. This creates problems in the data coordination and cooperation.**
  To tackle this problem, only common time-stamps between GridEye and smart meter devices are used. Since both measurements are provided with a specific time-stamps, the initial step is to match these time-stamps. It must be noted, that since energy is provided for the smart meters, the calculation of power requires correct conversion from energy to power based on the time-window of measurements.

- **Distribution grids are topologically simple, but otherwise highly complex networks. Specifically, the cables used overall have low resistance and reactance values, due to shorter cable distance. Moreover, most DSOs provide network connections with fictitious nodes to represent fuses, breakers, switches and sectionalizers. These have minimal or zero values, which in turn cause issues in the matrix multiplications in the algorithm.**
  To overcome this issue, and also enhance the algorithm, network reduction is necessary. Fictitious nodes are located in cabinets usually, containing many branches around them. In the network reduction process, nodes that should be kept or removed are identified. The fictitious nodes (which account for 40% of the total nodes in networks examined) must all be removed. To achieve this, a two-step procedure is required for the total network reduction, based on the Kron reduction method for eliminating nodes without loads [23]. In the first part of the procedure, the important nodes (smart meter and Grideye devices) that must be kept are identified, as well as nodes that are cross-sections and are necessary to be kept to retain the network's topological sense. In the second part, the network is checked to evaluate if there are any fictitious nodes left (lines with zero resistance). If they are, important nodes are reallocated to the closest node, to make the nodal removal possible. Thus, the fictitious nodes are removed.

- **The susceptibility to bad data is quite high, since the lack of concrete knowledge regarding the metering devices and their availability is questioned (since most metering devices are sent to their collector once per day)**
  The analysis of bad data is discussed in detail in Chapter 3.5.

All these procedures are necessary to adapt a transmission network state estimation to a distribution network one.

3.4. Least absolute value state estimation with linear inputs

The advantages between WLS and LAV methods have already been mentioned in the chapter 2. To define the algorithm's effectiveness and limitations, an LAV-based approach will also be implemented, as part of the thesis scientific questions. The advantage of this alternative method is that it does not require and additional information or implementation from the ultra fast WLS formulation previously analyzed. but the same inputs are given and the results are then directly compared. This helps to better understand the problems that lie in each method, the limitations, as well as the advantages. Similarly to the WLS and any optimization method, the goal is to minimize the error between the estimated and the measured value. Using the same notations as the WLS method

$$\boldsymbol{Hx} - e = \boldsymbol{z} \tag{3.28}$$

where still $\boldsymbol{H}$ is the measurement function, $\boldsymbol{x}$ is the state matrix and $\boldsymbol{z}$ is the measurement matrix related only to the slack node.

It must be noted to this point, that the LAV method requires only the measurement function H and the vector containing the slack measurements z, as shown in equation 3.6. Thus, same inputs are provided as the WLS method, but the stop step process is not used. So, other than defining the constraints of the algorithm, no further actions are necessary for the data preparation process.

As mentioned, the goal is to minimize the absolute value of the errors, instead of the square of the errors. In this case the measurement function $\boldsymbol{H}$ and states are complex numbers, which is harder to deal for optimization. To overcome this issue, the optimization is separated into real and imaginary part. Taking into consideration the initial approach for the optimization:

$$\begin{aligned} min \quad & J(\boldsymbol{z} - \boldsymbol{h}(\boldsymbol{x})) \\ s.t. \quad & g(x) = 0 \end{aligned} \tag{3.29}$$

where g are the constraint functions, the problem is translated into the following:

$$min \quad c^T(R_{real} + R_{imag}) \tag{3.30}$$

but the constraint functions are now

$$s.t. \begin{cases} H_{real}x_{real} - H_{imag}x_{imag} - z_{real} & \leq R_{real} \\ -(H_{real}x_{real} - H_{imag}x_{imag} - z_{real}) & \leq R_{real} \\ 0 & \leq R_{real} \\ H_{real}x_{imag} + H_{imag}x_{real}z_{imag} & \leq R_{imag} \\ -(H_{real}x_{imag} + H_{imag}x_{real})z_{real}imag & \leq R_{imag} \\ 0 & \leq R_{imag} \end{cases} \tag{3.31}$$

where R_{real} and R_{imag} are the $m\times1$ sized residuals (or errors) of the optimization's objective, c^T is a $1 \times m$ sized matrix of ones, H_{real} and H_{imag} are the real and imaginary parts of the measurement function matrix $\boldsymbol{H}$, x_{real} and x_{imag} are the real and imaginary parts of the states of vector $\boldsymbol{x}$.

3.5. Bad data management

A great obstacle against optimization is the inclusion of bad data as inputs. Especially when dealing with smart meters, the probability of bad data is higher. Inherently no optimization method has a bad data detector, but methods treat bad data differently than others. As previously mentioned, WLS optimization is more prone to bad data, as the optimization's goal is to minimize the square of errors. Thus, it is expected, that an LAV optimization is better in that regard, since the optimization aims to minimize the absolute value of the errors.

For a successful bad data analysis, 3 steps must be completed, which are:

- Bad data detection
- Bad data identification
- Bad data correction

The first one, which is bad data detection, aims only to identify if there are bad data in the given inputs. The second step's goal is to identify which of the provided measurements was deemed as bad data, and the last part aims to correct the data given, if possible. Obviously, the lower data redundancy, the more difficult it is to complete the above three-parts of bad data. It must be mentioned that for all static state estimation algorithms, all bad data methods operate after the state estimation is completed. This means that a successful bad data detection requires a further re-run of the state estimation, to validate the outputs.

Typically, there are two methods capable of detecting and identifying bad data in static estimators, which are the $\boldsymbol{x^2}$ test (or chi-squared test) and the **normalized residual** test.

Nevertheless, the x^2 test, despite the low-computation cost, can not identify with precision the measurement containing the bad data [24]. For this reason, most bad data detection algorithms opt for the normalized residual test method. Based on this method, the goal is to calculate the residuals under a normalized scope (to make sense with respect to the weights used). Then, if the largest one of the residuals is over a threshold manually chosen, this measurement corresponding to this residual is deemed as bad data, meaning it should be either removed or altered. An obvious disadvantage to this method is the manually decided threshold for the largest residual to remove, as well as the inability to cope if the measurement providing the error is critical (if the measurement is removed then there is lack of observability). Generally, the residuals are calculated as:

$$\boldsymbol{r} = \boldsymbol{z} - \boldsymbol{H}(\hat{\boldsymbol{x}}) \tag{3.32}$$

where $\hat{x}$ represents the outcomes of the state estimation. Since the measurement function $\boldsymbol{H}$ is linear, this means that combining output equation 3.21 and 3.32, the residual vector is calculated as shown in [25]:

$$\begin{aligned} \boldsymbol{r} &= \hat{\boldsymbol{z}} - \boldsymbol{z} \\ &= (I - \boldsymbol{H}\boldsymbol{G}^{-1}\boldsymbol{H}^T R^{-1})\boldsymbol{z} \\ &= (I - \boldsymbol{H}\boldsymbol{G}^{-1}\boldsymbol{H}^T R^{-1})(\boldsymbol{H}\boldsymbol{x} + \boldsymbol{e}) \\ &= (I - \boldsymbol{H}\boldsymbol{G}^{-1}\boldsymbol{H}^T R^{-1})\boldsymbol{e} \\ &= \boldsymbol{S}\boldsymbol{e} \end{aligned} \tag{3.33}$$

where $\boldsymbol{S}$ is a matrix defined for simplicity and $\boldsymbol{G}$ is the gain matrix, which was directly used in the state estimation. The second to last step simplified the equation due to $\boldsymbol{S}\boldsymbol{H} = 0$, thus:

$$\begin{aligned} \boldsymbol{S} &= I - \boldsymbol{H}\boldsymbol{G}^{-1}\boldsymbol{H}^T R^{-1} \\ \boldsymbol{G} &= \boldsymbol{H}^T R^{-1}\boldsymbol{H} \end{aligned} \tag{3.34}$$

Since the error assumption was a normal distribution (for the WLS method), the residual covariance matrix is calculated as

$$\begin{aligned} \Omega &= E(\boldsymbol{r}\boldsymbol{r}^T) \\ &= \boldsymbol{S}E(ee^T)\boldsymbol{S}^T \\ &= \boldsymbol{S}R\boldsymbol{S}^T \\ &= \boldsymbol{S}R \end{aligned} \tag{3.35}$$

The last step is verified by substituting 3.34 for S. Thus, the calculation of the normalized residuals is essentially affected by the standard deviations of the measurements residuals inside the Ω diagonal elements. This means that the normalized residuals are calculated as:

$$\boldsymbol{r}_i^N = \frac{R_{ii}}{\sqrt{\Omega_{ii}}}\boldsymbol{r}_i = \frac{R_{ii}}{\sqrt{S_{ii}R_{ii}}} \tag{3.36}$$

Thus, the normalized residuals are affected by the square root of the Ω and the residual value.

At this point, the threshold is established. This means that if the largest normalized residual $\boldsymbol{r}_i^N > t_{threshold}$, then the value is deemed as bad data. Nevertheless, due to imperfect matches of the state estimation, there will always be a largest normalized residual. The threshold is necessary to ensure that after a value of error in the normalized residual values, there could be bad data. This value is manually defined, based on specific network operating points and characteristics. To make it more clear, without a threshold, there will always be a triggered algorithm of bad data identification and correction. To overcome this, usually the threshold is adapted to the needs of the estimator. With this part, the detection and identification processes are completed.

The next and final step is to decide whether to correct the bad data measurement, or to completely remove it. The consequence of removing the bad data is reduced redundancy and measurement unavailability on the node or branch of removal. Thus, no information

can be provided at that node, which is at the expense of the algorithm as a whole. On the other hand, bad data correction can carry the inherent errors of the estimator inside to the changed value. Nevertheless, it is deemed better to change the value of the bad data, since it does not only provide information about the node that had bad data, but also it does not change the data structure. The bad data correction process is provided as shown in [26] as:

$$\begin{aligned} \boldsymbol{z}_{corr,i} &= \boldsymbol{z}_{bad,i} - \frac{\boldsymbol{r}_i}{S_{ii}} \\ &= \boldsymbol{z}_{bad,i} - \frac{R_{ii}}{\Omega_{ii}}\boldsymbol{r}_i \end{aligned} \tag{3.37}$$

To overcome the issue of lack of redundancy in the low-voltage networks, historical data are necessary for the smart meters. In bad data analysis under high redundancy, no further information are required, and all steps in the detection, identification and correct are as explained above. For the case of low-voltage grid, as the algorithm is destined for such a network, the voltage assumptions and the low-weights initialized on voltages, make the currents the only information available at the disposal of the estimator. As such, the estimator can not sometimes define which node could potentially have bad data. To this end, historical data can estimate more probable nodes that could potentially be described as bad data. Without historical data, the algorithm can understand the part of the network (under which GridEye device) bad data potentially occur, but can not identify the correct node. Thus, if the threshold is exceeded, the historical data prove helpful to the assumptions created for the linear low-voltage state estimation.

4

Experimental Results

4.1. Information regarding the results and test cases

In this section, basic information will be provided regarding the simulation processes, the test cases, the validation process and the methodology. There are necessary for the in depth analysis of the results.

4.1.1. Validation and error reference

Although the trust in GridEye device measurement accuracy is quite high, a prior power flow analysis is necessary in every algorithm evaluating procedure. First of all, power flow is an established tool for acquiring network operating conditions. Thus, evaluating a new algorithm's performance requires a established method as reference, capable of validating the outputs results. So, power flow provides a comparison possibility, that is otherwise impossible.

For the power flow calculation, the network parameter information, slack bus voltage and power injections of all nodes were required. The network parameter information are provided by the DSO, and the goal is to create an admittance matrix, which is a matrix containing information regarding the nodal connections and admittances [23]. Furthermore, active and reactive injections of smart meters were used, and their calculation was discussed in 3.3. Nodes that were not smart meters had 0 power. Slack voltage is calculated by combining the GridEye device measurements of the transformer (current and voltage) and network topology.

Moreover, device and topology errors exist in real networks. Inherent device errors and topology provided errors are not uncommon, when provided by DSOs. These errors can be hard or impossible to trace and could create differences between the outputs and the inputs of the state estimation. To avoid this issue, power flow outputs are used as inputs, **in the form that the measurement device would provide them**. That way, power flow provides a generalized solution for the problem, disregarding the issue of wrongly provided data. Effectively, instead of providing the algorithm directly with metered data, the equivalent output of power flow, but in the same measurement type as the and with a device-specific signature, will be provided. Thus, the algorithm can be evaluated without external factor jeopardizing its performance.

4.1.2. Methodology

As mentioned in Chapter 3.3, many processes are required for running a state estimation, like data preparation, network creation, network reduction et.c. The general idea of these functions are shown in the figure 4.1. To get the real network sense, noise is inserted, in the test cases that it is possible (due to computational burden noise ensues). The noise is equivalent to the level of error estimated by the device measured. The noise equivalents for the used devices are:

- 0.1% for voltage, when provided by GridEye device

- 1% for current, when provided by GridEye device
- 3% for active power, when provided by smart meter device

The noisy inputs will be provided to the estimator, whose outputs will be evaluated with respect inputs provided. It must be mentioned, that since noise increases the computational cost of the algorithm, only specific test scenarios were evaluated by noise injection.

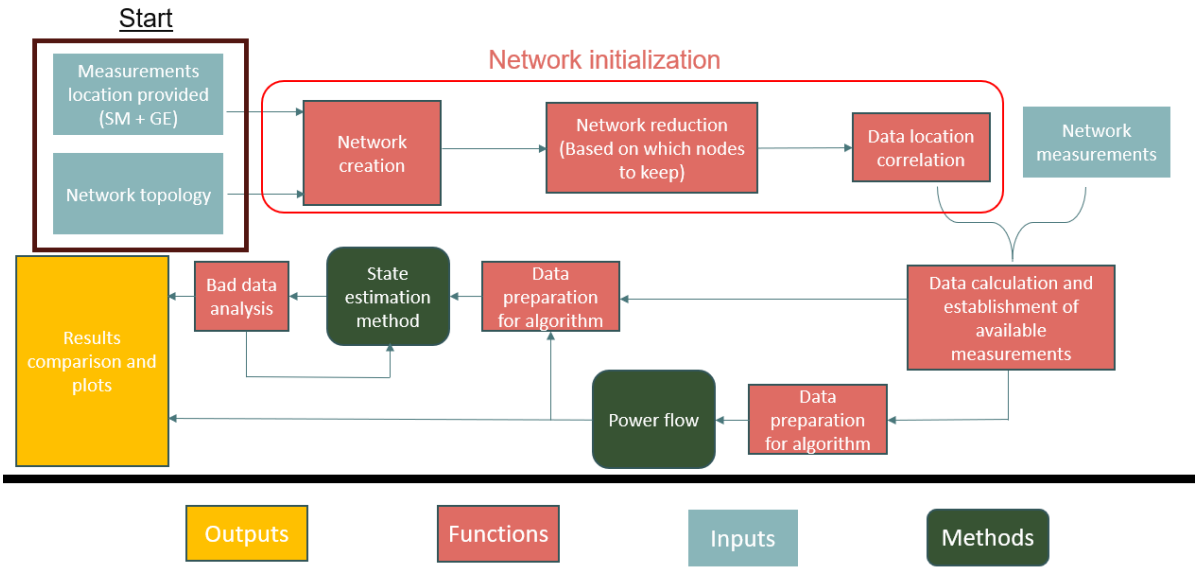


Figure 4.1: Simplified methodology of State Estimation inputs.

Also, bad data analysis, was only considered for one test case, and is not implemented in every test scenario. Bad data is discussed only in 4.3.7. The bad data methodology as a block diagram is shown in figure 4.2.

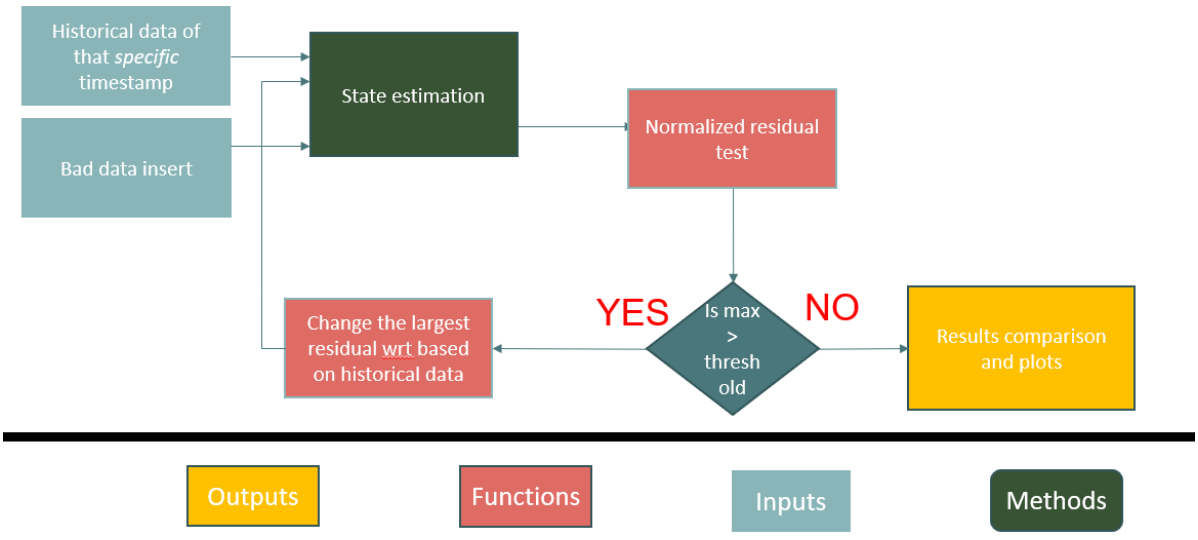


Figure 4.2: Bad data block diagram.

4.1.3. Test cases

The first test case consists of a reduced network with only GridEye devices. This helped acquire a better sense of the algorithm in a smaller network with sufficient measurements

and high redundancy to check the validity of the outputs. Thus, in the first case, no smart meter measurements were used, and the outputs were evaluated both with respect to both power flow outputs and other GridEye devices within the same network.

In the second test case, an existing large-scale low-voltage network was provided as an input. The initially provided network was unreduced. Within this network there were GridEye devices and smart meters.

Different scenarios are tested for the test cases. Unless mentioned specifically, noise is not included. These cases are namely:

- Test case 1: Network with GridEye devices.
    1. **Network with only GridEye devices, only WLS method**. This is the first test case, where only the WLS method was implemented. Noise was included.
    2. **Use of only 2 devices, both methods used**. For this scenario, both methods were implemented. To reduce the redundancy, only two GridEye devices were used. Noise was included.
- Test case 2: Network with GridEye and smart meter devices.
    1. **Assumed voltage knowledge, both methods used**. In this scenario, voltage of smart meters were "assumed" to be known, based on the data provided by the power flow outputs. This is to validate whether the algorithm can operate in high-redundancy, to then find the errors for lower redundancies.
    2. **True measurement availability, both methods used**. In this scenario, everything that is available based on the measurements provided is used. This would be the case base, based on the available information.
    3. **Effect of number GridEye devices in noiseless scenario, WLS method**. In this scenario, the comparison was made between using a single or three GridEye devices in the network.
    4. **Effect of number GridEye devices in noisy scenario, WLS method**. In this scenario, the comparison was made between using a single or three GridEye devices in the network. This is the scenario where noise is included for all devices, to validate the algorithm's adaptability to noise.
    5. **Bad data injection in smart meters, both methods used**. In this scenario, the base case (containing all three GridEye devices) was used. At this point, in every SM device bad data are injected or different values. Only one bad data injection is created (not multiple bad data simultaneously).

Last but not least, the error is calculated as

$$e = x_{input} - x_{output} \tag{4.1}$$

where input describes the input for the specific node or branch of the state estimation, and output describes the output of the same node or branch of the state estimation.

4.1.4. Simulation

To evaluate the effectiveness of the developed algorithm, simulations were conducted in Matlab, version 2015b [27]. The LAV was implemented based on YALMIP-Matlab implementation [28]. The solver was decided to be CPLEX solver. For WLS, the method was created from scratch. The whole algorithm was developed within the Matlab environment and the measurements provided as inputs were provided from existing networks both within Switzerland. Every function used was created by the author of the report.

4.2. Test case 1: Network with only GridEye devices

The first case consists in a real-network within Switzerland, where a week worth of measurements were provided from the end of August. The initial topology consists of **253 nodes**

and 252 branches, as it is seen in 4.3. The MV/LV transformer of this network is 250KVA. This network has a high penetration of renewable energy, which means higher fluctuations in voltage between day, night, sunny or cloudy time-stamps. In this network, as it's the standard working case, network reduction was established to only work with the GridEye devices, since in this network no smart meters we available. Thus, the network where the final estimation algorithm will be tested consists only of the GridEye devices, which are 4, and the transformer. Thus, in this discussed network, there will be noted 5 nodes and 4 branches, which are only correlated to GridEye measurements, as shown in the figure 4.4.

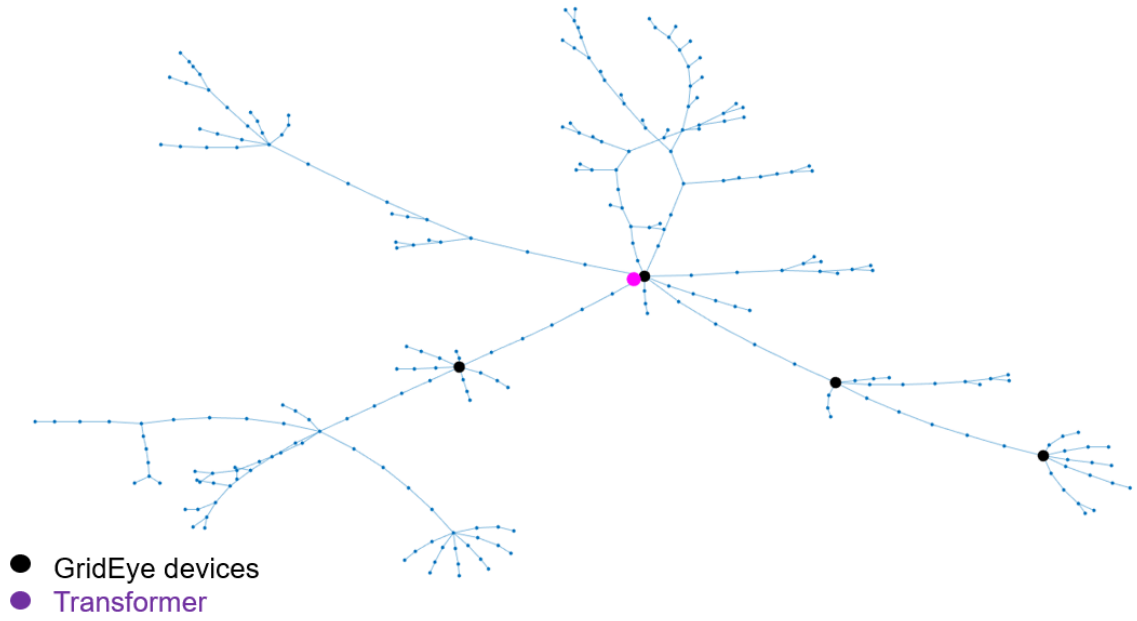


Figure 4.3: Unreduced network for case 1.

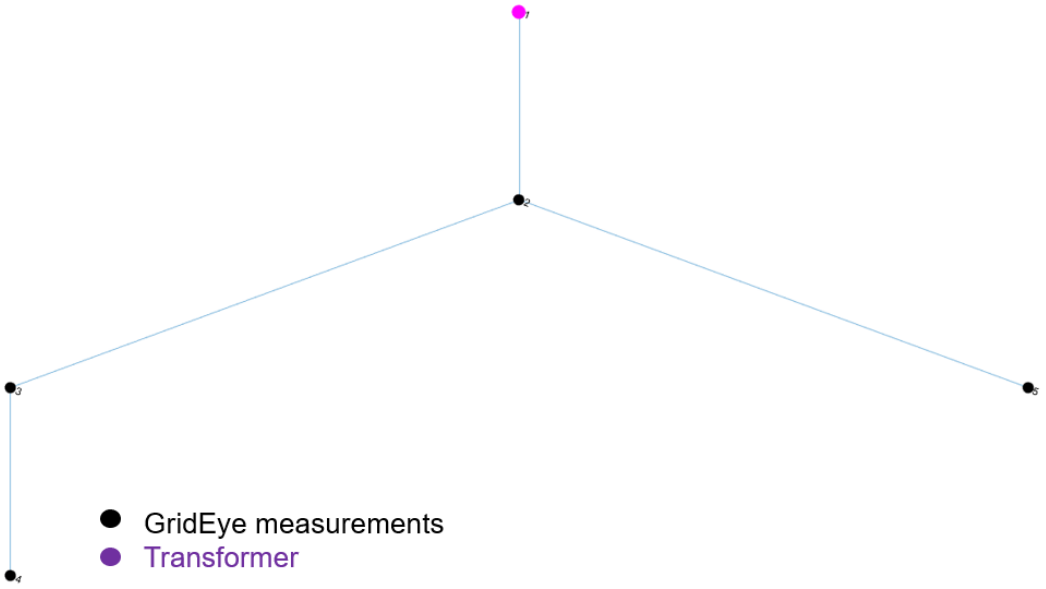


Figure 4.4: Reduced network for case 1.

GridEye measurements and network topology characteristics were provided for this case. It should be noted that since this is the reduced network, the network was treated as such network was the initial case, meaning that the state estimation was created only for the aforementioned nodes. What this achieved in higher knowledge within the network, which is an important step in creating a state estimation algorithm from scratch. Thus, this case is a case of high redundancy, which theoretically should enhance the State estimation outputs.

Furthermore, from the provided GridEye data measurements, only voltage, branch current, active and reactive power of branches were used for every phase.

To further expand the algorithm, another sub-case is examined where only 2 GridEye devices are used. By taking advantage of GridEye's ability to measure all the phases for up to 3 different branches, branch currents can be provided, while lacking the nodal voltage of the unmeasured nodes. This is used to start expanding the algorithm without complete network knowledge. The base test cases for this are visualized in the figure 4.5, where the branches measured are also shown:

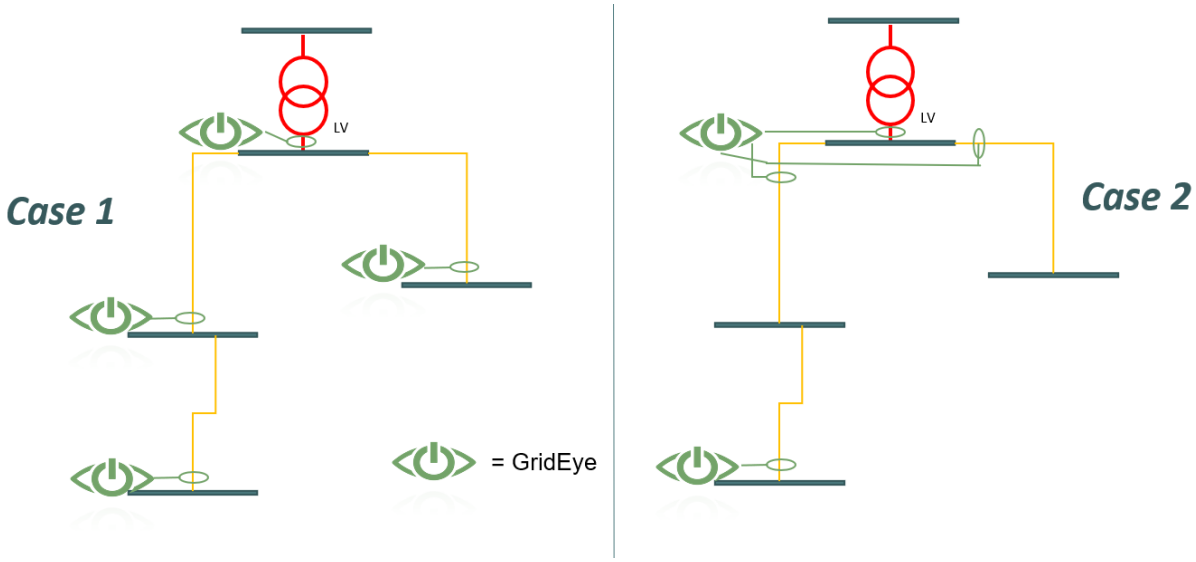


Figure 4.5: One-line diagram of the main test case divided into two sub-cases.

4.2.1. Test case 1 - Scenario 1: Use of all GridEye devices with only WLS method

In the first part of the test, the Ultra fast WLS was only implemented with weights for every device, since the trust of every GridEye device is the same. In this test scenario, all GridEye devices are included, as shown in the lest test scenario of the figure 4.5. The LAV method was not implemented for this scenario. The outputs of the State Estimation are compared with the power flow outputs, and the noise that is inserted is with regard to the error of the GridEye device (0.1% for voltage measurement and 1% for current measurement). In this scenario 1000 noise cases with normal distribution and noise levels mentioned above are implemented. For this reason only the visual representation of the state estimation outputs for the voltage, current and phase angle outputs of transformer node and transformer branch are provided. The other results are too similar to visualize. The outcomes are shown in the figures 4.6, 4.7 and 4.8.

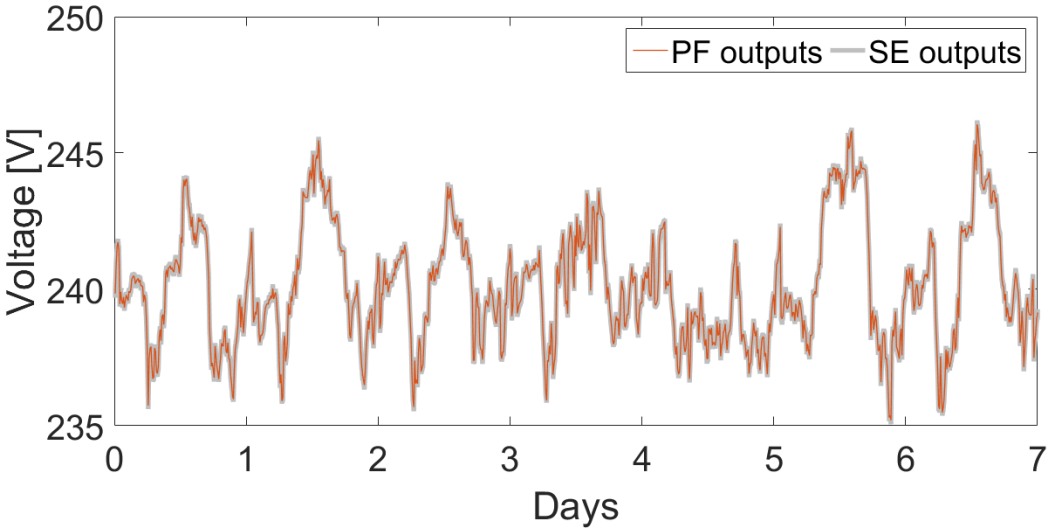


Figure 4.6: Voltage magnitude output for SE for all the noise cases for a week.

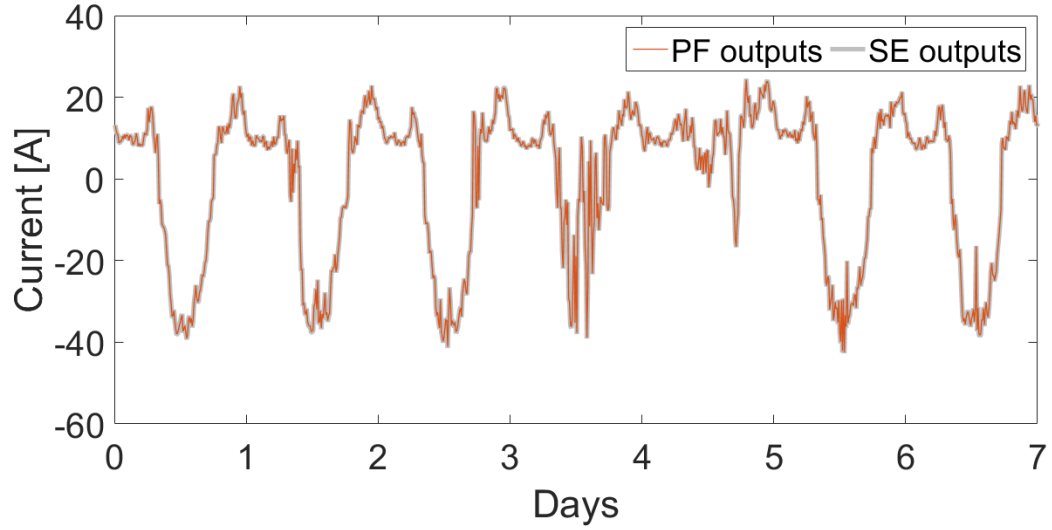


Figure 4.7: Current amplitude output for SE for all the noise cases for a week.

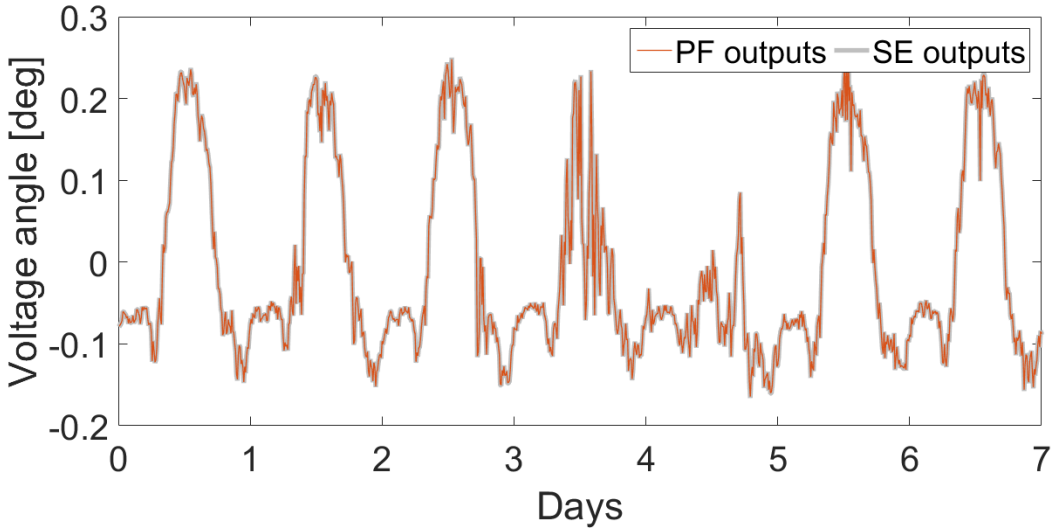
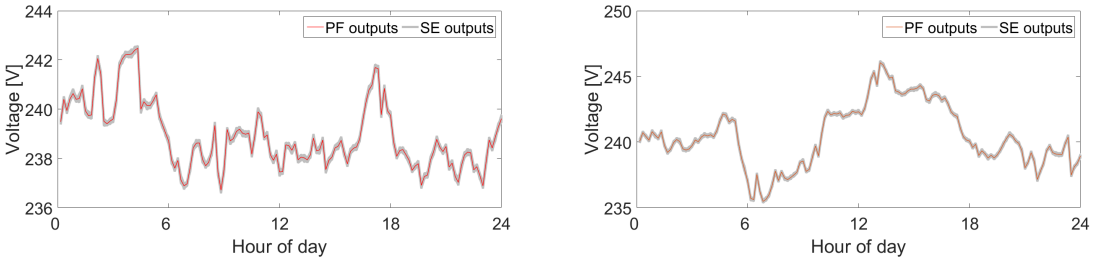


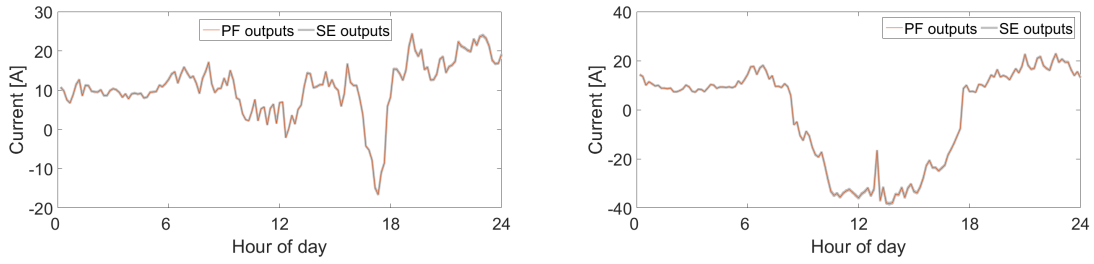
Figure 4.8: Voltage phase angle output for SE for all the noise cases for a week.

As it's obvious from the above graphs, the state estimation successfully estimates the states based on the provided equations. For deeper analysis, it is obvious that the day 4 represents a cloudy day, where day 6 represents a sunny day. The error variations within the mentioned days for voltage magnitude, current amplitude and phase angle are depicted in the figures 4.9, 4.10 and 4.11.



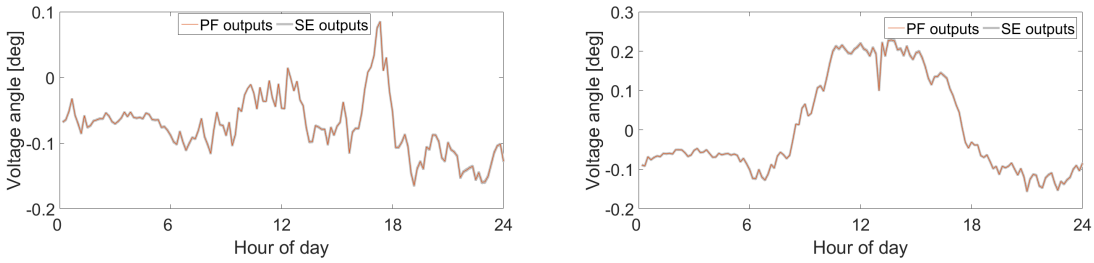
(a) Cloudy day. (b) Sunny day.

Figure 4.9: Voltage magnitude differences for a cloud to a sunny day.



(a) Cloudy day. (b) Sunny day.

Figure 4.10: Current amplitude differences for a cloud to a sunny day.



(a) Cloudy day. (b) Sunny day.

Figure 4.11: Voltage phase angle differences for a cloud to a sunny day.

Based on these figures, it's quite obvious that the State Estimator based on the Ultra-fast WLS method successfully captures the network measurements for noise cases, in all scenarios (with high and low DG-penetration). To evaluate the errors, histograms are provided in figure 4.12, 4.13 and 4.14

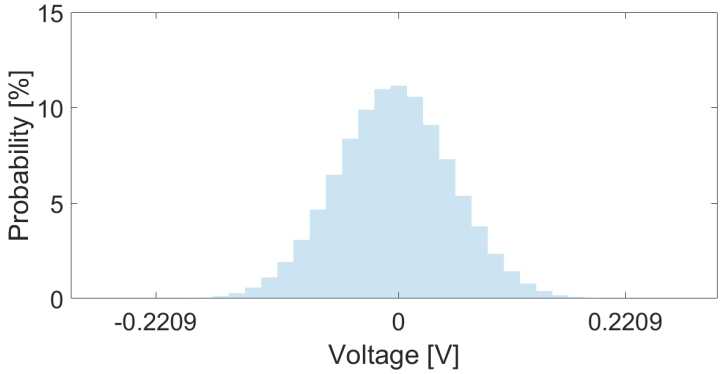


Figure 4.12: Histogram of voltage magnitude errors for SE for all nodes and all time-stamps due to noise

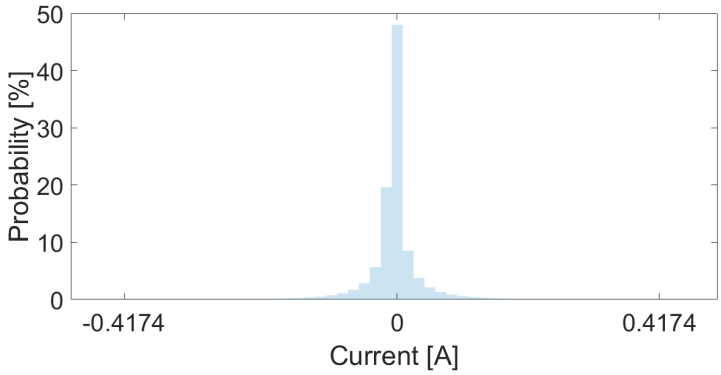


Figure 4.13: Histogram of current amplitude errors for SE for all nodes and all time-stamps due to noise

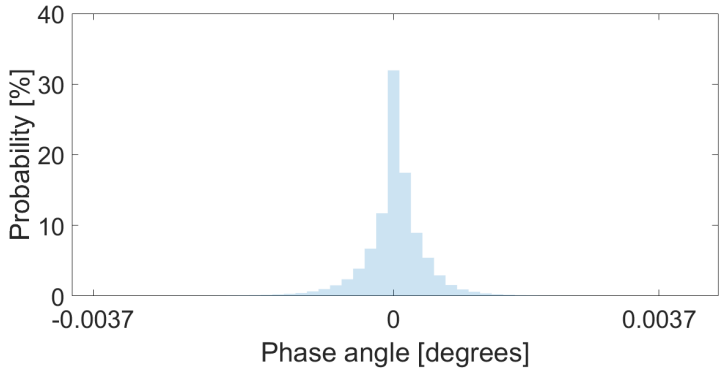


Figure 4.14: Histogram of phase angle errors for SE for all nodes and all time-stamps due to noise

The errors of the outputs are following a normal distribution, as the noise provided. The mean error revolves around zero, while the maximum error for voltage is 0.2209V ($\approx 0.1\% \times 240V$) and for current is 0.4174A ($\approx 1\% \times 41.7A$), which means that the outputs are equal to the injected noise. Thus, low-errors occur for this test case.

4.2.2. Test case 1 - Scenario 2: Use of only 2 devices, both methods used

This test scenario uses less GridEye devices, as shown in the right part of the figure 4.5. In this test scenario also the LAV method was included included. Since the LAV and WLS methods outputs are too similar, only plots with direct comparison (maximum errors, histograms) between the methods are considered. Higher level analysis is only done for this test case scenario, due to the similarity between the methods and the low numerical error. While maximum errors and histograms are not robust statistical ways or representing errors, maximum errors and histograms are sufficient for this test scenario. Moreover, the comparison is made only with respect to the unknown branches and nodes, which can be noted in figure 4.5. Thus, first, the error distribution within the nodes and branches are plotted in figures 4.15, 4.16, 4.17

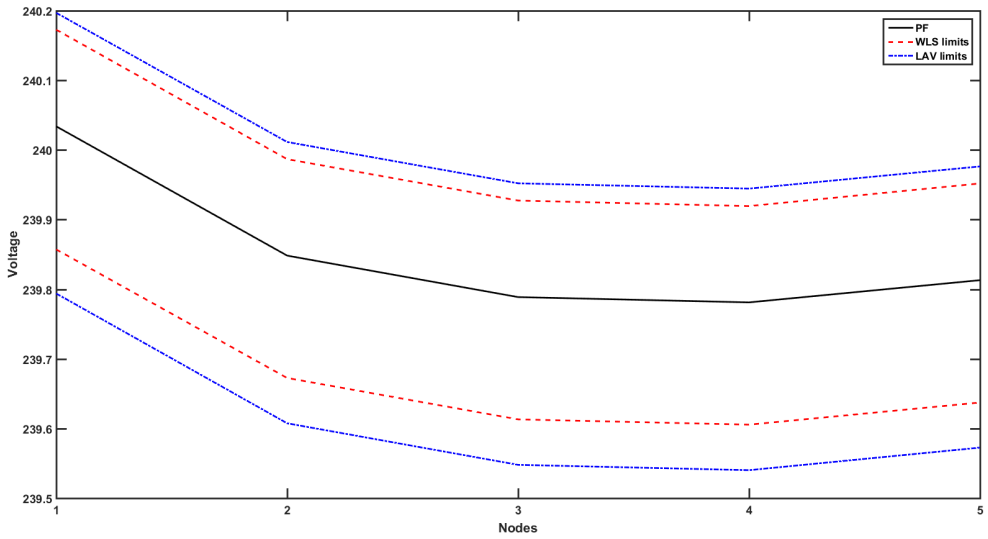


Figure 4.15: Maximum voltage magnitude error distributed for every node for all noise cases

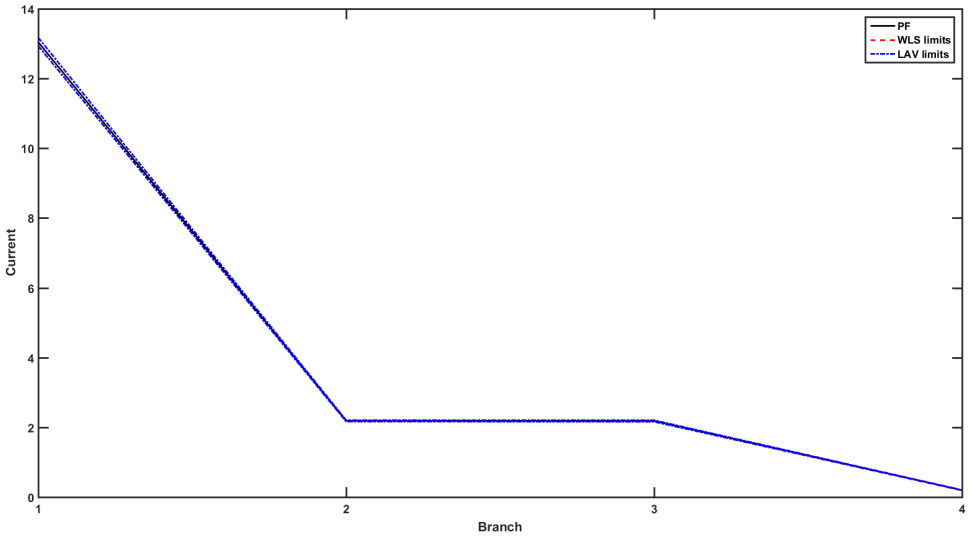


Figure 4.16: Maximum current amplitude error distributed for every branch for all noise cases

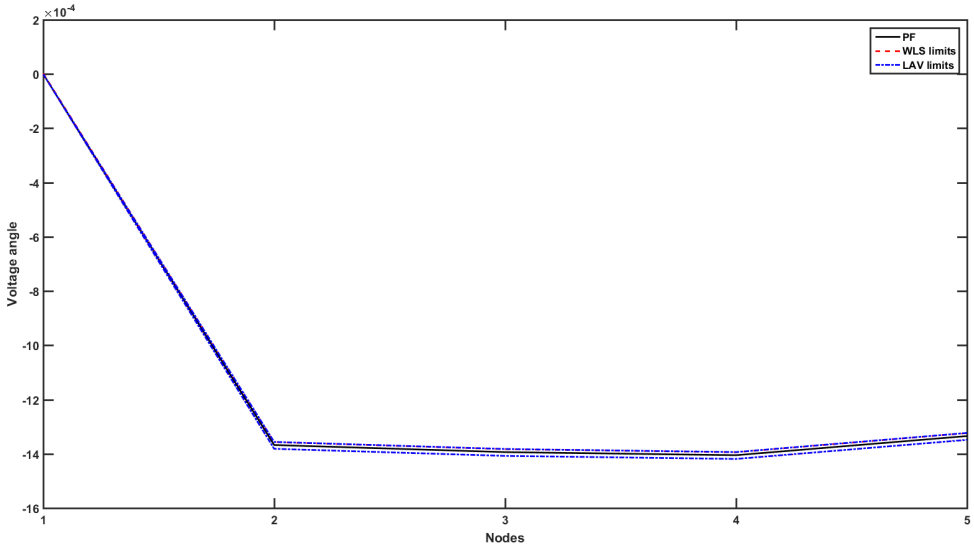


Figure 4.17: Maximum voltage phase angle error distributed for every node for all noise cases

Moreover the histogram of the previously mentioned plots, for the direct comparison of the errors is shown in figures 4.18, 4.19 and 4.20. It must be noted that these histograms are not the same with the histograms provided for the WLS method in the figures 4.12, 4.13 and 4.14, because only the voltages and currents of unknown nodes and branches respectively are compared.

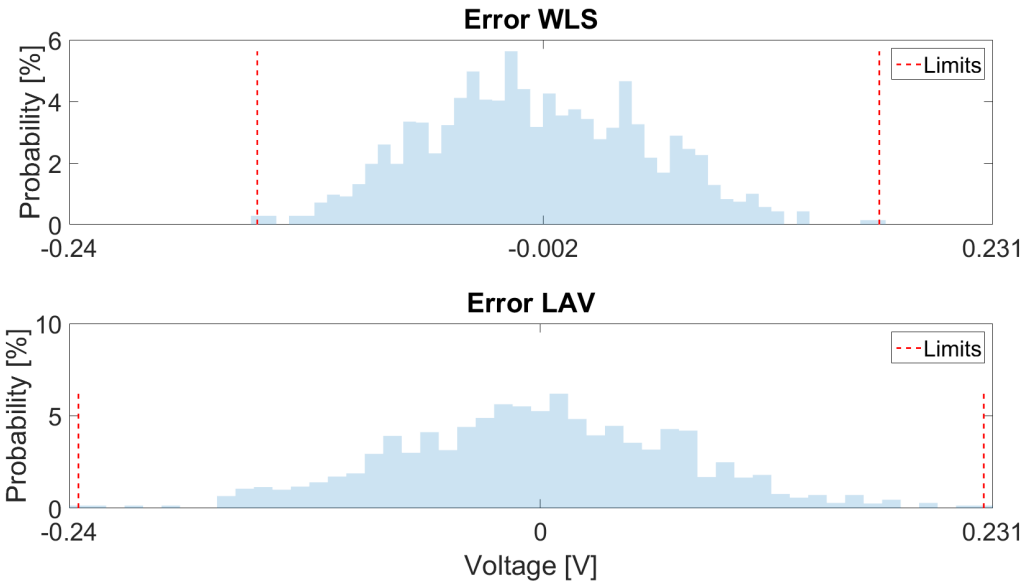


Figure 4.18: Histogram comparison for voltage magnitude

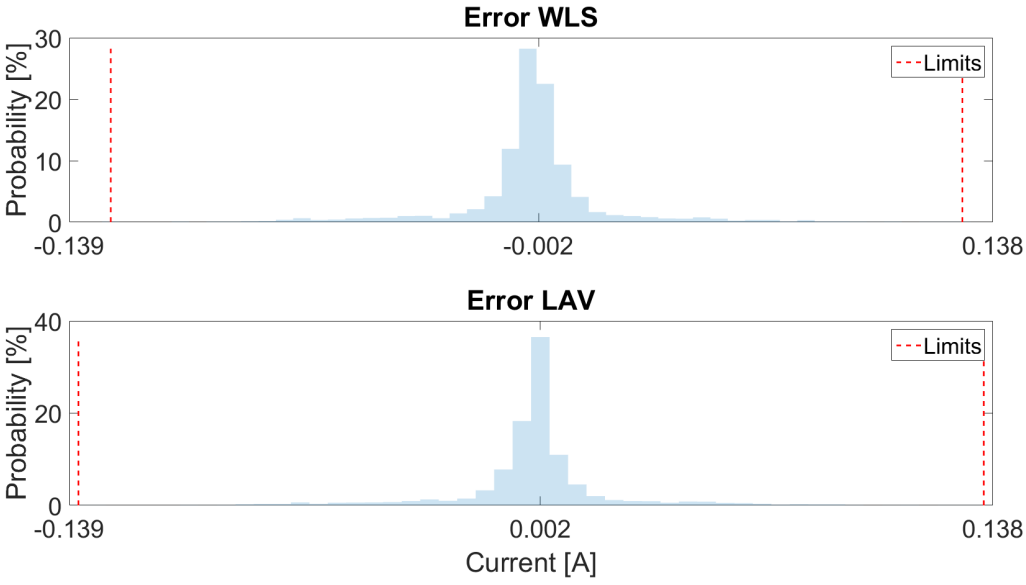


Figure 4.19: Histogram comparison for current amplitude

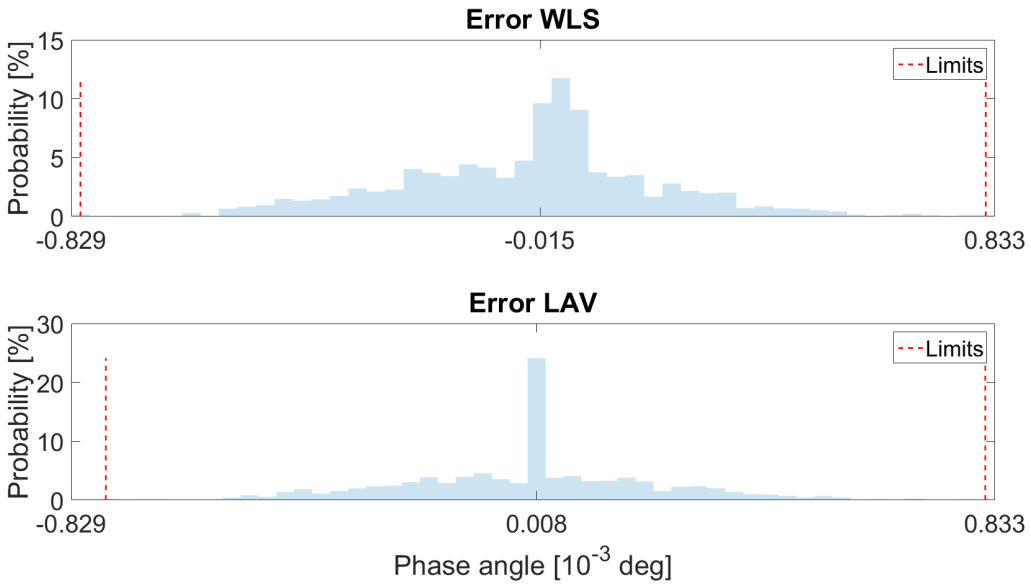


Figure 4.20: Histogram comparison for voltage phase angle

The difference between the comparison figures (4.18, 4.19, 4.20 and 4.12, 4.13, 4.14) is due to running more noise scenarios for the case of only having the WLS method (50000 in the former, 1000 in the latter), as it's faster. Thus the outputs get more normalized, whereas in the inclusion of LAV it's highly computationally costly to do likewise.

For this case, WLS produces lower errors for voltage magnitudes and current amplitudes for the same inputs. Nevertheless, phase angle seems to produce better outputs for the phase angle. Since phase angles are really low in magnitude ($\leq 1e^{-04}$, they are not the dominant factor in the current calculation, but the current is.

Overall, minimizing the square of errors for this case produces better results.

4.3. Test case 2 - Larger-scale network containing smart meter devices

Contrary to the previous network, this network is an unreduced real network, where only 3 GridEye devices are available, 1 in the transformer cabinet and the other 2 in other cabinets within the network. The original networks consists of **143 nodes and 142 branches**. The number of **smart meter devices are 177**, which are installed in *41 different nodes*. Since a node can have multiple loads, the aggregate of Smart Meter devices for every node is used. Thus, for this network, **41 Smart Meters are visualized**. **3 GridEye devices** are also installed at 3 different locations within the network. The original network provided is in figure 4.21:

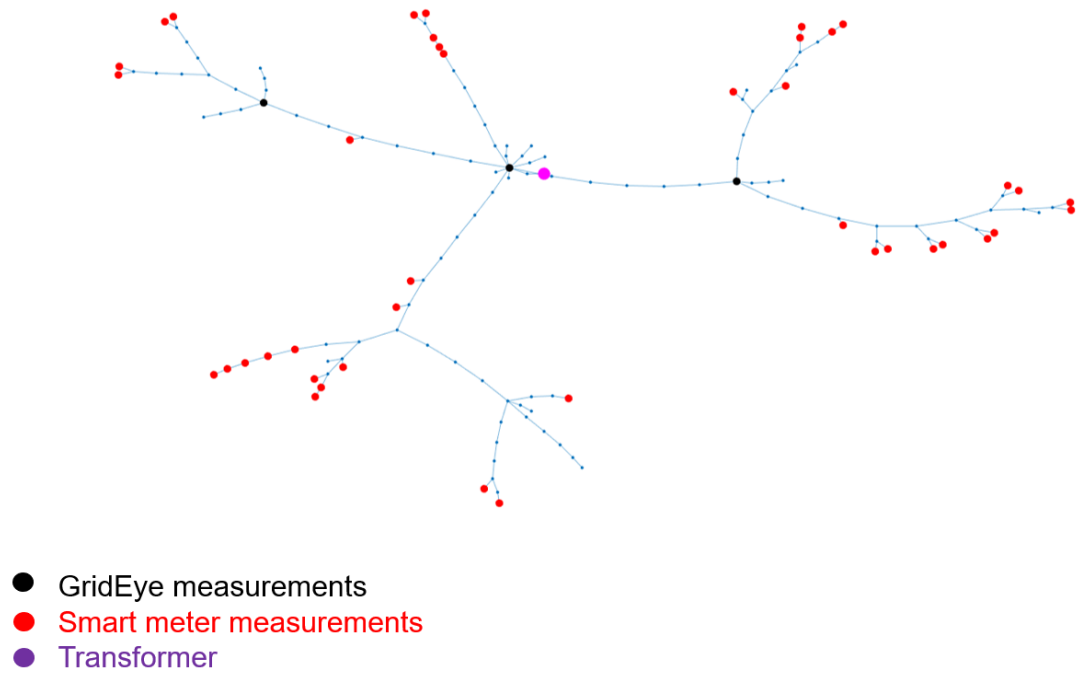


Figure 4.21: Single-line diagram of the second case-network.

The available measurements will also be used on the time-span of 1 week. Based on the visualization of the network, it can easily be observed that this network would require some reduction, to reduce the number of states and improve the algorithm's computational efficiency. This is done by utilizing the Kron reduction method, as mentioned in [23], in two steps. First, reduce all the nodes that appear only twice and are not metering nodes (i.e. the continuous branches). Nodes that appear only two times and are not metering nodes means that these are intermediate nodes that could be removed, without influencing the network by the Kron Reduction, since Kron reduction is tricky on situations where are node is connecting multiple branches together. In the second step the nodes that may appear more than twice in the connection matrix (From - To), but are not important to the examined network (not correlated with any measurement whatsoever). Moreover, the algorithm now requires manual weights representing the trust level of every measurement in the measurement function, as explained in Chapter 3.3 . Thus, the algorithm is expanded by adjusting the weights based on the trust of every measurement used in every specific equation. The explanation for the weights is shown below, but the numerical values are shown in 4.1. It must be noted that the higher the value, the lower these measurements impact the State Estimation. The reasoning behind the range of values chosen is described in the following:

- The highest trust is for the **zero-injection nodes** within the network. Since there are no major assumptions for these nodes, the trust level is much higher compared to the other nodes. This, nevertheless, can cause accumulated errors. This means that the nodes and branches that aggregate the errors of the metering devices (or leverage points), will be expected to have higher errors.
- The next highest trusted equations within the measurement function are the **nodal voltages of the GridEye devices**. These nodes require no assumption.
- Following the nodal voltages, the next trusted values are the **branch current equations provided by GridEye devices**. The trust level of nodal voltages is higher than branches provided by GridEye, due to lower noise ratio.
- **Smart meter current calculation** requires two major assumptions. First, that active and reactive power are uniformly distributed within the time-period of the Smart Meter

device. Second, the current calculation requires voltage assumption in nodes with unknown voltages. This means that equations with smart meters have lower weights than the previous cases. Finally, since smart meter devices are assumed to have 3% error, they should impact the state estimation less.

- Finally, **for assumed voltages**, since the upper part of the measurement matrix $\boldsymbol{H}$ (as shown in equation 3.8 is constituted by voltage equations in the form of $\boldsymbol{V_k} = V_k e^{j\delta_k}$ and all voltages that are not GridEye devices are unknown. Thus, the weight used should make these totally unimportant to the state estimation. This is done, because for the linear SE estimation a high measurement redundancy is required. Since this is not available, assumed voltages of low importance as used to create overcome the observability issue that ensues otherwise.

Weight	Value
Zero-injection node	0.1
Voltage GridEye	0.3
Current GridEye	0.7
Current SmartMeter	3
Voltage of unknown nodes	100000

Table 4.1: Numerical values of the weights used for the WLS method

Since power flow outputs are used, it's important to validate their match to the transformer's metered outputs of the GE device. This is shown in figure 4.22.

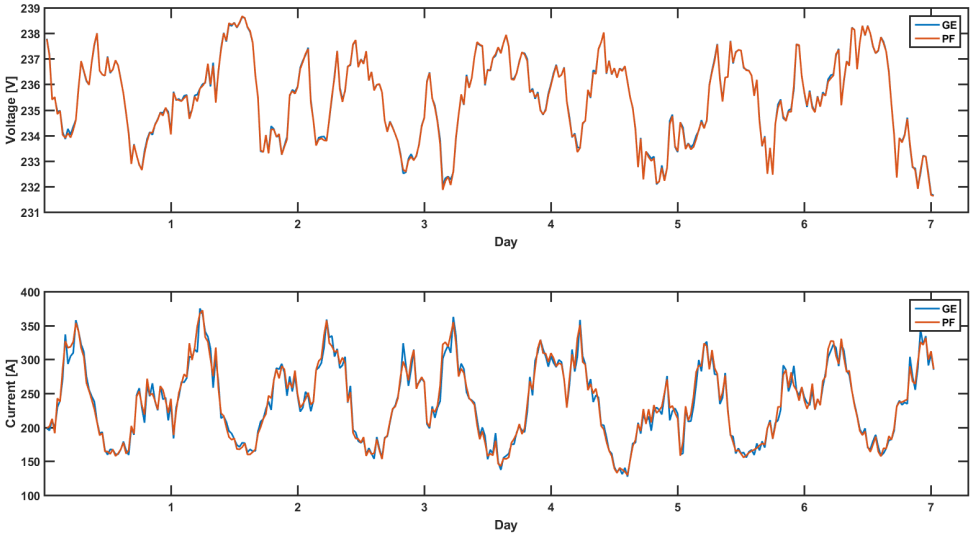


Figure 4.22: Comparison of the PF measurements and GE inputs provided for voltage magnitude and current amplitude.

After everything mentioned regarding reducing the network size, the reduced network is shown in figure 4.23:

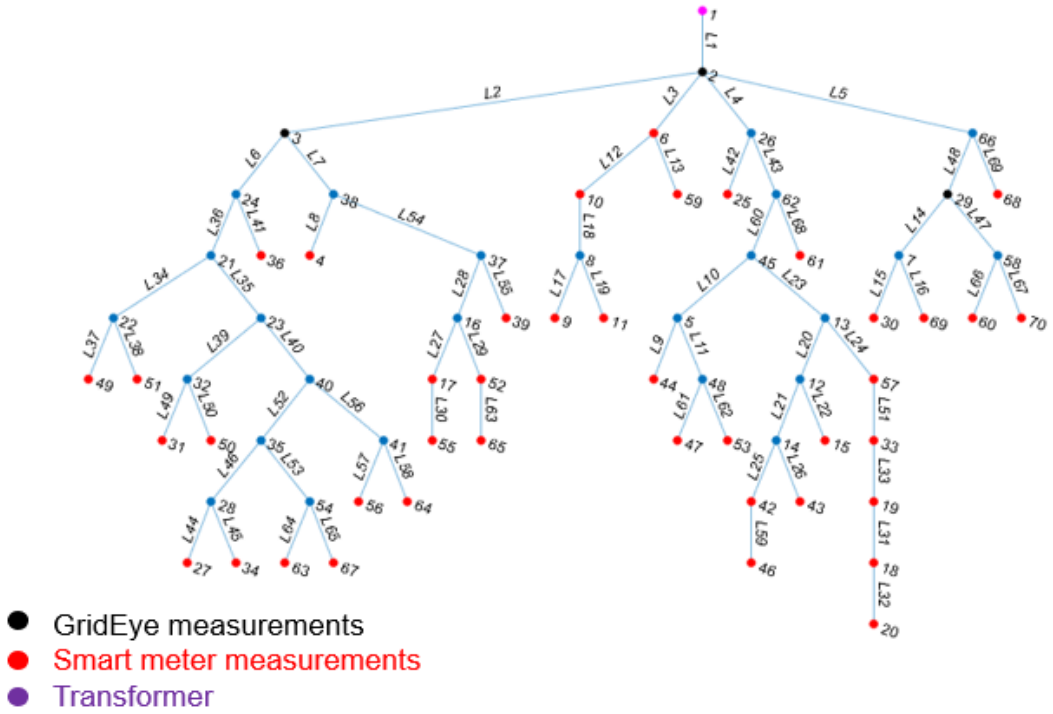


Figure 4.23: Reduced network with the radial representation.

The reduced network consists of **70 nodes and 69 branches**. The enumeration of the reduced network will be kept for all tests. The network state estimation will be for a week during December, where there was a time-match between the measurement units. Every time-stamp represents a 30-minute interval, which is the common ground for the first phase between 10-minute and 15-minute interval measurements. The range of voltage values with respect to time and with respect to every node are visible in the figures 4.24 and 4.25. Figure 4.24 is used to get a general idea of the numerical distribution of voltages during specific time-stamps, which will be useful to understand the time during the day of higher occurring errors. The figure 4.25 shows the nodal voltage variation for every node during this weekly period. The figures visualize the variation of voltages within the network. Similar for the voltage visualization, the current distribution within the branches is noted at 4.26. The examined network is of high consumption for a low-voltage network, thus higher current variations are expected.

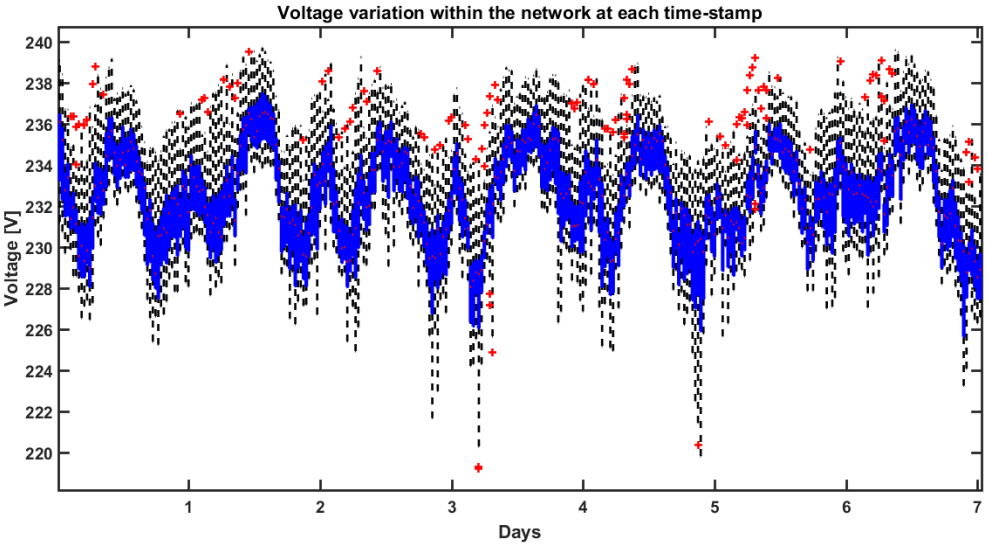


Figure 4.24: Power flow voltage magnitude variation for every time-stamp.

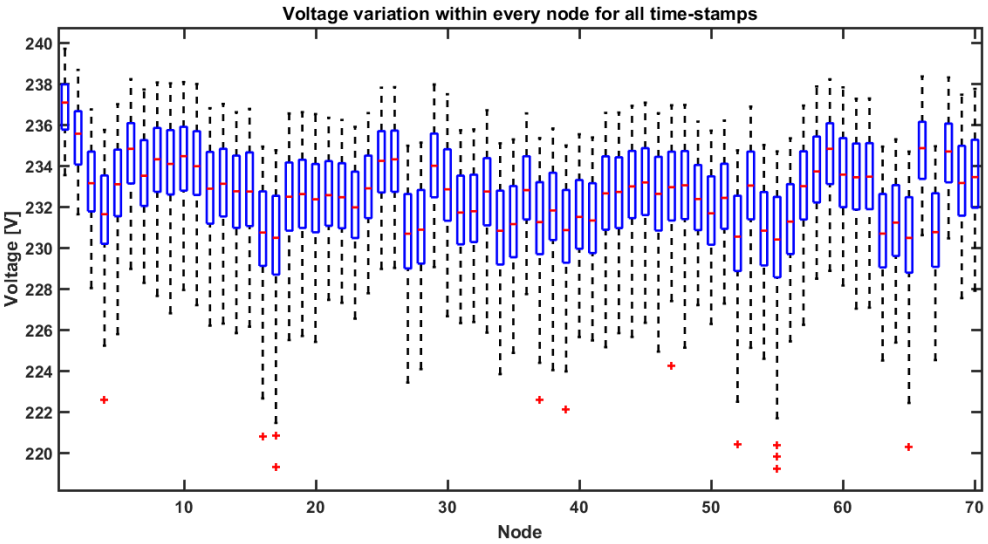


Figure 4.25: Power flow voltage magnitude variation for every node.

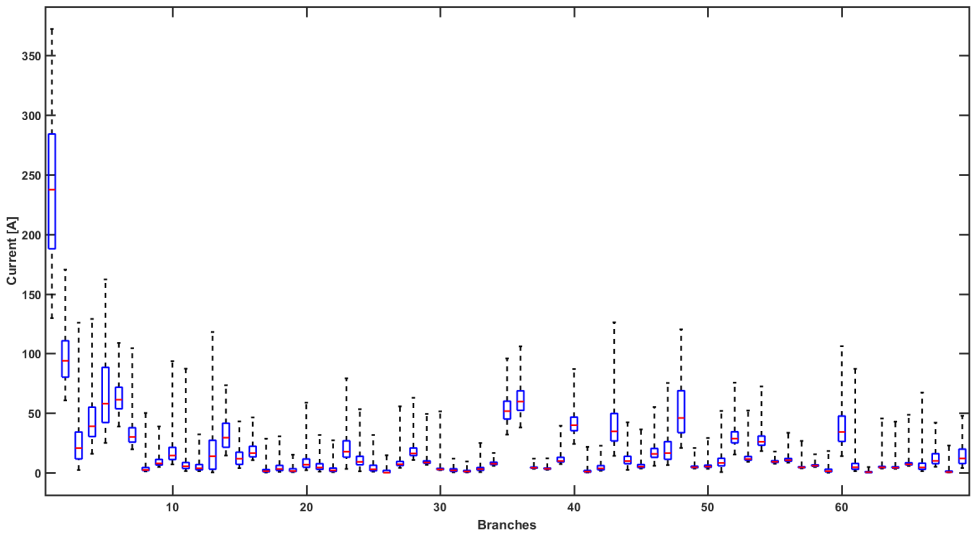


Figure 4.26: Power flow current amplitude variation for every branch.

For this test case, to understand the advantages and limitations of the algorithms, many scenarios will be studied. These scenarios will be numerically correlated in the end of the chapter. For every scenario, the distributions of errors within time and node/branch will be provided. Distribution of errors within the nodes will be shown in boxplots. This is done to realize the true variation within the errors for every node. Since robust statistics are median, and quartiles (25% and 75%) [29], boxplots provide the robustness of every node. Plots will be used to realize the maximum error within every specific time-frame. Histograms will be used to show how to errors of all nodes and branches vary for the examined period. This will be done separately for voltage magnitude, phase angle and current amplitude. A network visualization of the *four* most inaccurate nodes and branches for the studied time-span will be provided, when deemed necessary. Some scenarios include noise variations and some do not. Finally, in all figures provided, when a comparison is made, similar limits are used to easily visually compare the differences within the graphs.

4.3.1. Test case 2 - Scenario 1: Assumed voltage knowledge, both methods used

In this scenario, the measurement redundancy is high, since voltage availability is assumed from the power flow outputs as voltage magnitudes. This represents a case of having a Smart Meter or GridEye device in every one of the nodes within the network. Since voltage is available and there is no noise, really small errors are expected as outputs. This can be made obvious in figures 4.27,4.28 and 4.29, where close to zero errors are achieved. This means that the algorithm works for high redundancy in larger-scale networks.

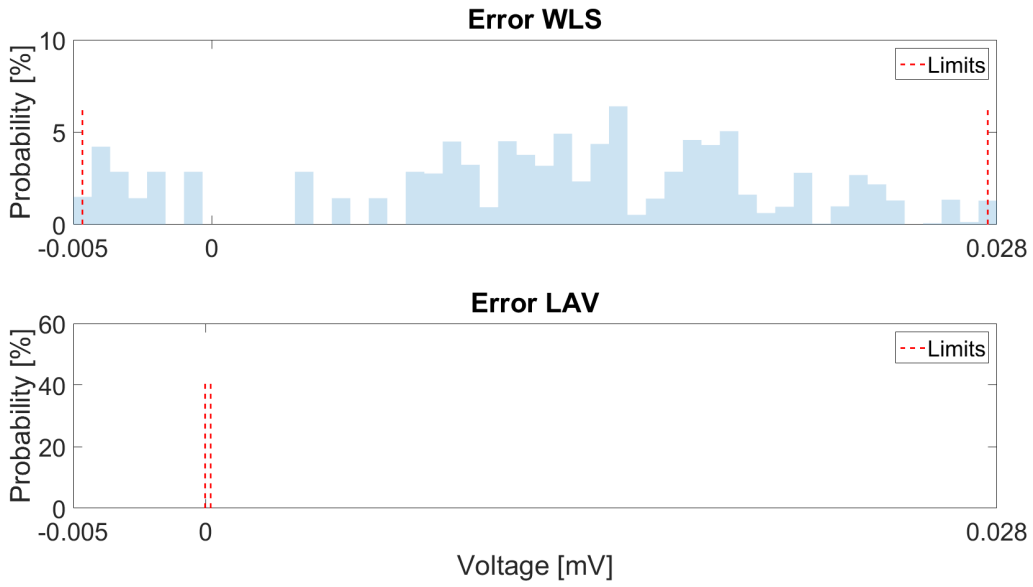


Figure 4.27: Histogram of voltage magnitude error distribution of both methods.

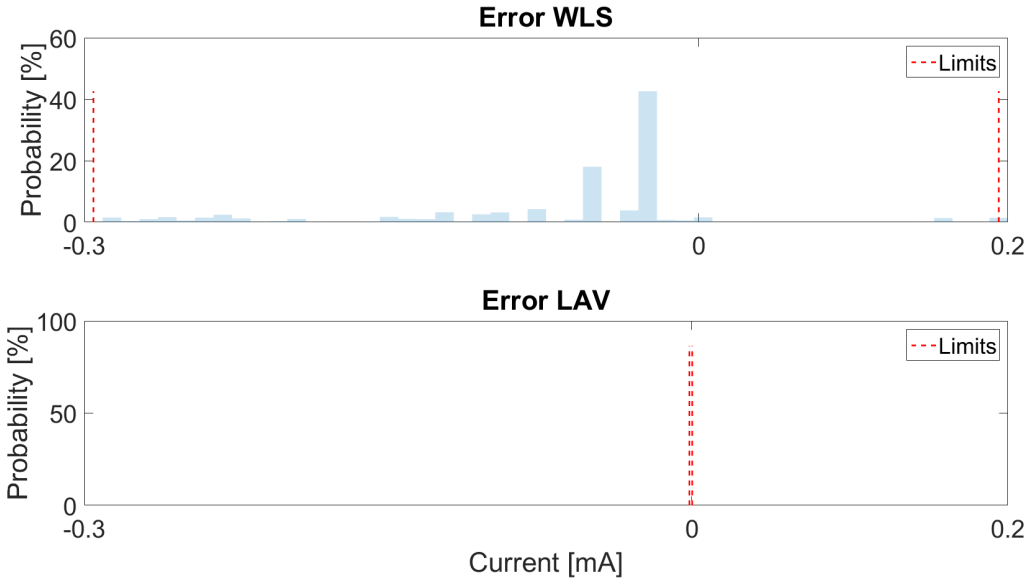


Figure 4.28: Histogram of current amplitude error distribution of both methods.

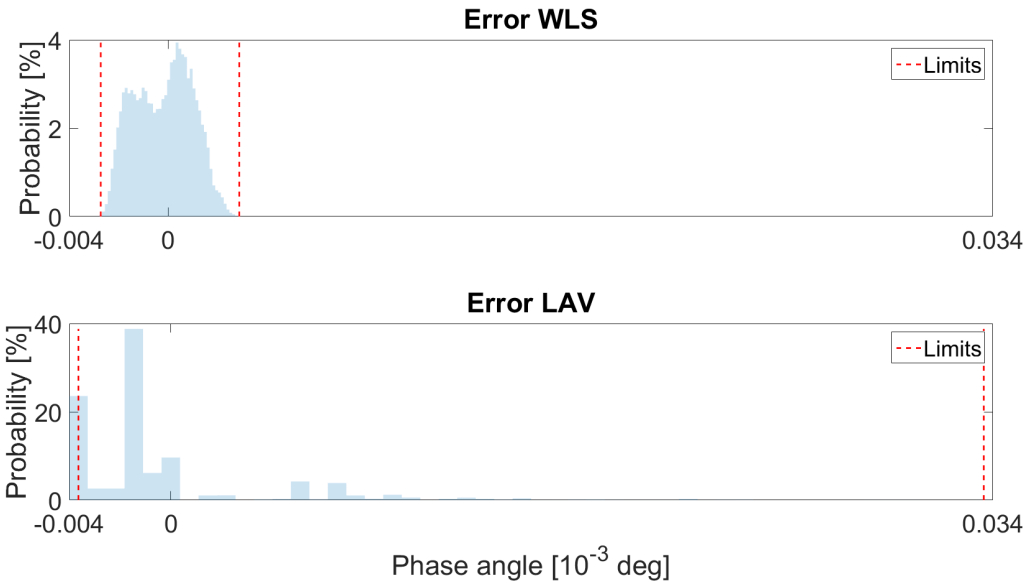


Figure 4.29: Histogram of voltage phase angle error distribution of both methods.

The errors are minimal (degree of magnitude of 10^{-5} for voltage and 10^{-5}, thus the wanted results for this ideal scenario are achieved.

4.3.2. Test case 2 - Scenario 2: True measurement availability, both methods used

The difference of this scenario, compared to the previous one, is the lack of knowledge of the nodal voltages of smart meter devices. Still all GridEye devices are used. This scenario is the base case scenario, meaning that all available measurements are used. The distribution of error for every node and branch are shown in figures 4.30, 4.31 and 4.32.

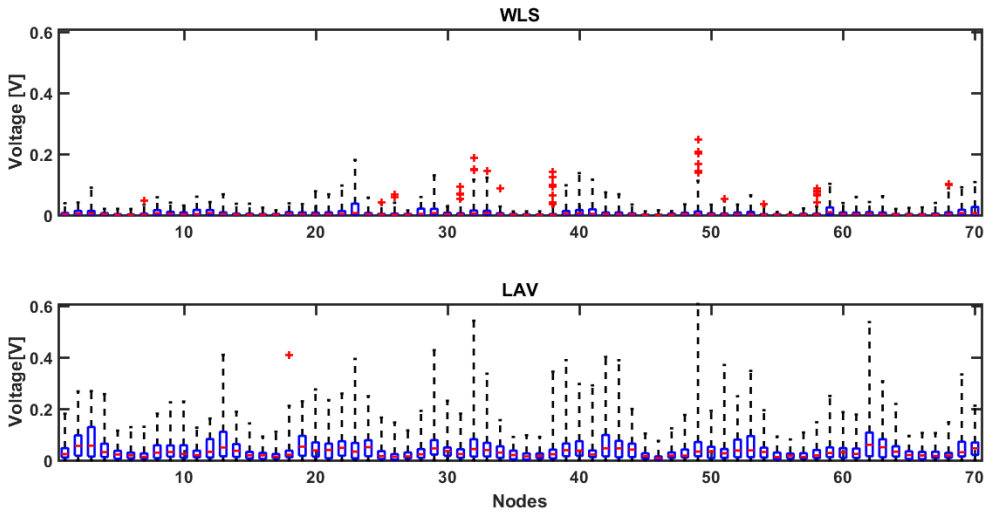


Figure 4.30: Voltage magnitude errors for every node.

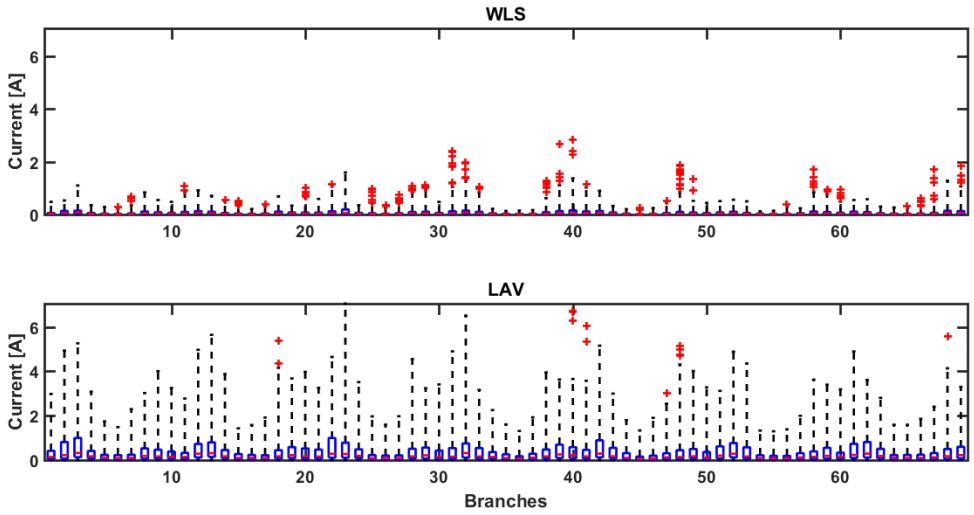


Figure 4.31: Current amplitude errors for every branch.

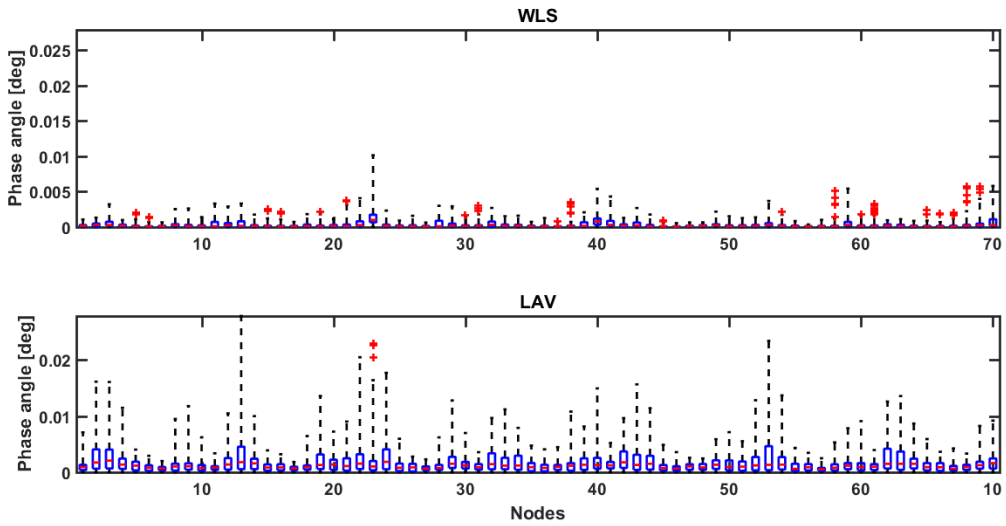


Figure 4.32: Voltage phase angle errors for every node.

As it's obvious from figure 4.30, the errors are higher for the LAV method. The nodes that are susceptible to outliers and highest errors (nodes 32, 38 and 49) are either loads of the highest consumption or their closest nodes. The highest errors occur at the time of highest consumption, as this causes higher errors for the branch current calculation. While slightly more comparable values for the phase angle are computed for both methods, still higher level of voltage phase angle error ensues for the LAV method, as shown in 4.32. Mainly voltage magnitude error is responsible for the major differences in the branch currents, as shown in the figure 4.31. Similar patterns and outliers for all the mentioned figures occur for both LAV and WLS method. This means that both methods operate quite similarly, but the minimization of squares for assumed values seems better for this case.

The maximum error between inputs and outputs is provided in the figures 4.33, 4.34 and 4.35. These figures are necessary to understand which time-stamps produce the highest errors and to validate that trully the times of highest consumption cause higher errors (see 4.22).

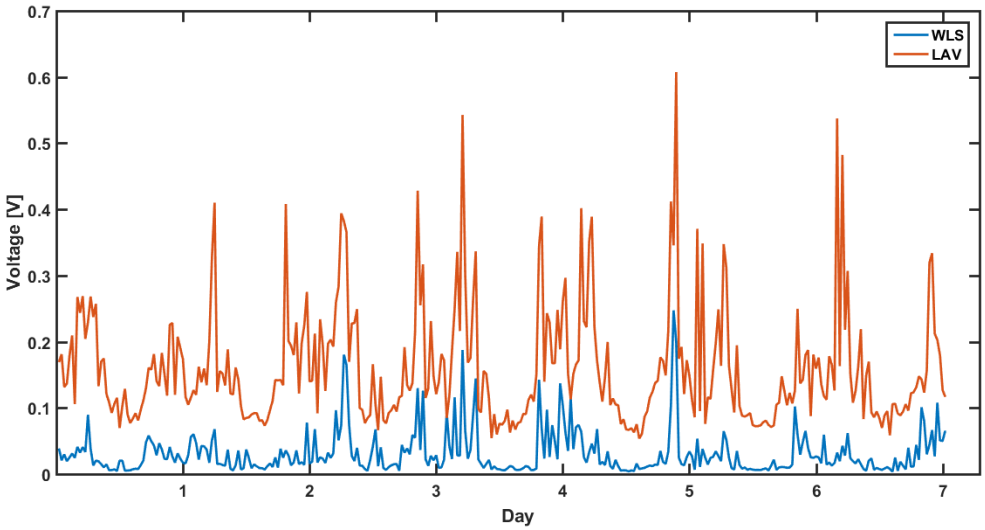


Figure 4.33: Maximum voltage magnitude errors for every time-stamp.

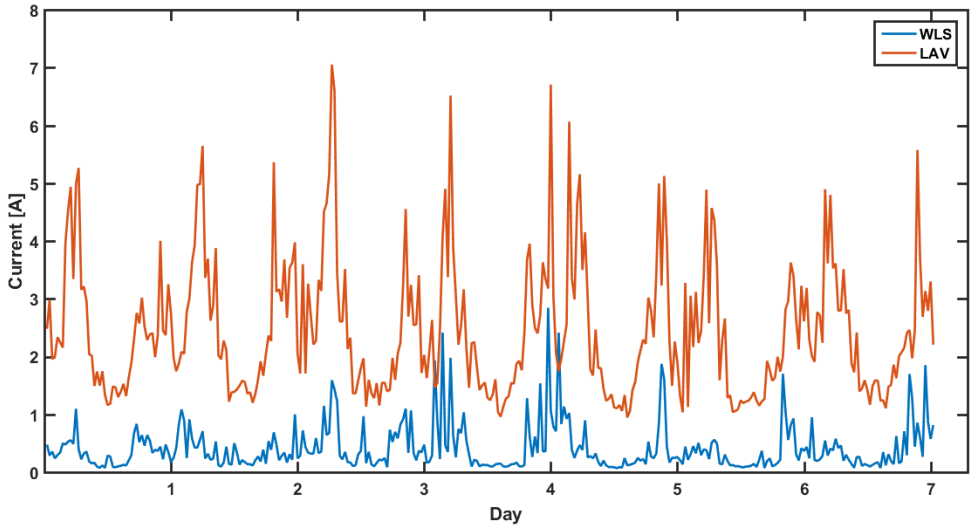


Figure 4.34: Maximum current amplitude errors for every time-stamp.

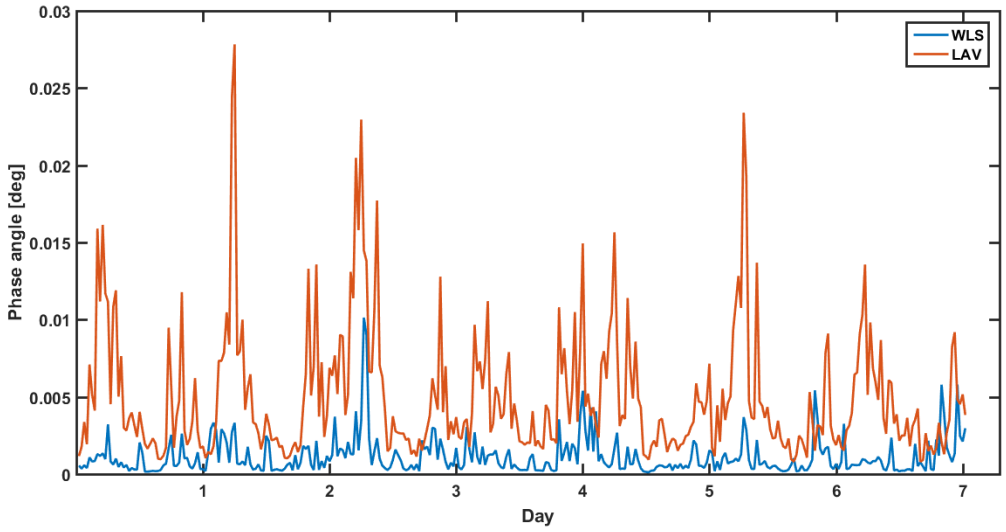


Figure 4.35: Maximum voltage phase angle errors for every time-stamp.

Similarly to the boxplots, the maximum error for voltage, current and phase angle at every time-stamp is lower for the WLS methods. It's also more visible now that the same patterns occur for both methods, but also in times of higher consumption, LAV is more prone to errors than the WLS one. The higher error for the voltage magnitudes thus creates higher maximum error for every branch. Since the errors are provided in absolute terms, to further comprehend the error distribution of the methods irrespective of node or time-stamp, the histogram of the errors is shown in figures 4.36,4.37 and 4.38.

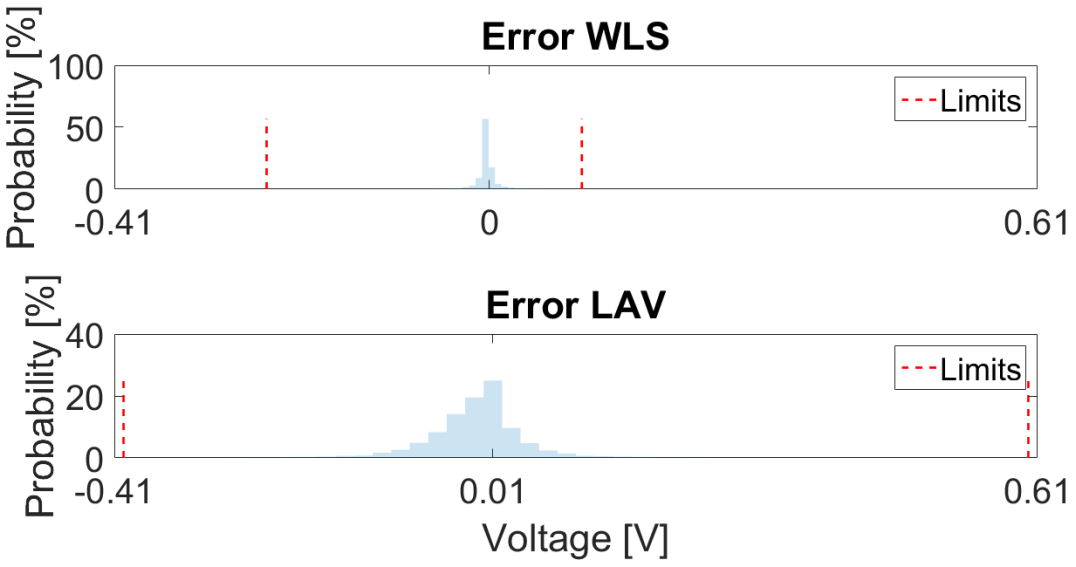


Figure 4.36: Histogram of voltage magnitude error distribution of both methods.

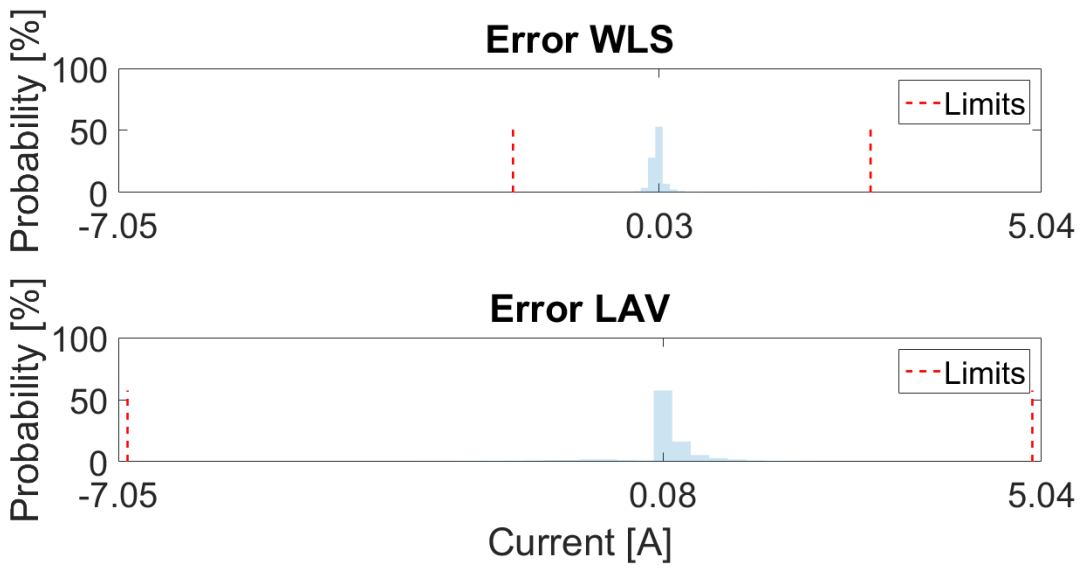


Figure 4.37: Histogram of current amplitude error distribution of both methods.

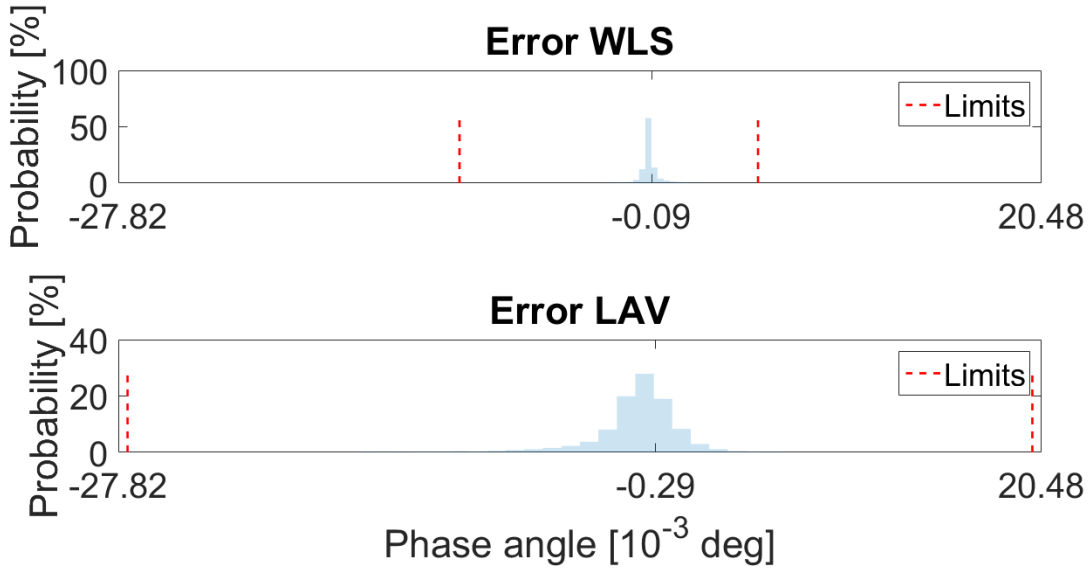
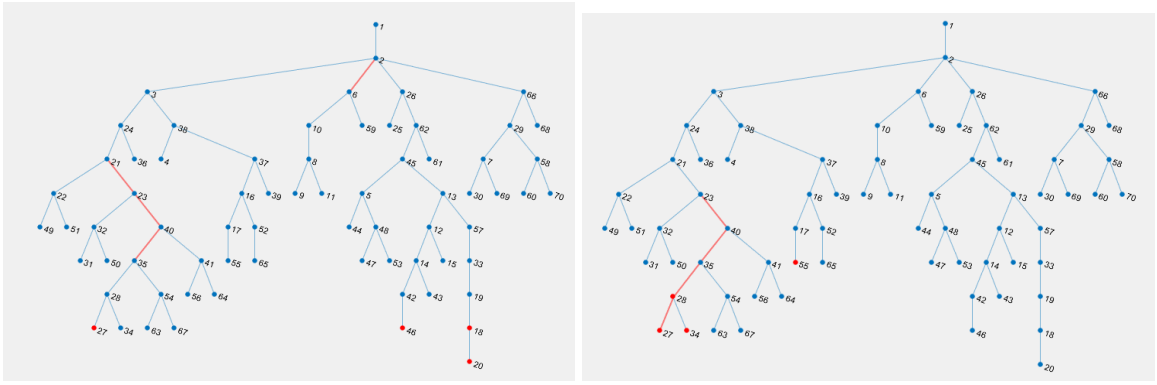


Figure 4.38: Histogram of voltage phase angle error distribution of both methods.

Both methods still follow a distribution around 0, with a higher range of error for the LAV method, as previously discussed. The voltage assumptions cause lower current estimates of branches, which in turn creates that mismatch between inputs and outputs. It's also worth noting that the errors are the positive side are higher for the LAV and lower for the WLS. For branch currents, the signs of errors are changed, while still having a much higher error for the LAV case. Similar results can be reached for the phase angle distribution error distribution.



(a) WLS. (b) LAV.

Figure 4.39: Most erroneous nodes and branches of SE outputs throughout the test period.

Furthermore, it can be seen in 4.39 that similar branches have the most mismatches for the examined time period for both methods. In this case of WLS, highest nodal error occurs at different nodal positions that with the LAV method.

4.3.3. Test case 2 - Scenario 3: Methods comparison with only using one GridEye

In this examined scenario, the use of explicit use of nodal voltages and line currents only of the GE device in the transformer is provided, alongside with the smart meter devices. The difference thus lies in not knowing the voltage and currents correlated to the GridEye devices. This is done to evaluate the importance of the quantity of GridEye with respect to the outputs. The boxplots are provided in 4.40, 4.41 and 4.42.

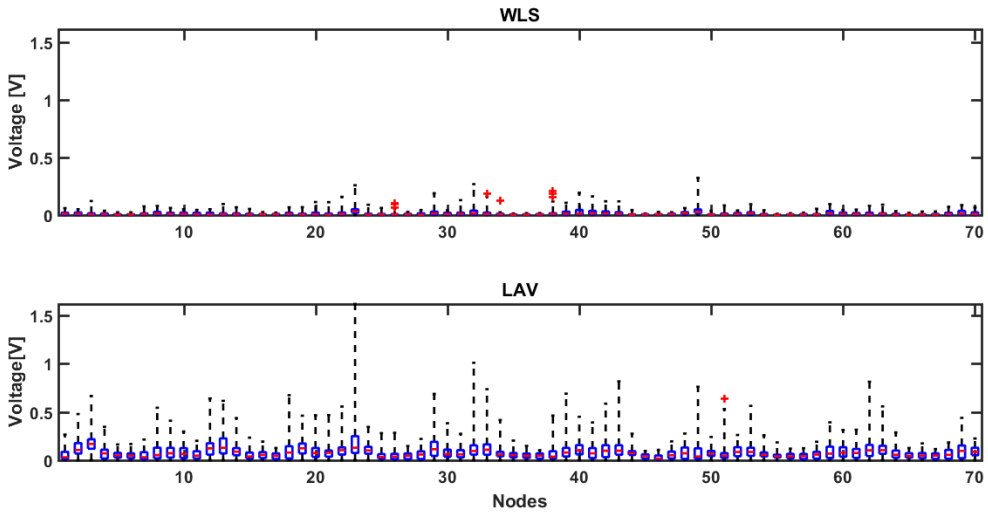


Figure 4.40: Voltage magnitude errors for every node.

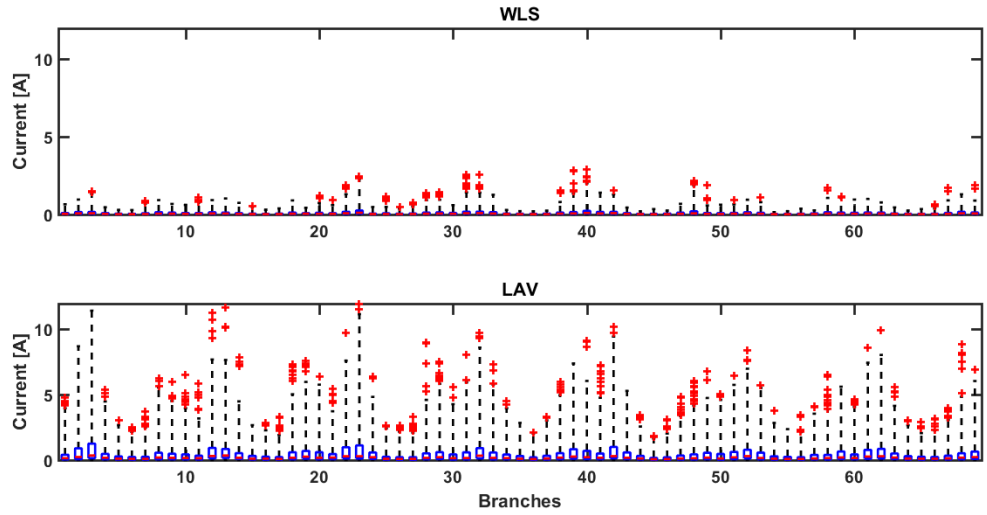


Figure 4.41: Current amplitude errors for every branch.

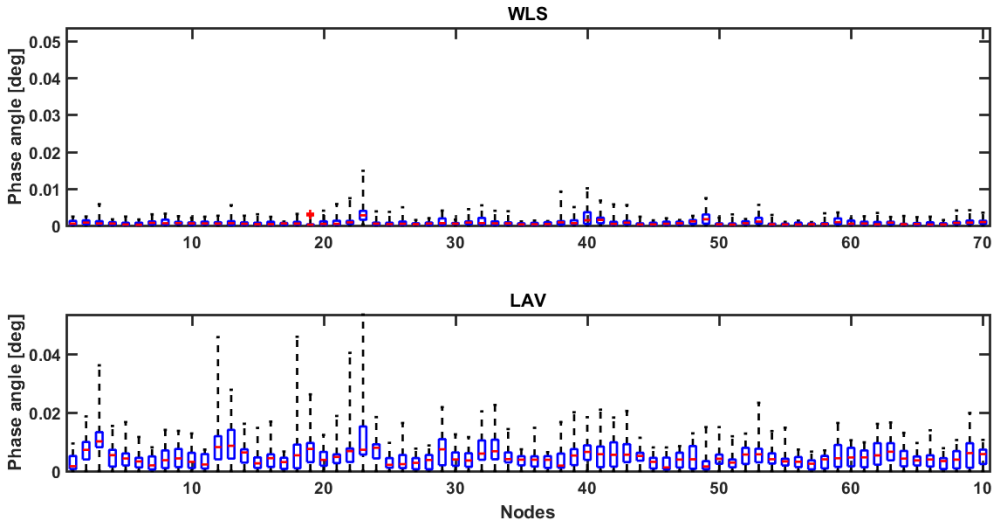


Figure 4.42: Voltage phase angle errors for every node.

Here the limitations of the LAV implementations are more clear. While the WLS is only slightly worse in terms of outputs, the LAV method is much worse and not trustworthy. The patterns are still kept for the nodes, but the error in some cases exceed 1V. This instantly affects the branch currents, as it's obvious in 4.41. As current is calculated through the voltage state outputs, currents are more erroneous one for the LAV. Similarly for nodal phase angle,the outputs are considerably worse for the LAV than the WLS.

The maximum error between inputs and outputs is provided in the figures 4.43, 4.44 and 4.45.

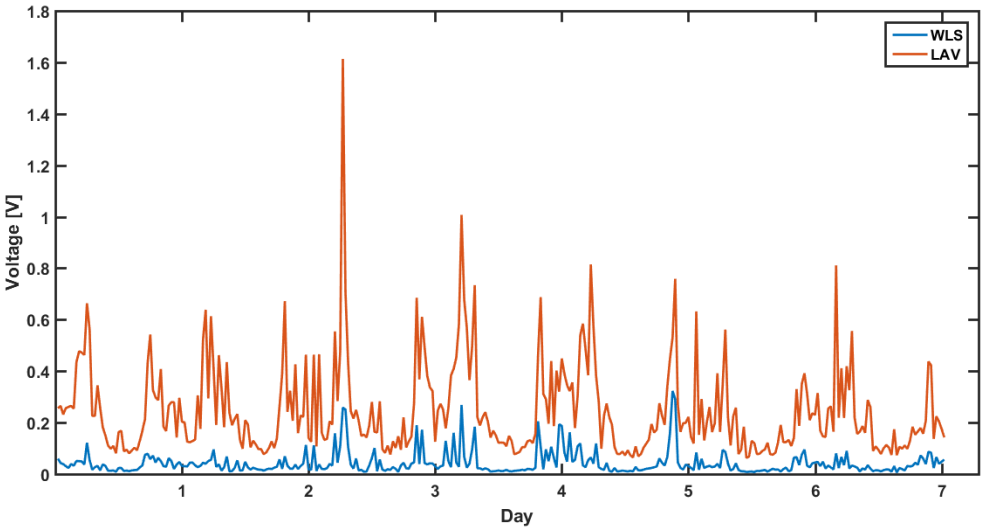


Figure 4.43: Maximum voltage magnitude errors for every time-stamp.

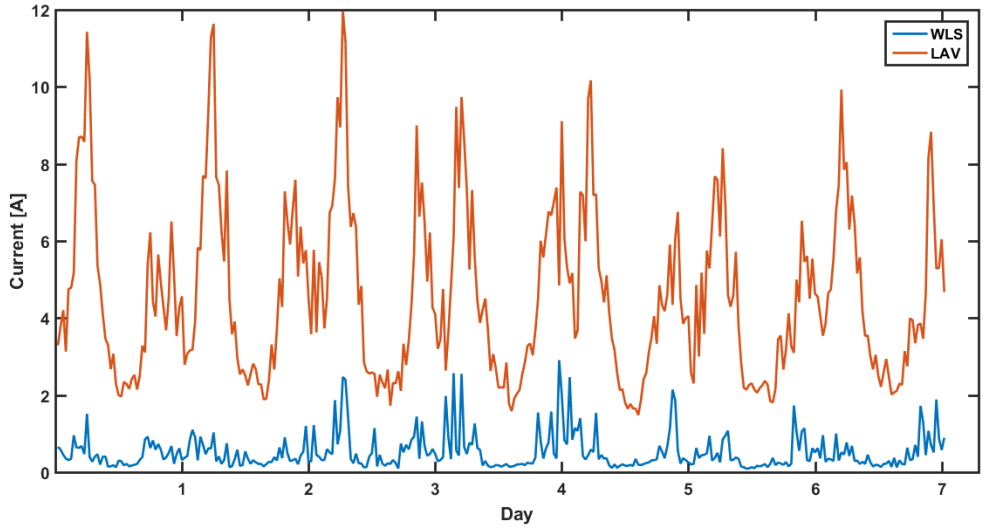


Figure 4.44: Maximum current amplitude errors for every time-stamp.

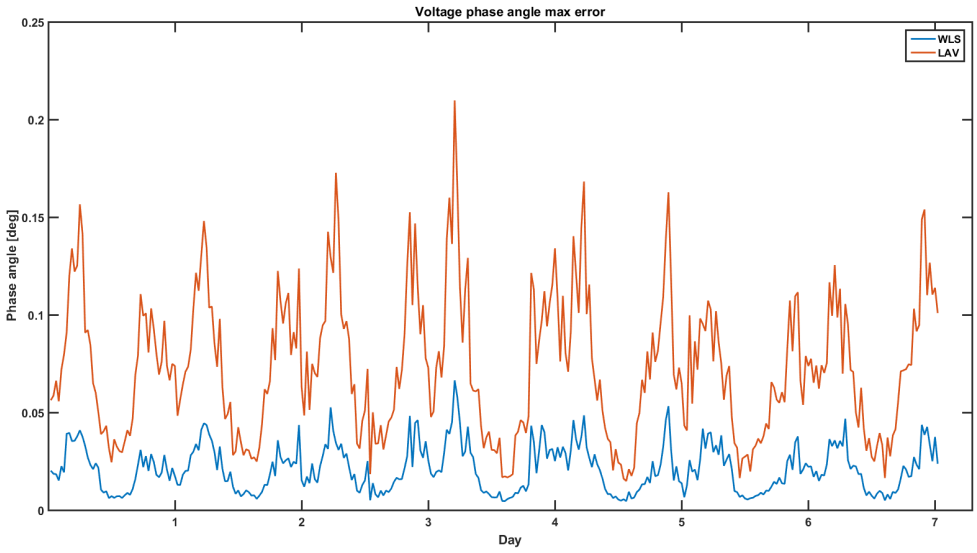


Figure 4.45: Maximum voltage phase angle errors for every time-stamp.

What was previously mentioned regarding the differences between both method outputs is quite visible in figure 4.43. The pattern of errors is repeated, meaning at highest errors of WLS the highest errors of LAV are noted. Nevertheless, this error at times of high consumption make LAV, at this state, prone to an assumption-based SE. Similar problems occur for the branch currents and the phase angle, rendering LAV for this specific scenario inadequate. Nevertheless, WLS errors are within adequate limits. Since the errors are provided in absolute terms, to further comprehend the error distribution of the methods irrespective of node or time-stamp, the histogram of the errors is shown in figures 4.46, 4.47 and 4.48.

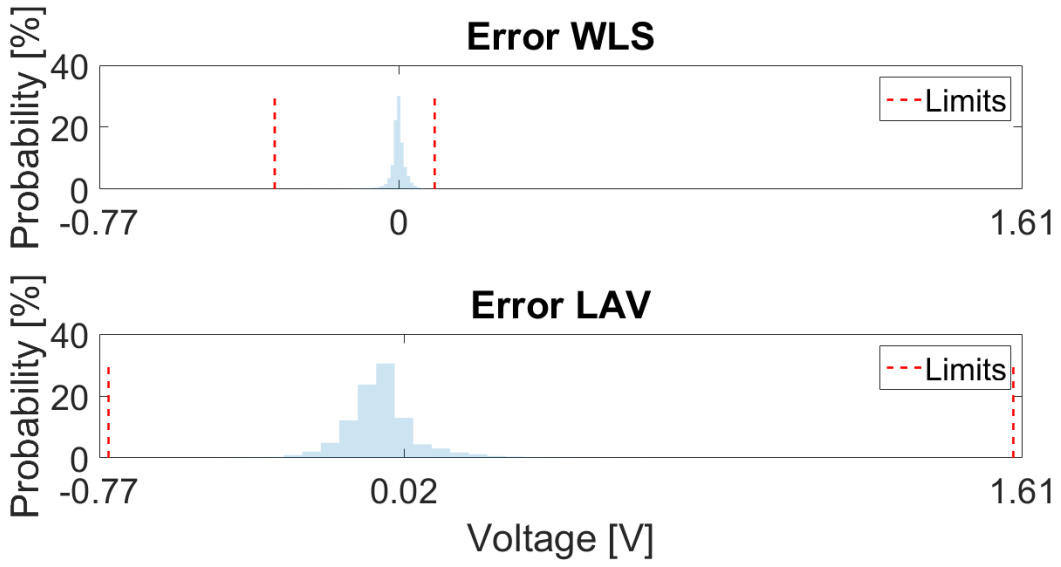


Figure 4.46: Histogram of voltage magnitude error distribution of both methods.

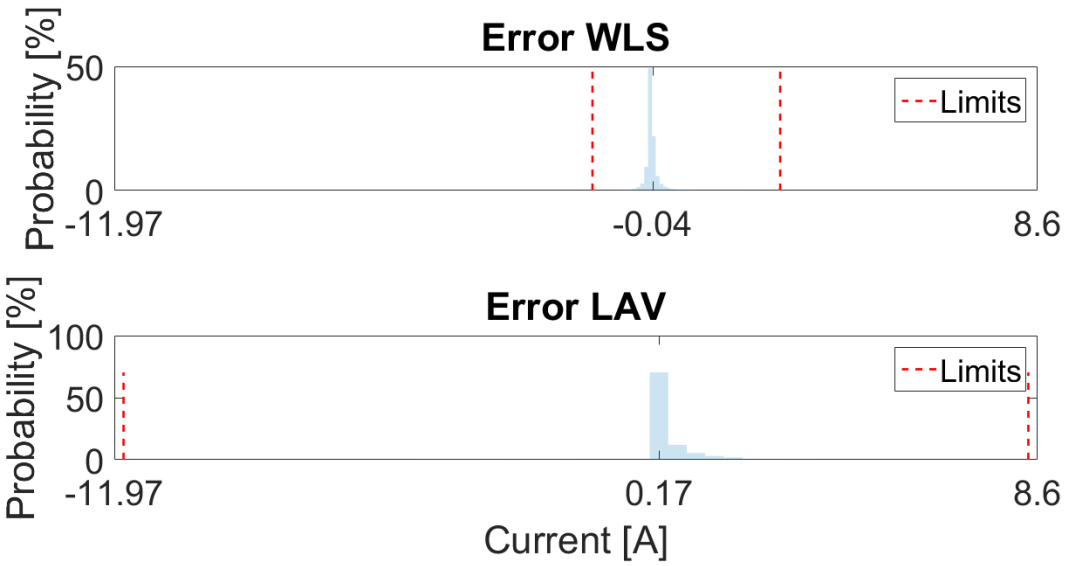


Figure 4.47: Histogram of current amplitude error distribution of both methods.

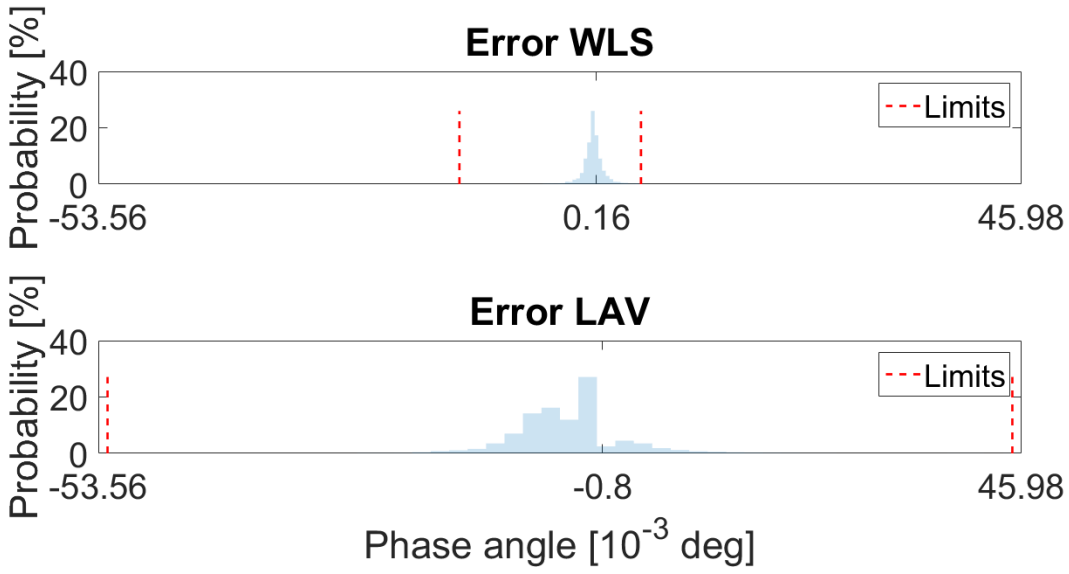
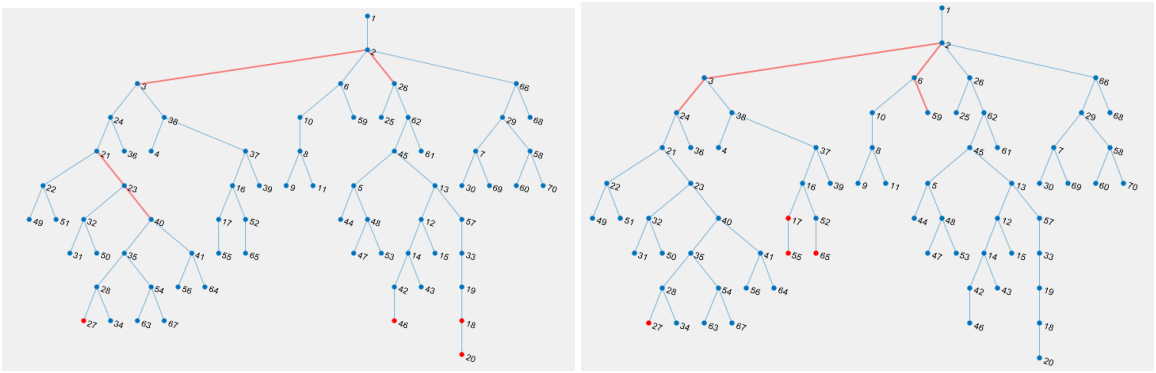


Figure 4.48: Histogram of voltage phase angle error distribution of both methods.

While both methods follow a normal distribution around 0, it's obvious that LAV has a higher variance for all histograms. While the extreme cases are slim, they produce untrustworthy outputs. It's important to notice that most errors are around 0 for all histograms.



(a) WLS. (b) LAV.

Figure 4.49: Most erroneous nodes and branches of SE outputs throughout the test period.

The highest errors still appear in the similar nodes for the WLS, as it's shown in figure 4.49. But, the branches of highest errors change, moving to branches closer to GE. This is done, since as previously mentioned, zero-injections nodes are treated with high weights, thus sometimes carrying the errors of other nodes to the nodes closer to the transformer. For LAV, similar nodes are also causing higher issues, and the accumulated error of those nodes cause errors to different branches than the WLS. Thus, the quantity of GE devices also change where most errors occur for branches mostly.

4.3.4. Test case 2 - Scenario 4: Effect of number GridEye devices in noiseless environment, WLS method

In this case, the difference between using multiple or a single device will be analyzed for a noiseless scenario. Since it was inconclusive whether noise was interfering with the higher current amplitudes of the previous case, a noiseless comparison disambiguate the differences. Also, due to the advantages of WLS, for this case only the WLS method will be directly compared for this scenario. This makes sense, because in the noisy scenario it is difficult to implement LAV, thus to directly correlate them afterwards, LAV will not be used.

As it is seen from figure 4.50, slightly higher voltage errors occur when using only one GE device, but these errors are comparable between the two cases. More outliers appear in the case of a single device. Furthermore, the number of the devices do not play a significant role in the branch current calculation, as it's obvious from figure 4.51. There are obvious differences in the branches where the GE was installed (branches 2 and 48), while the rest seem unaffected by the the number of devices implemented. Still, phase angle errors remain similar for both cases, as it's shown in 4.52. Node 24 produces the highest possible phase angle error, which is a zero-injection node close to high consumption loads.

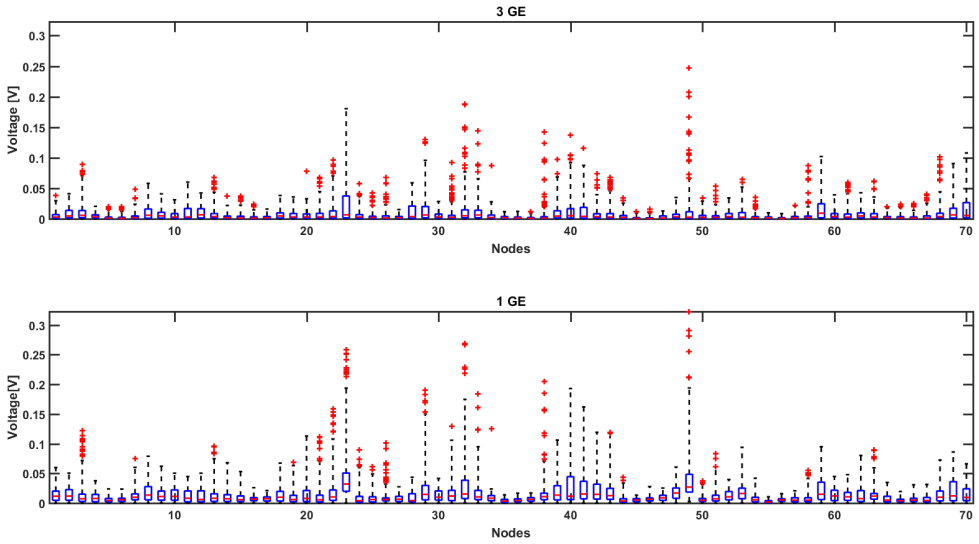


Figure 4.50: Voltage magnitude errors for every node.

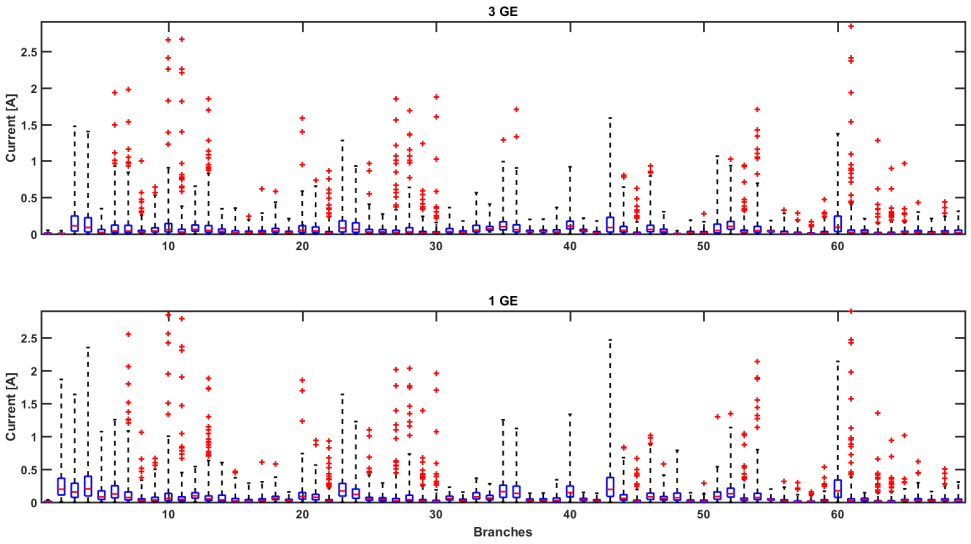


Figure 4.51: Current amplitude errors for every branch.

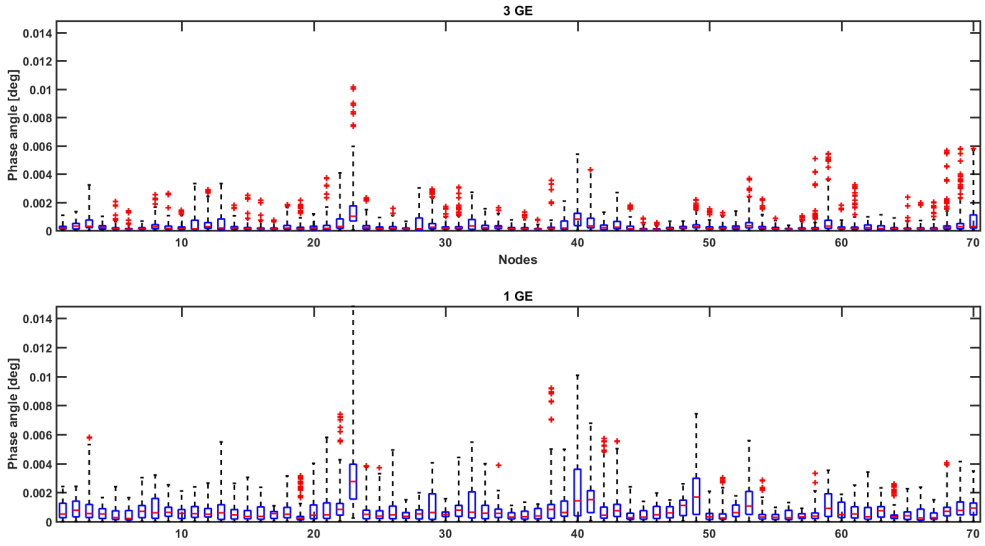


Figure 4.52: Voltage phase angle errors for every node.

In terms of maximum error per time-stamp, both cases produce similar outputs, as it is shown in figure 4.53. Slightly higher peaks at times of high consumption are noticed, but overall the outputs are not much different neither for current outputs, as noted in figure 4.54. Nonetheless, there is no significant disadvantage in terms of maximum error for phase angle when using a single module of GE in the network is noted in 4.55.

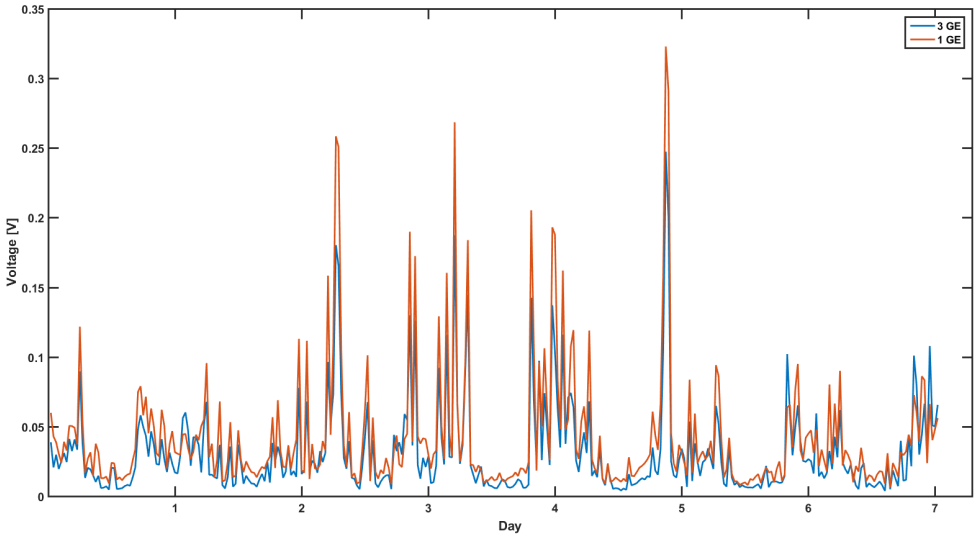


Figure 4.53: Maximum voltage magnitude errors for every time-stamp.

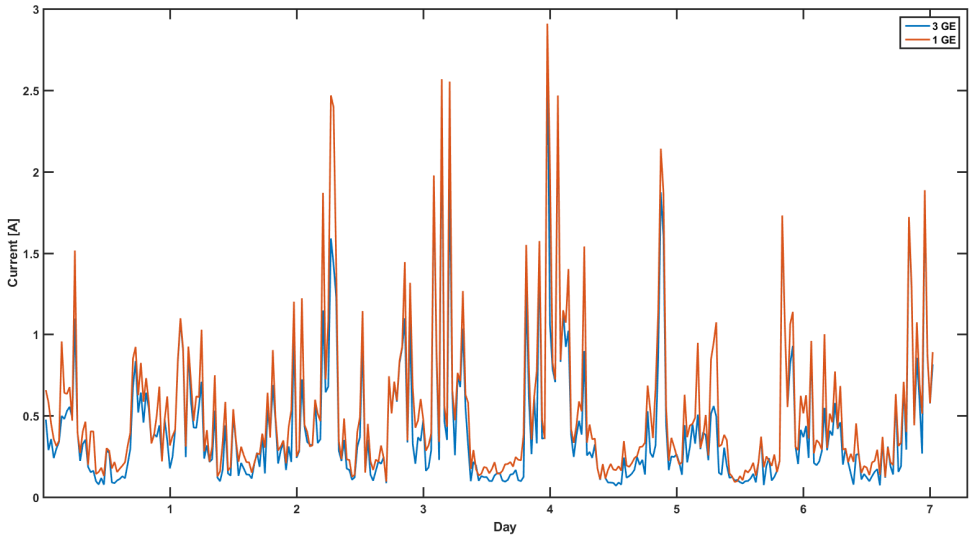


Figure 4.54: Maximum current amplitude errors for every time-stamp.

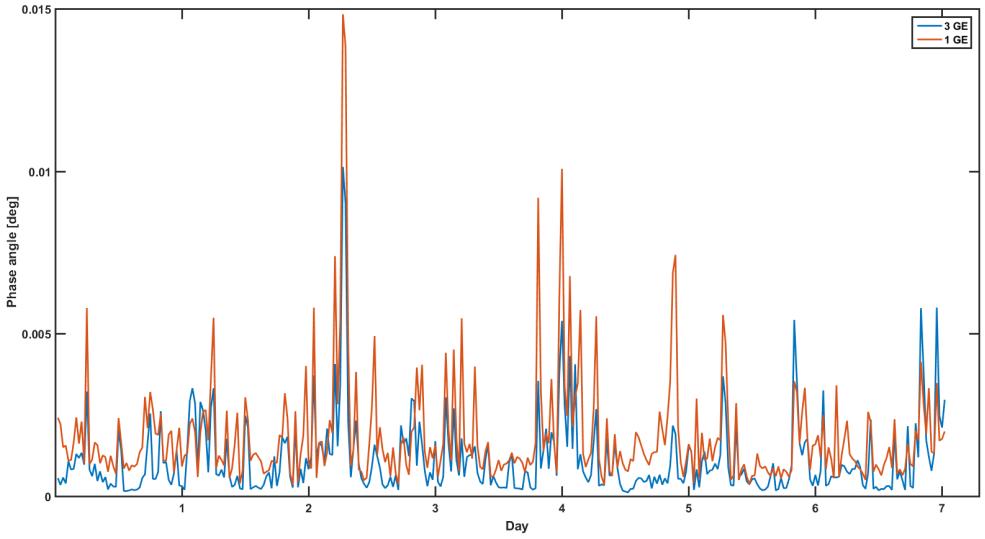


Figure 4.55: Maximum voltage phase angle errors for every time-stamp.

Finally, what was earlier discussed in terms of errors, is shown also in figure 4.56, where lower differences are shown between the two examined cases. Slightly more centered around zero are the the errors when using 3 GE devices for all histograms. The boundaries are similar. For cases where 1 GE device is better, is because the accumulated errors between the neighbouring nodes or branches to GridEye devices create a mismatch, which in turn creates slightly higher errors than not specifying the nodal or current values.

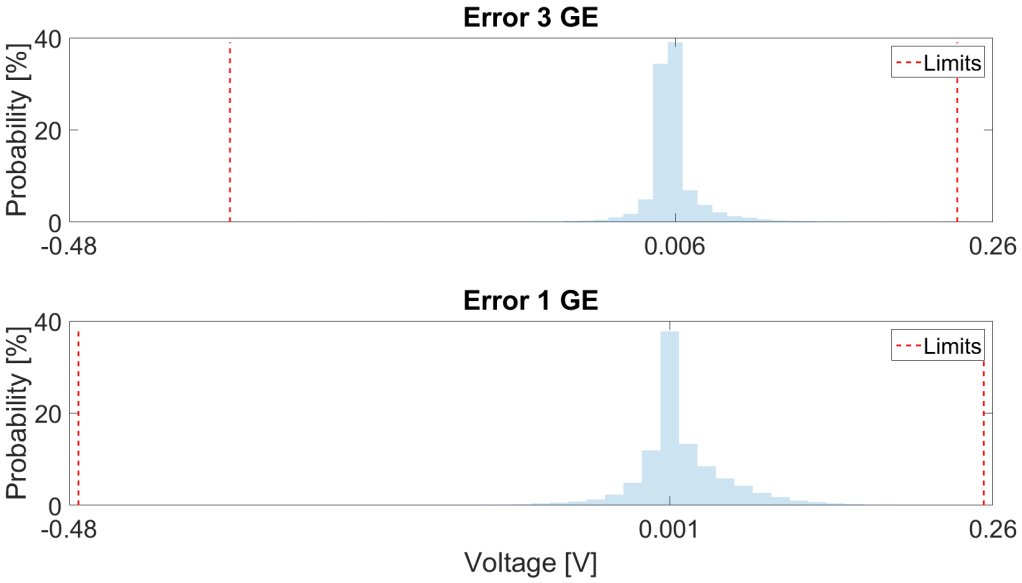


Figure 4.56: Histogram of voltage magnitude error distribution of both methods.

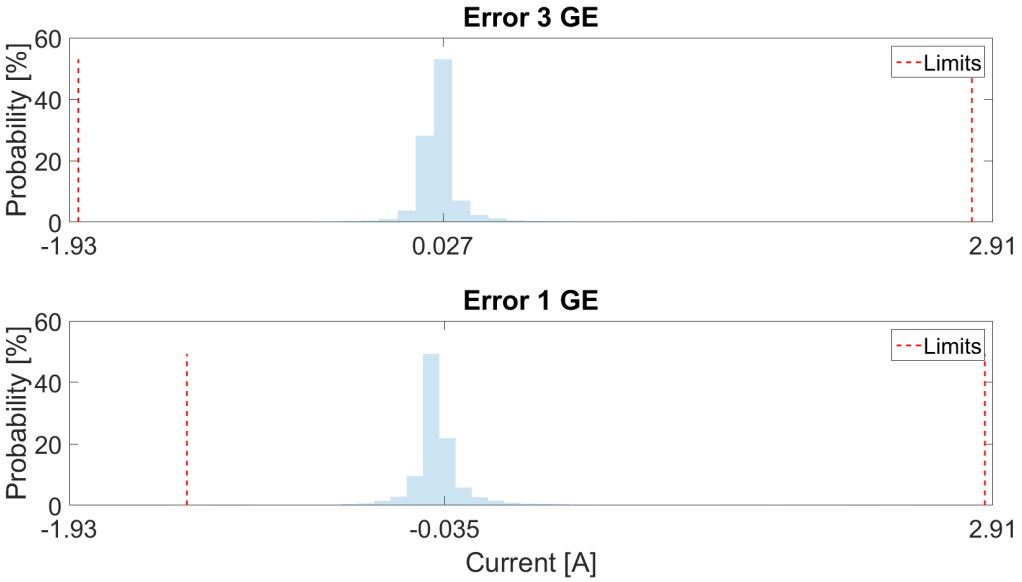


Figure 4.57: Histogram of current amplitude error distribution of both methods.

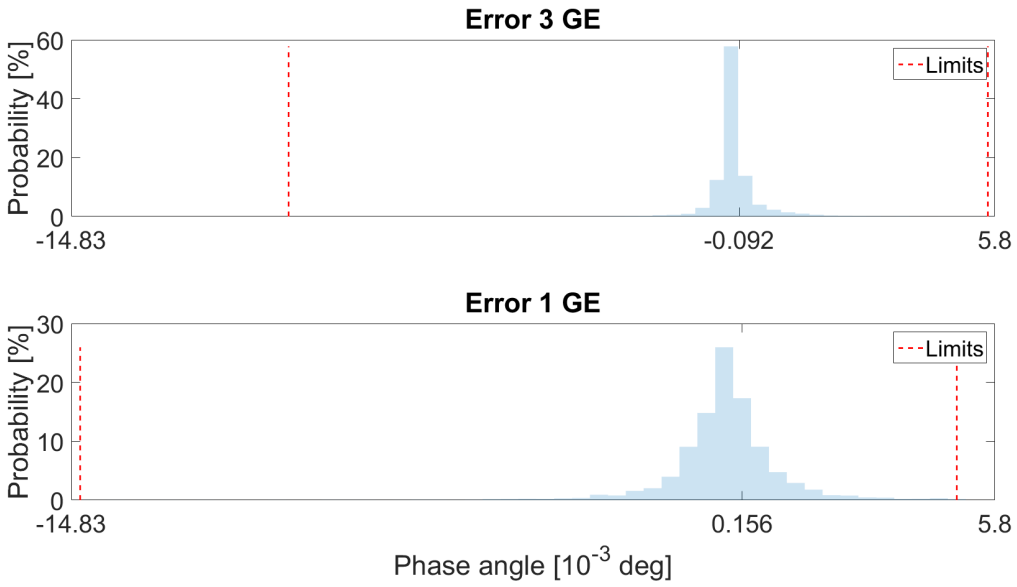


Figure 4.58: Histogram of voltage phase angle error distribution of both methods.

As it's overall shown, the amount of GridEye device enhances the outputs, but not to a big margin.

4.3.5. Test case 2 - Scenario 5: Effect of number GridEye devices in noisy environment, WLS method

Noise is an important part of any mathematical calculation within the power systems. For an algorithm to be valid, noise must be inserted relative to the possible error of the metering devices. This error for GE devices is **0.1% for voltage measurements** and **1% for current measurements**. Similarly, the SM device error is assumed to be at **3% for active power**. These noises will be implemented in the measurements, to validate the error with respect to a noiseless input. **The noises cases for every time-stamp are 100**.

The computation cost for LAV to produce outputs for all times in a noisy environment renders LAV incapable of noise inclusion. Since WLS method produces better and faster results, it will be preferred as the method of SE when introducing noise to the measurements. Two noise sub-scenarios will be evaluated. The first is the base case of including all GE devices of the network and the second only the GE installed in the transformer will be included. Finally, the network representation of highest errors will not be included in these parts, since the outputs are similar to the previous cases and do not provide any new input for the state estimator. It must be also mentioned that since there is knowledge of more branch currents, more errors are implemented in the scenario of 3 GE devices, which can probably contradict each other.

It makes sense to have higher variations for all cases, compared to noiseless scenarios. Since including noise in the level of 0.1% for voltages, which maximum voltage is 240, as shown in 4.25 (thus a possible additional 0.24V of variation through all measurements), as well as 1% metering error for transformer current, which maximum value is 360A, as shown in 4.26 (thus maximum possible variation of 3.6A). The first notable remark from 4.59 is slightly higher voltage magnitude error outputs for the case where only one GE device. More outliers appear in the single device. Nevertheless, no notable differences exist for this case. Similar for the current outputs, 3 GE are similar to 1 GE, and it's due to the noise of the current that affects the estimation of other branches. Nodal phase angle has no significant differences.

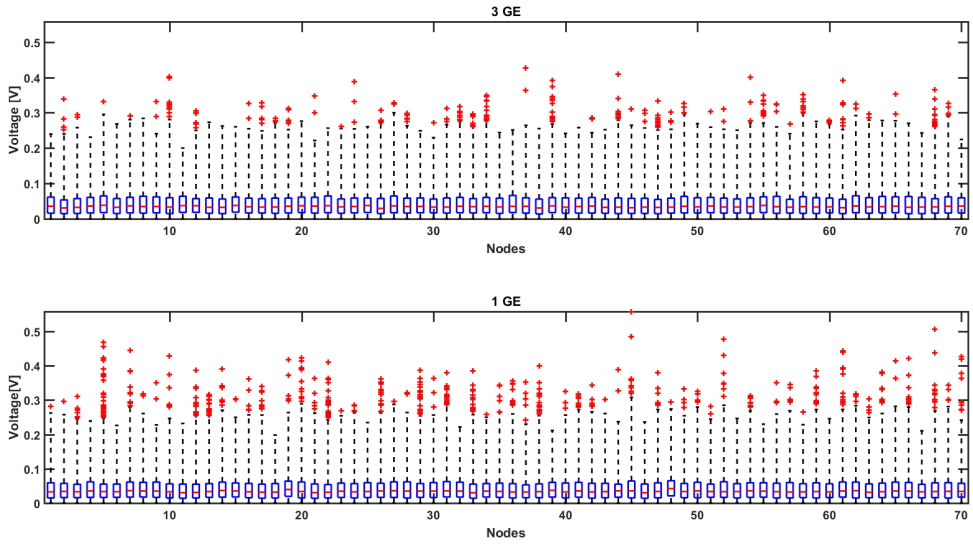


Figure 4.59: Voltage magnitude errors for every node including all noises.

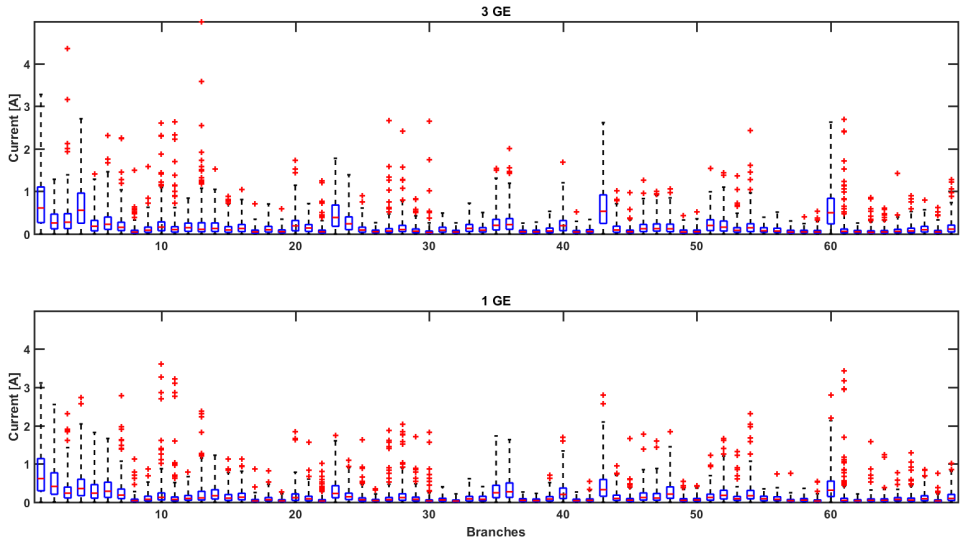


Figure 4.60: Current amplitude errors for every branch including all noises.

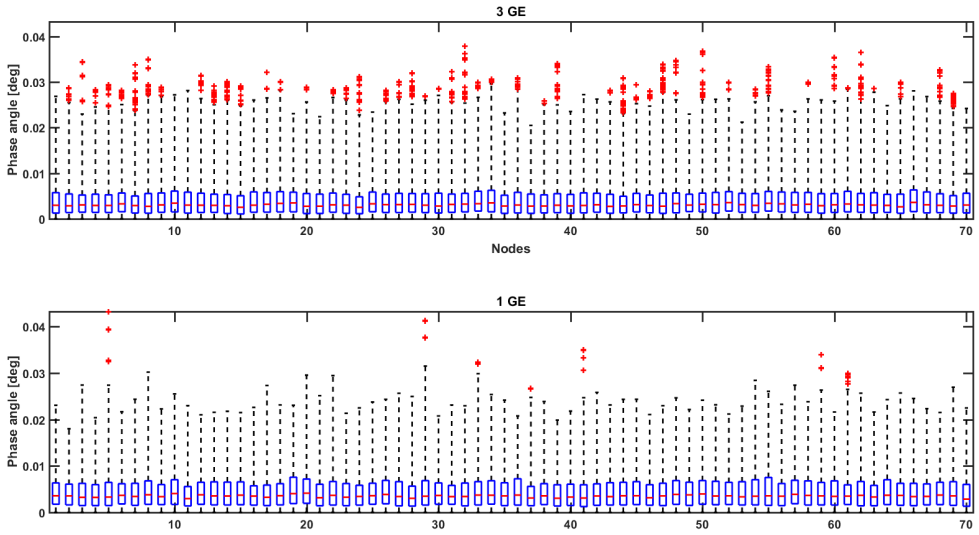


Figure 4.61: Voltage phase angle errors for every node including all noises.

In terms of maximum voltage error, 1 GE device produces slightly higher errors in time-stamps of really high consumption. But, overall, figure 4.62 shows that the errors are quite comparable. Similar results are estimated for current of branches, as noted in figure 4.63. Moreover, the mismatch of phase angle is minimal, as shown in figure 4.64.

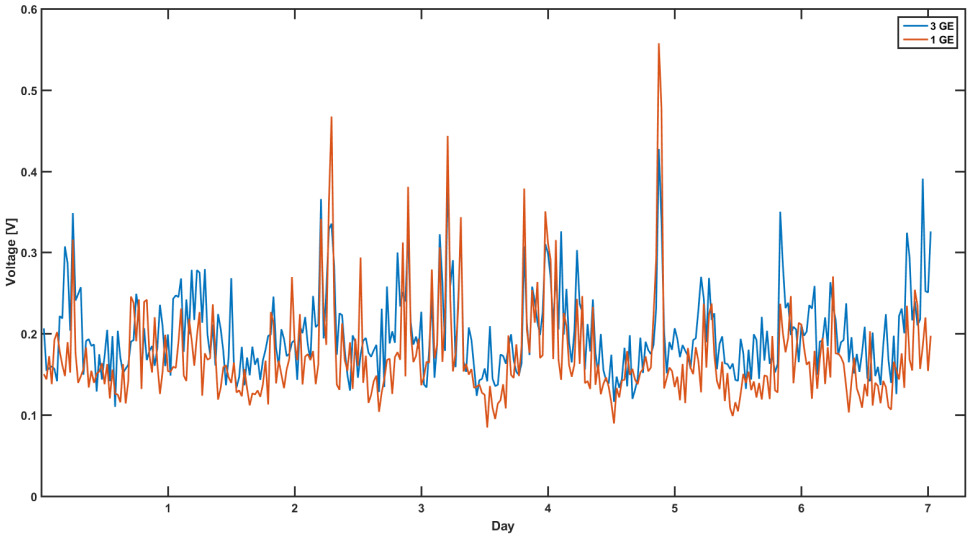


Figure 4.62: Maximum voltage magnitude errors for every time-stamp.

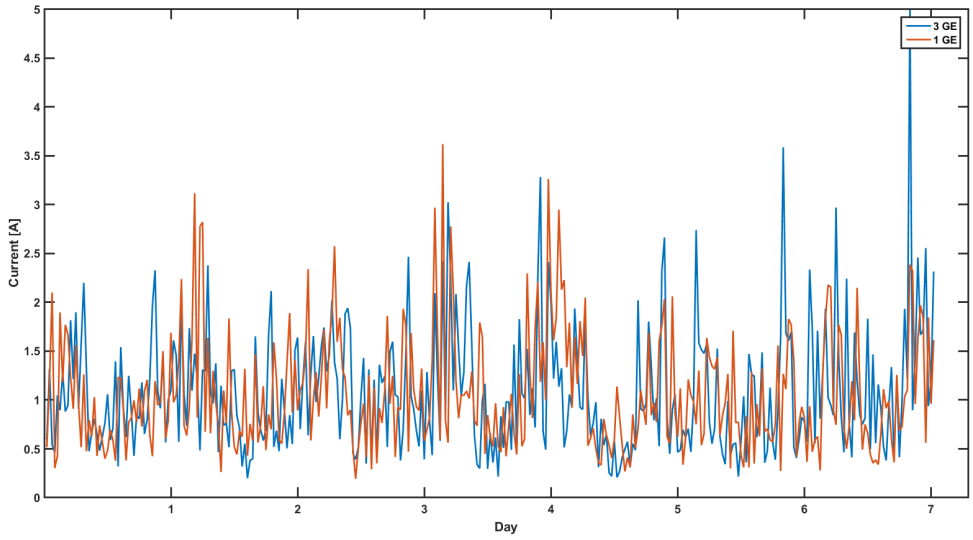


Figure 4.63: Maximum current amplitude errors for every time-stamp.

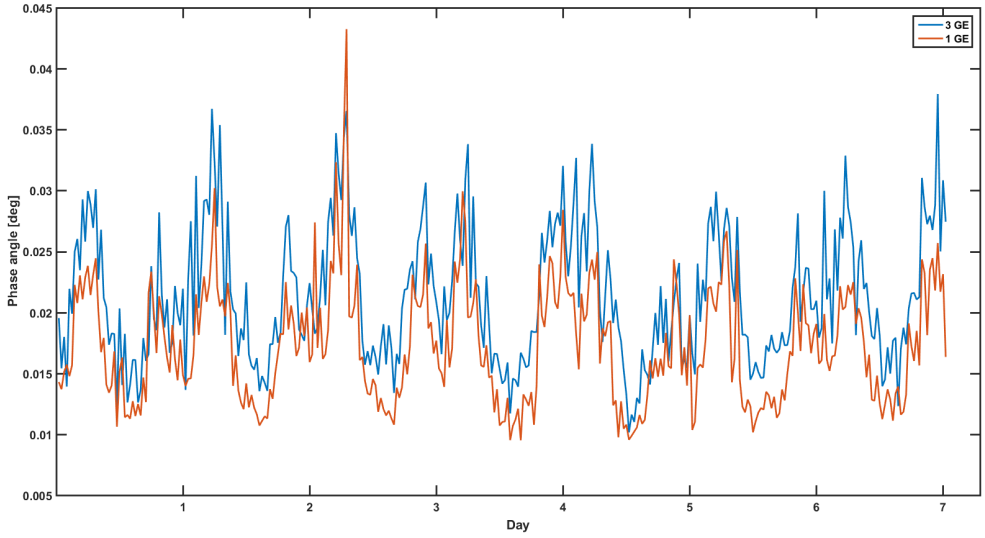


Figure 4.64: Maximum voltage phase angle errors for every time-stamp.

The distribution of voltage errors is shown in 4.65, where the normal distribution around a mean value of 0 is kept for the voltages for both examined cases. The limits slightly differ, but overall they follow the same pattern. Similar outputs are shown for current and phase angle, as noted in 4.67 and 4.66.

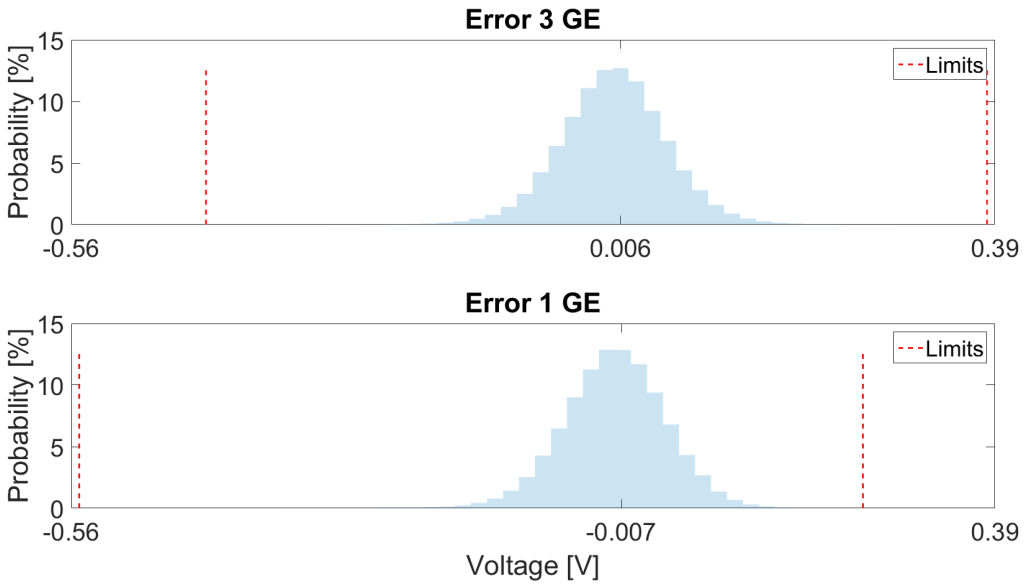


Figure 4.65: Histogram of voltage magnitude error distribution for different number of GE devices.

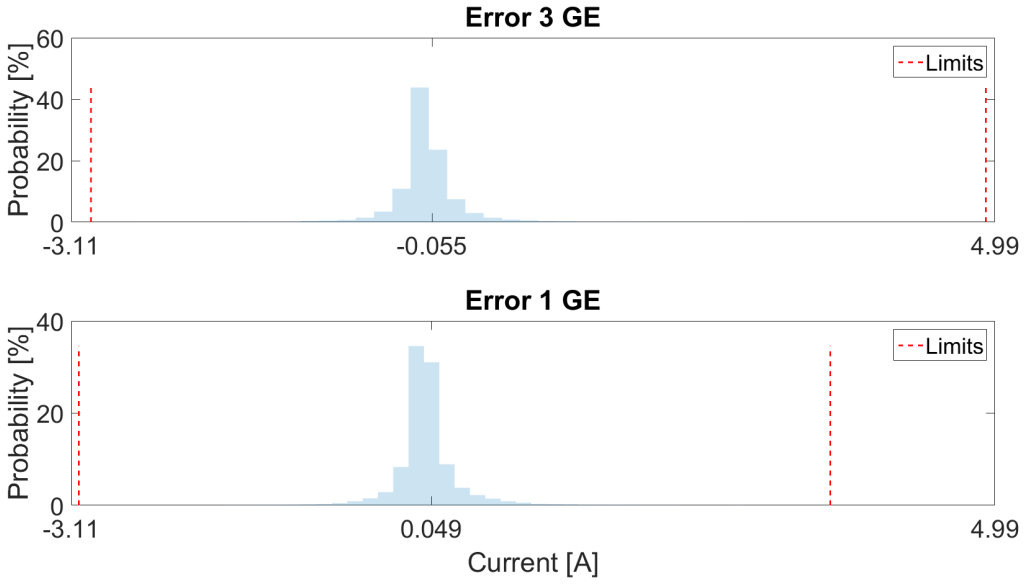


Figure 4.66: Histogram of current amplitude error distribution for different number of GE devices.

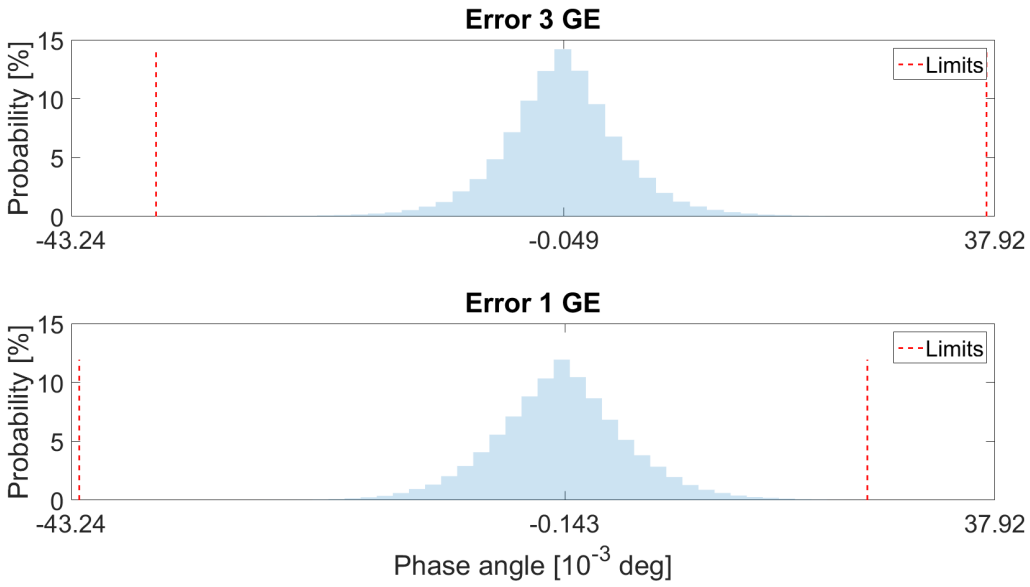


Figure 4.67: Histogram of voltage phase angle error distribution for different number of GE devices.

It's shown that 3 GE devices do produce similar results in both cases. Thus, for complete network knowledge, using more devices could be rendered useless. Nevertheless, more devices are useful for cases of less network knowledge, where more assumptions for the load distribution must be made.

4.3.6. Test case 2 - Numerical comparison of results

Based on the visual representation of the outputs, understandably WLS is better than LAV, and possibly using more devices is worthy in terms of outputs. For mathematical comparison, the Root Mean Square Error was used. RMSE is a quadratic scoring rule that measures the average magnitude of the error. It's the square root of the average of squared differences between prediction and actual observation. Since the WLS optimization's goal is the RMSE, it is expected to have inflated results for this method, compared to LAV. Still, since the WLS is superior thus far to the LAV, this mathematical approach is deemed acceptable. The results for method comparison are shown in the table 4.2.

		Ideal	3 GridEye	1 GridEye
WLS	*V* [V]	1e-05	0.01450	0.02320
LAV		1e-05	0.06052	0.11500
WLS	*I* [A]	1.3e-04	0.15718	0.19734
LAV		3e-04	0.74165	1.23219
WLS	*Vph* [deg]	0	1.00E-05	2.00E-05
LAV		0	5.00E-05	0.00011

Table 4.2: Numerical error comparison of WLS and LAV for scenarios 1,2 and 3

A similar approach is done to identify the difference in the use of number of GE devices, with and without noise. In this scenario, the use of 2 GE devices was also examined, but due to the similarities in the outputs, the visualization of the outputs is rendered unnecessary. For this case, combinations of usage of GridEye devices are made. Similar to the case of using just a single GridEye device to the network, two GridEye devices are used. Always the GE installed on the transformer is kept, so the combinations are made for the other 2 devices. The possible scenarios are thus:

1. All devices, which will be called **3 GridEye**

2. Two of them using GridEye devices on transformer and node 3 as seen on figure 4.23, which will be called **2 GridEye (1 + 3)**

3. Two of them using GridEye devices on transformer and node 29, which will be called **2 GridEye (1 + 29)**

4. A single GridEye device on the transformer **1 GridEye**

The outputs are seen on table 4.3. Since, as mentioned, the disadvantages of LAV became obvious, the comparison was done only by WLS methods.

		3 GridEye	2 GridEye (1 + 3)	2 GridEye (1 + 29)	1 GridEye
noise	*V* [V]	0.05872	0.06439	0.06439	0.05785
noiseless		0.01450	0.02291	0.01533	0.02320
noise	*I* [A]	0.32057	0.32923	0.32923	0.31104
noiseless		0.15718	0.19345	0.16295	0.19734
noise	*Vph* [deg]	9.00E-05	0.00012	0.00012	0.00010
noiseless		1.00E-05	2.00E-05	1.00E-05	2.00E-05

Table 4.3: Numerical comparison of the impact of GridEye devices on errors, scenarios 3 and 4

It's obvious that the amount of GE increase the performance ratio of the algorithm. Still, one GE is capable of producing quite good results in terms of outputs.

4.3.7. Test case 2 - Bad data injection in smart meters, both methods used (Scenario 2)

As mentioned on chapter 3, the optimization is always prone to bad data errors. These become even more critical in networks with low information amount, as the low-voltage networks. To this end, sometimes bad data analysis is either too burdensome computationally, depending on the number of data. In this section, a bad data detection, identification and correction algorithm was used. It is worth emphasizing that the bad data analysis is always after the state estimation outputs are calculated, and will require re-runs of state estimation in the event of existence of bad data.

To validate the algorithm, both methods were used at a specific time, due to high computation costs of doing it for every different time-stamp. Since the most prone to error measurements are the Smart meters, the error detection is only for smart meters. Thus, for this case, the GridEye devices are deemed are the most trustworthy devices in the network.

The insertion of bad data in smart meter measurements is done for three scenarios for every device, thus there will be 41×3 cases. These three scenarios are:

- Using 0 instead of the value of power the SM device.
- Using 10 times the value of power of the SM device.
- Using the negative value of power of the SM device.

It must also be mentioned that statistical data of the smart meter devices at a specific point are used, to guide the choice of the largest normalized residual. This is done as the estimator can not understand completely which device could be the erroneous one.

Scenario 1: Zero output of Smart meter device

In this scenario every device's active power is set to zero. This is done for one device at a time. The outputs for voltage magnitude and current amplitude are shown in figures 4.68 and 4.69.

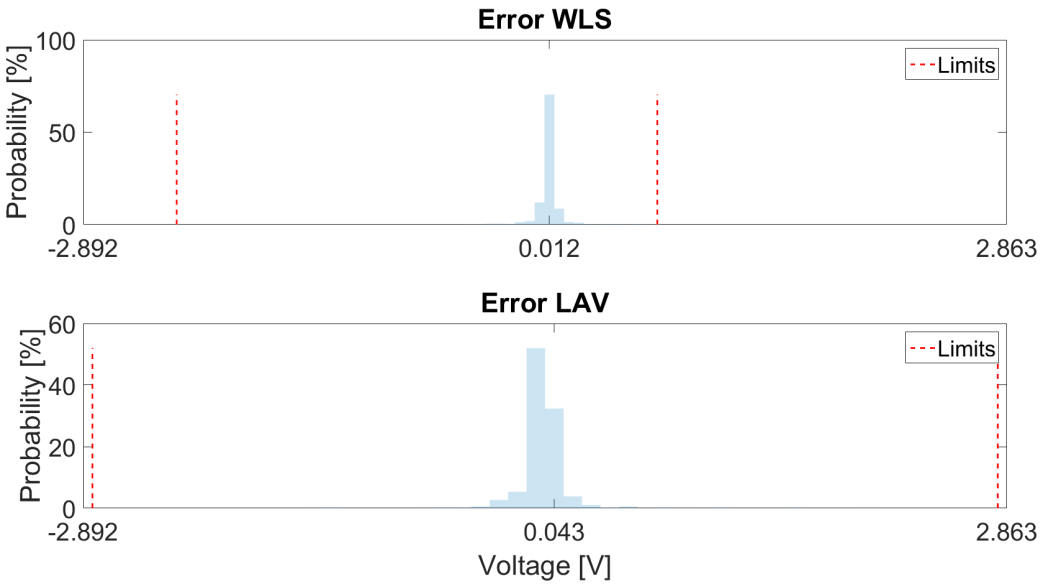


Figure 4.68: Histogram of voltage magnitude error in the event of bad data for both methods.

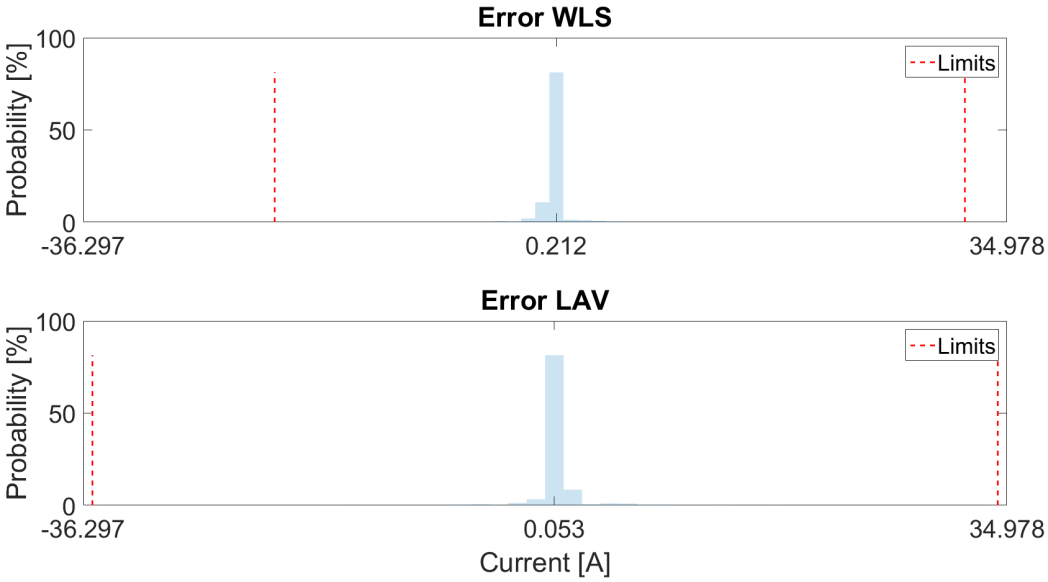


Figure 4.69: Histogram of current error in the event of bad data for both methods.

As it's obvious the error is not only higher, but under no circumstance does it provide quality state estimates. While still performing slightly better, WLS does not provide quality outputs, as the maximum error for voltage and current are close to 2.5V and 30A respectively. The outputs after the bad data algorithm are provided in figures 4.70 and 4.71.

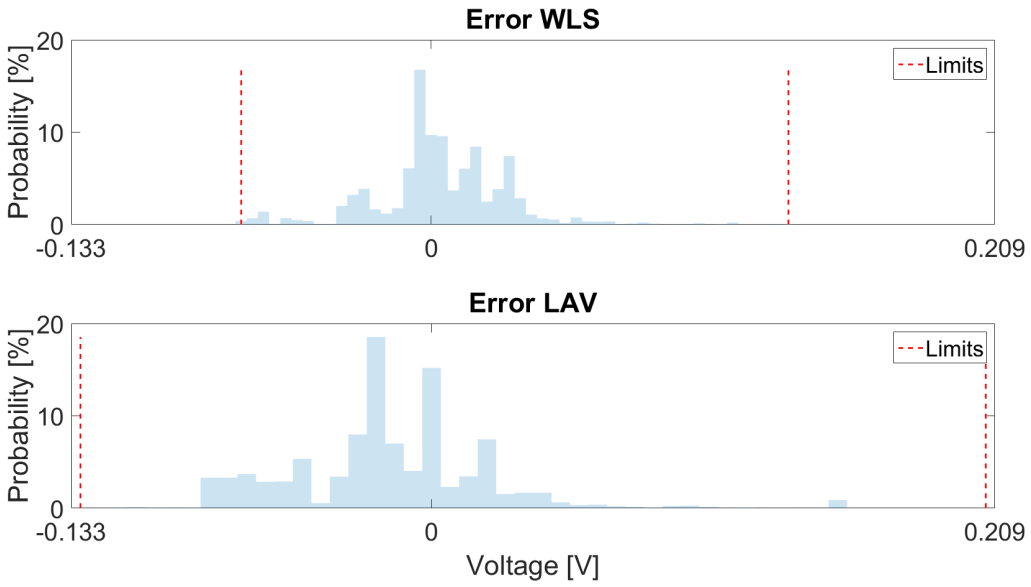


Figure 4.70: Histogram of voltage magnitude error in the event of bad data for both methods after bad data algorithm.

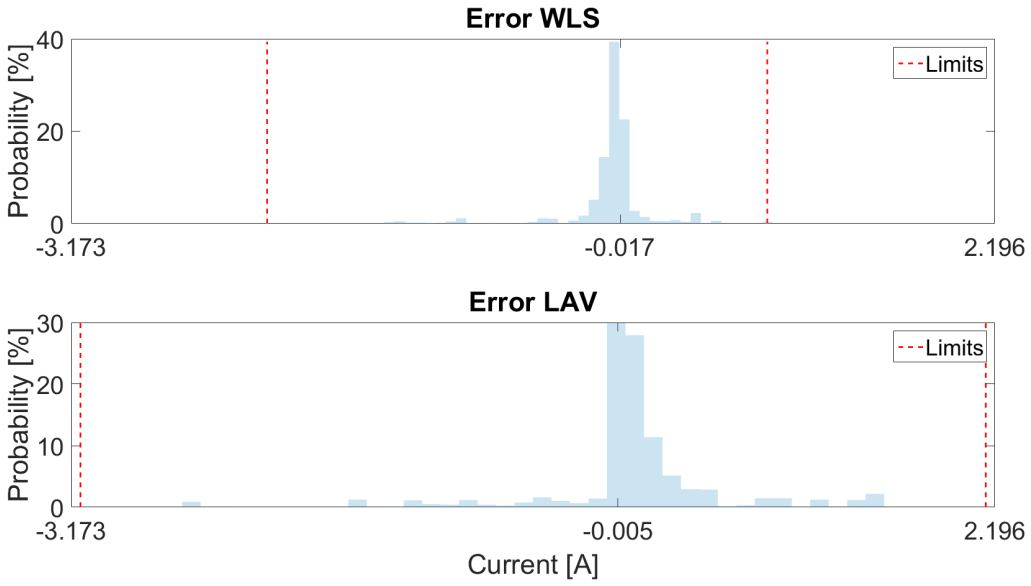


Figure 4.71: Histogram of current error in the event of bad data for both methods after bad data algorithm.

It can be easily seen that both voltage and current remain within the expected range for the whole network. Still WLS outperforms the LAV estimator on this case.

Scenario 2: Ten times the output of Smart meter device

In this scenario every device is set to ten times the initial input, and still one at a time. Normally, the outputs prior to the bad data correction are provided. The outputs for voltage magnitude and current amplitude are shown in figures 4.72 and 4.73.

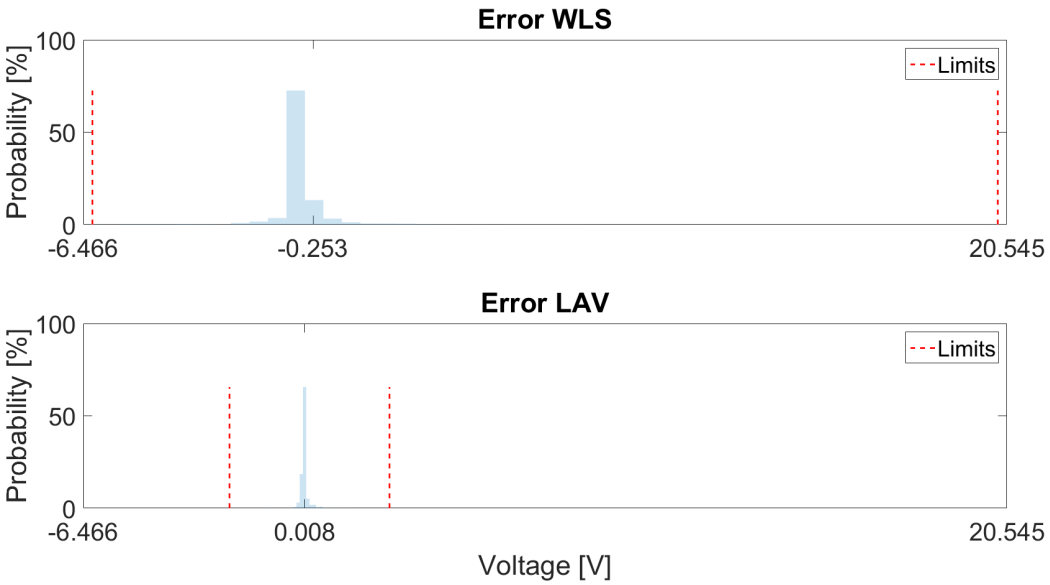


Figure 4.72: Histogram of voltage magnitude error in the event of bad data for both methods.

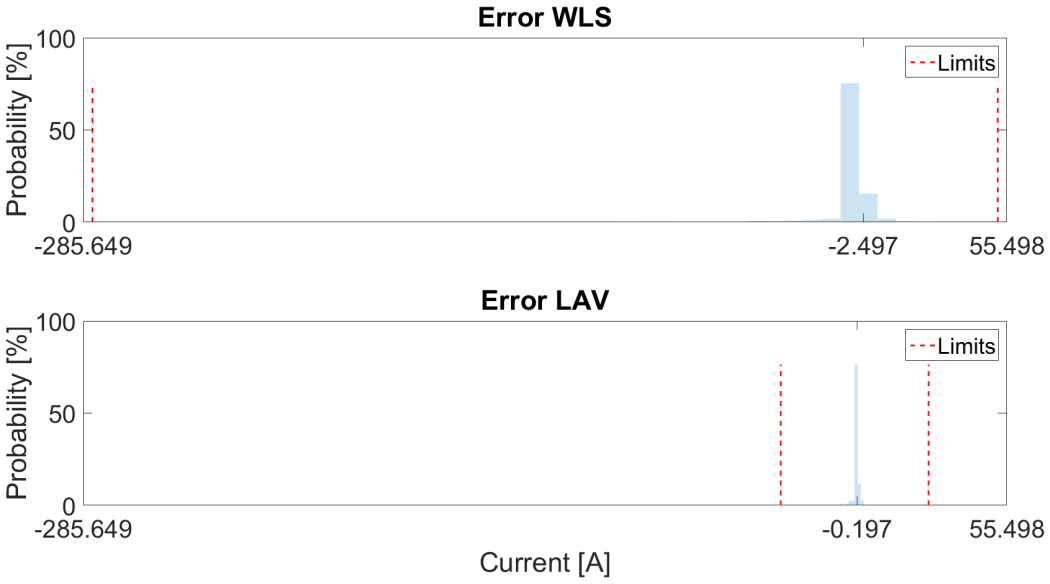


Figure 4.73: Histogram of current error in the event of bad data for both methods.

It is noted that both estimators fail to adapt to such a high error, thus providing terrible values for both current and voltage. It is also noteworthy that LAV provides better outputs at this scenario, which is expected as it's a more robust estimator to bad data. The outputs of the bad data algorithm are shown in 4.74 and 4.75.

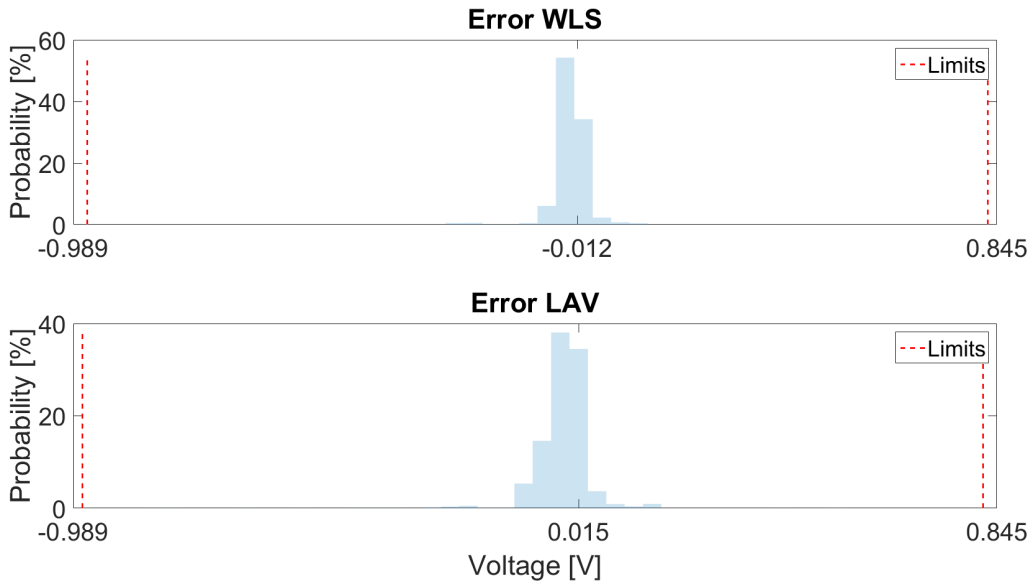


Figure 4.74: Histogram of voltage magnitude error in the event of bad data for both methods after bad data algorithm.

Figure 4.75: Histogram of current error in the event of bad data for both methods after bad data algorithm.

The difference is notable, as both methods provide the similar outputs for this scenario. While most errors remain around 0, some extreme cases where the algorithm is unable to identify which smart meter has bad data provides some higher error for voltage and currents. Still, the network's operating conditions are fairly met, despite the large looming error of input.

Scenario 3: Negative output of Smart meter device

In this scenario every device is set to negative the initial input, and still one at a time. This scenario also is important to note the case of production, since the test network was only consuming Normally, the outputs prior to the bad data correction are provided. The outputs for voltage magnitude and current amplitude are shown in figures 4.76 and 4.77.

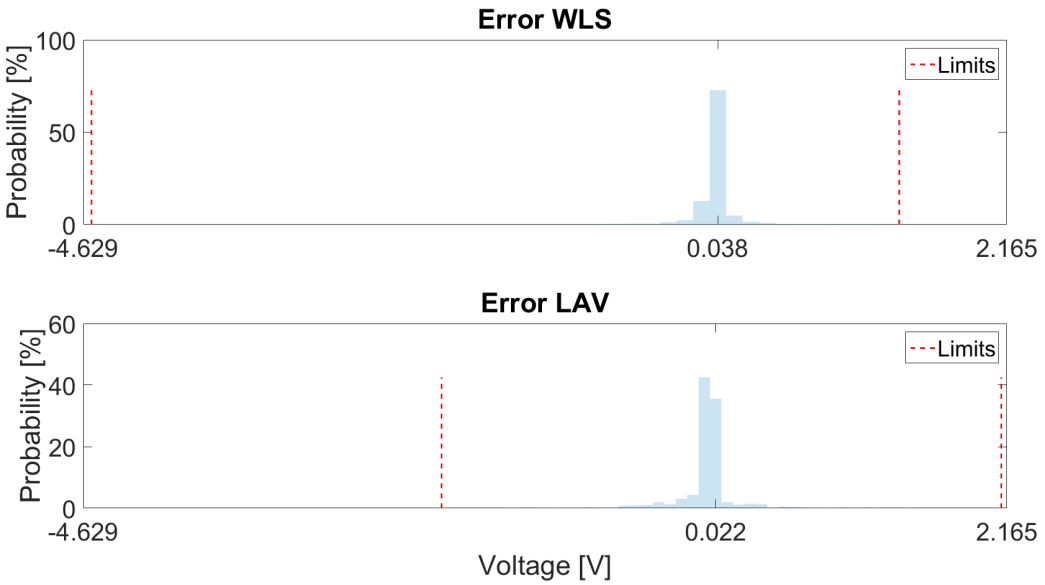


Figure 4.76: Histogram of voltage magnitude error in the event of bad data for both methods.

Figure 4.77: Histogram of current error in the event of bad data for both methods.

It's quite obvious that both estimators fail to adapt to such a high error, providing error values similar to the first scenario for both current and voltage. Still, as the second scenario, LAV provides better outputs at this scenario, which is expected as it's a more robust estimator to bad data. The outputs of the bad data algorithm are shown in 4.78 and 4.79.

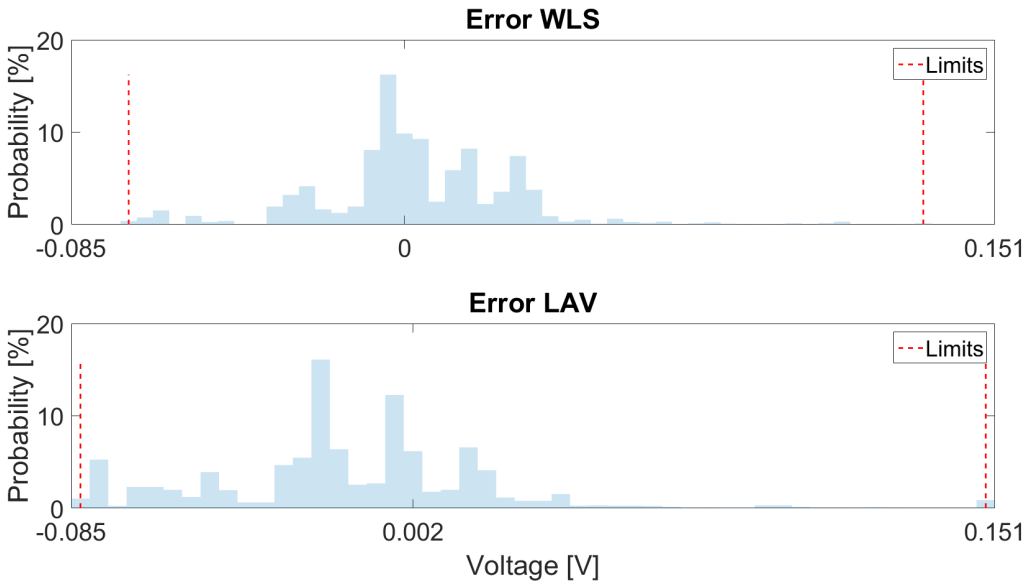


Figure 4.78: Histogram of voltage magnitude error in the event of bad data for both methods after bad data algorithm.

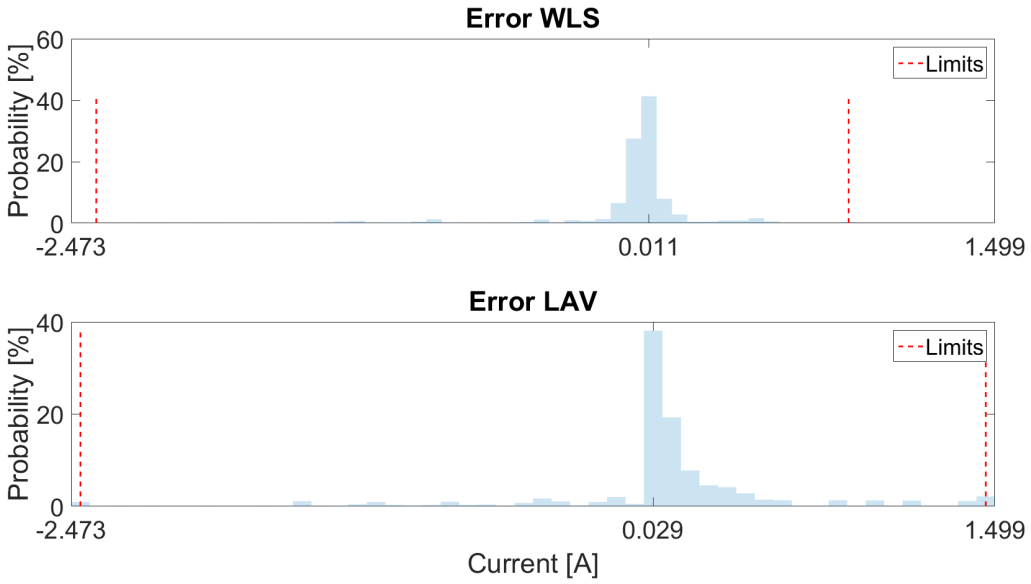


Figure 4.79: Histogram of current error in the event of bad data for both methods after bad data algorithm.

The difference is notable, as both methods provide the similarly good outputs for this scenario, similar to the first one. The maximum and the mean errors do remain within some good margins. The network's operating conditions is correctly depicted, despite the error of input.

5

Conclusions

This chapter contains all the outcomes of the research analysis for this project. The visual and numerical outputs are available in Chapter 4, but the outputs with respect to the questions posed in Chapter 1 are discussed in this chapter.

5.1. Summary of the project

The goal of this research was to create a linear state estimation algorithm in the low-voltage network. The transition from state estimation in the transmission grid to the distribution grid is no easy task due to lack of sufficient and accurate measurements to correctly represent the operating conditions of the low-voltage network. Currently in the distribution grid, at most, only transformers are monitored due to their importance, as well as the general monitoring of the specific low-voltage grid. Thus, an observability issue ensues in low-voltage networks. The rise of smart meters led to research of their use further than customer visualization devices. An idea to overcome the issue of lacking measurements in the low-voltage network is the use of smart metering devices. This mentioned idea is still raw, due to many hurdles in such an implementation. In this project, lower-resolution measurements (smart meters), as well as SCADA-type measurements of higher-resolution (GridEye data), were used to calculate a state estimation. Generally, what was achieved in this project was:

- A linear state estimation algorithm for the distribution grid was developed, which will be used for the needs of the measurement provider, DEPsys S.A.. The algorithm, which constitutes of several functions, from network creation, network reduction, measurement correlation and power flow comparison, achieved good computational performance and accuracy. Its main advantage is, that it is a fast process, able to cope with different types of measurements with variable number of inputs. Moreover, the algorithm was created and run for two optimization methods, based on WLS and an LAV one, which have different optimization goals. WLS outperformed the LAV method. It is important to note that all voltage errors were **lower than 0.14%** of the inputs provided for the WLS method.
- The total aggregation of smart meter devices was matched every 30 minutes to the GridEye device allocated in the transformer node. This correlation helped generate assumed reactive power outputs for the smart meters, based only on the active power provided by the smart meter and the GridEye device on the transformer. Thus, profiles were created, by correlating two different types of devices, to estimate a reactive power consumption throughout the day for the smart meter devices. This, in turn, was useful for the state estimator, as smart meter currents were calculated through that process.
- An analysis was made for the effect of the number of GridEye devices, with respect to the errors of the state estimator. For the state estimation problem, it was found that more GridEye devices enhance the outputs, but the degree of enhancement is not by a large margin.

- Inserting noise didn't affect the outputs more than the level of the noise. This means that the noise does not affect the outputs, more than the inherent possible errors of the used devices.
- Bad data analysis for the state estimator was performed. All state estimators face the problem of bad data, especially when dealing with smart meters. In this analysis, an algorithm was developed to detect, identify and correct the bad data possibly provided. The output of this analysis is that historical data are necessary to perform such an algorithm. The residuals that exist due to the voltage assumption of the smart meters, as well as the lack of redundancy, are obstacles in the algorithm's ability to identify the problematic injection. Thus, the identification process is aided by historical information. The correction process does not reach the values of the case of no bad data injection, which is a direct correlation of the existing residuals due to voltage assumptions of smart meters. Nevertheless, the state estimation produces accurate outcomes.

5.2. Research questions

As mentioned 1.4, the main research question that was asked was **whether such an algorithm is applicable for the low-voltage network and what are the errors produced for the linear state estimation algorithm under assumptions for the smart meters**.

- In the first test case, which consisted only of GridEye devices, the errors were minimal when compared to the power flow outputs. Only some minor mismatches were noticed for voltage magnitude, current amplitude and voltage phase angle. These results were promising, since that meant for a reduced network, the state estimation could fully adapt to the provided data. Nevertheless, the case scenario always requires input of noise, due to the inherent errors that could be provided by the measurements. The noise levels inserted, which were normally distributed with the maximum error based on the levels of trust of GridEye devices, proved that noise does not get amplified by the algorithm, since the maximum errors were in the range of error introduced by the noise inserted. Essentially, the output errors were not magnified by the noise, which meant that the first test-case produced high quality outputs. Overall, the first test-case, which resembled a network of high-redundancy, produced similar outputs to the power flow.
- In the second test case, which consisted of both GridEye and smart meter devices, a test-case scenario was first implemented, which constituted of the active power and voltage of the smart meters. These values were provided from the power flow outputs. Essentially, this artificially created test-case created a higher-redundancy, to validate the degree of successful operation of the algorithm. This was the first examined scenario, this process helped identifying problems when running the algorithm and limitations. Essentially, voltages of smart meters were only used for the current calculation of smart meters. It was proved that the outputs were completely matching the desired power flow outputs for all voltage magnitude, current amplitude and voltage phase angle. This meant that, in cases where voltage or current of smart meter devices is provided, the outputs should perfectly matching the power flow ones.

  In the real case, where no voltage of smart meters was provided, and only the active power of every smart meter was included, the outputs had a higher error, as expected. This error naturally occurred, since the calculation of current, which is the necessary input for the measurement function, required a voltage estimation of the smart meter. To reduce the error, the estimated voltage of the smart meters was with respect to the closest GridEye device, which as previously mentioned, provide voltage magnitude. For this linear state estimation, in terms outputs, accuracy must be traded, when observability is not met (lack of availability of all voltages and currents throughout the network). Nevertheless, the errors were minimal compared to the operating point of the network for every specific node and branch. This trade-off of accuracy to computation time was characterized as a successful scenario, since the errors were minimal for all the monitored values (voltage magnitude, current amplitude and phase angle).

Based on the above, it is noted that the smart meter assumption requirements for a linear implementation of the state estimation in the low-voltage network do not hinder the state estimation's accuracy. The trade-off of low-accuracy errors for linearity is important for bigger and more complex networks.

Addressing the main scientific question involved the answering of the sub-questions that were posed in Chapter 1. These scientific questions and their answers are summarized below:

- **Which state estimation algorithms can be considered, that satisfy both high precision and computation time?**
  As mentioned thoroughly in chapter 2.4, the most prominent methods for state estimation implementation in power systems are the WLS, LAV and the SHGM methods. Overall, the choice of the method is dependent on the measurement availability. It was concluded that WLS method outperformed the other two methods in the absence of PMU data for transmission systems.

- **How can the selected state estimation algorithm adapt from the transmission network that they are initially created for, to the distribution grid?**
  State estimations were designed for the transmission network of the power systems. High-redundancy is absent from distribution network. Should the basic observability condition be met, it was found out that assumptions can help the algorithm produce results with high accuracy. This can be achieved by adjusting the weights in the WLS method, not only by the measurement device trust level, but also by the degree of knowledge available about that specific device. Thus, this forces the optimization to effectively ignore measurements that are assumption-based.

- **How does the algorithm deal with different redundancy levels?**
  The low-redundancy case, which is the main test scenario studied, consisted in the lack of voltage knowledge of the smart meter devices. As previously mentioned, the smart meter voltage was necessary for the current calculation of the smart meter devices. The algorithm was proven to be matching the power flow outputs when the smart meter voltage was provided, but in the case of no voltage knowledge errors of low margin occurred. The maximum errors, which were low compared to the operating point (0.24V and 2.2A), proved that the lack of high redundancy for state estimation is a trade-off between the linearity and accuracy. Since linearity creates a problem of certain convergence, the errors were deemed minimal compared to this provided advantage. What this also ensures, is that in cases where the DSOs could provide further information about the smart meter devices, the outputs will also be enhanced. Nevertheless, with only active power provided, the rest of the assumptions created a solvable linear problem.

- **How the objective function of the state estimation influences the outputs of the estimator?**
  Essentially, the objective function defines the method used for the state estimation. Although the advantages of each method used for state estimation has been studied a lot in literature, this research aimed to also differentiate between the advantages of two different methods, both based on the same inputs. The WLS problem was linear, since the matrix involved were linear, but the LAV optimization is still an iterative procedure, despite the linear measurement function. Nevertheless, linear measurement function reduces the computational burden. *For one run case*, after initialization, **the solving time for the WLS and LAV was 0.02 and 0.06 seconds respectively**. The research aimed to use same inputs for both methods, which meant that both methods could be directly correlated with respect to their outputs. Due to the fact that LAV is iterative procedure by a CPLEX solver, the computation time outcome was expected. The question lied in which method produced better outputs, for all the test cases, to examine whether the LAV could outperform the WLS one. In the first case, the error margins were too similar for voltage magnitude, current amplitude and phase angle. For the second case, in the ideal scenario, the errors were even smaller than the WLS one, despite the fact that WLS were deemed negligible. Nevertheless, in the rest of scenarios for the second test-case, the outputs of LAV, in terms of maximum error and distribution

error, were worse for all scenarios. This means that the LAV method, performs worse in scenarios where voltage assumptions must be made. Thus, since LAV has a higher computational burden and the outputs were worse than WLS for cases of knowledge of only active power of smart meters, LAV method was deemed inferior.

- **How does redundancy in the higher-resolution (GridEye) devices affect the state estimation?**
  Another question that rose, was the effect of the number of GridEye devices in the state estimation outputs. Essentially, do more devices used reduce the errors, as expected, and if yes, how much does the error change. This is a research question related to the interests also to DEPsys S.A., since the number of devices necessary to have good operating knowledge of the network is necessary. To evaluate this, more scenarios were created for both test cases.
  - In the first case, taking advantage of the GridEye's ability to measure multiple current in many phases, a case was considered where only two of the four devices were only used as inputs for the state estimation. For this case, due to still having sufficient redundancy, the outcomes were similar.
  - In the second case, three GridEye devices were used. Thus, the scenarios consisted in creating different combinations of the GridEye devices as inputs. As it was shown, the errors were higher the less GridEye devices were used, especially in a noiseless environment. This is expected, as GridEye devices contribute to further inputs in the state estimation. In the noisy environment, in some cases using more devices created higher errors, which is a result of conflicting outputs in the estimator. But, overall, one device used in the transformer could produce sufficient results as outputs, where the errors are deemed as low-enough. The case occurred for the LAV, where having a single devices was inferior to having more devices. Generally, for the LAV, the number of GridEye devices affected more the errors than the WLS one.
- **How does the algorithm deal with bad data?**
  Generally, all mathematical optimization methods are prone to errors, when bad measurements are provided. Since the smart meters are aggregating the consumed power within a time period, these were deemed as the most prone to bad data errors, and thus only smart meters were introduced to errors. This was proved in the figures 4.68,4.72 and 4.76 for voltage, where just one device could shatter all the outputs of the state estimation. The most important aspect of bad data analysis was deemed to detect and identify the position of the error. For the detection aspect, a manual threshold was used, based on observations of errors. Manual selection of threshold could lead to weak or strong thresholds, that either ignore or misinterpret data. This also means that the threshold should be defined based on the specific network and the devices used, and the redundancy of the specific network. Thus, a prior analysis is necessary for such an algorithm, to correctly detect bad data with a high degree of trust.

  Following the detection, identification proved to be a tricky point in the bad data analysis. As mentioned, the lack of redundancy, in a network with assumptions (due to the voltage assumption for smart meters), creates a problem in identification of the possible node of error. To this end, historical data of the smart meter devices based on a specific time point during the day were used. What this achieved is creating a range of possible values for the smart meter device, which if exceeded, could mean that this specific smart meter is more likely to be part of the bad data. While not ideal, this idea proved to be an asset in the identification process.

  Bad data correction, despite the voltage assumptions of smart meters, corrected the bad data to a sufficient degree. Of course, the assumptions for the voltage affect also the correction part of the detection process, although in most cases the correction was of high accuracy.

  Overall, the bad data analysis process was quite successful, as all three parts of the bad data analysis were met in different degrees.

5.3. Future work

Further improvements can be implemented in the algorithm to enhance the outputs under different scenarios. These improvements, which will not be part of the project, are subject to discussion with the DEPsys S.A., to decide the possible next goals for the expansion of the algorithm.

- **Better nodal voltage assumptions for smart meters.** It was shown, that when the smart meters voltage was available, just to calculate the current of the smart meter devices, the output errors were close to 0. This means that, for the real test case, where voltages are not available, linearity came to the cost of the accuracy. Some ideas to enhance the algorithm could be focused on making better voltage assumptions about the smart meter devices, based on probable historical data with respect to the transformer measured outputs. In this algorithm, the voltage used was corresponding to the closest GridEye device, which inherently creates an error in the estimator. Geographical position and historical data are considered, to further improve the outputs of the estimator.

- **Use more constraints in the LAV estimator**, to produce even better results, to possibly match or even overcome the WLS method in the cases where assumptions were made. Based on the outputs, it is quite obvious that LAV suffers more under uncertainty than the WLS. While the estimator treats weights in a similar manner, the optimization goal of WLS proved better for this scenario. As it was obvious from the ideal scenario with the smart meters, the LAV produced similar results with the WLS method. This means that, uncertainty, under circumstances, could not be improved with some constraints, that could further push the algorithm towards better results, since LAV is an optimization method with manually implemented constraints. Ideas on equations that could help the LAV were discussed, but to this point no concrete outcomes were achieved.

- **Probabilistic state estimation** is an already discussed idea, due to the nature of smart meter devices. Generally, in real networks, not every house is outfitted with a smart meter device. As such, smart meter devices installed alone are not able to create a state estimation problem under certainty or uncertainty. This, combining that with the fact that they are not always trustworthy or available during the day, prove that adjustments can be made to algorithm to operate on real case scenario with a higher degree of uncertainty. The probabilistic or stochastic scope aims, with a percentage of smart meter devices in specific points used, to create profiles for the rest of smart meter based on historical data, annual consumption and correlation to the aggregate data of the transformer. A similar idea is shown in [30], to create state estimation runs at times different that the provided data, under some uncertainty. To achieve this, data mining processes are necessary, to create clusters and profiles for the unknown smart meters. Then, correlations are necessary for time-stamps during different days to create these profiles, based on Markov-chain models. This is a process of high importance, and due to the nature of the work needed for such a case, it is deemed outside the scope of this report.

- **Enhancing the smart meter measurements for shorter time periods** based on GridEye provided data. Currently, the 10 and 15 minute window of smart meters and GridEye devices respectively, creates the issue of when should the state estimation be performed. In this research, the matching time of 30 minutes between the devices was used. For this time window, smart meter devices provide the aggregate consumption, meaning that there is no information of the processes that happen through this period. As such, the idea of optimal matching based on the GridEye devices data occurred. An idea to achieve this is through data fitting under uncertainty [31]. This still requires total knowledge of the network, optimal matching can be implemented in collaboration with probabilistic state estimation under uncertainty. The goal would be to create a state estimation problem, irrespective of the time window of the provided smart meter devices.

- **Adapt the algorithm to medium voltage and meshed networks**. The current algorithm was created for low-voltage network, assuming a radial network. Further analysis is required for the meshed network, as it is slightly more complex than the radial one, and requires some further attention. Moreover, although low-voltage networks require smart meters for state estimation, medium-voltage ones with just GridEye devices should have enough information to create a state estimation problem and then solve it.
- **Re-run algorithm in the case where only active power is available for smart meters.** It was noted, that the errors of the estimators occur in the current calculation for the smart meters, as the equation requires voltage knowledge. The estimation provides good outputs for voltage, but the outcomes can be even more accurate if the state estimation is re-calculated for the newly found voltages. This way, such errors are avoided. Obviously, the main issue is that the algorithm is run two times, but nevertheless, a case could be made for scenarios of the highest errors, if that is deemed necessary.

Bibliography

[1] F. C. Schweppe and J. Wildes, ”Power System Static-State Estimation, Parts I, II, and III. In *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, no. 1, pp. 120-135, Jan. 1970. doi: 10.1109/TPAS.1970.292678.

[2] A.Monticelli, *State estimation in electric power systems: a generalized approach*,vol. 507. Springer Science & BusinessMedia, 1999.

[3] Abur A., Exposito AG. Power system state estimation: theory and implementation. CRC press; 2004

[4] K. Dehghanpour, Z. Wang, J. Wang, Y. Yuan and F. Bu, ”A Survey on State Estimation Techniques and Challenges in Smart Distribution Systems,” in *IEEE Transactions on Smart Grid*, vol. 10, no. 2, pp. 2312-2322, March 2019. doi: 10.1109/TSG.2018.2870600

[5] `https://www.depsys.ch/grideye/`

[6] Ahmad, Fiaz & Rasool, Akhtar & Ozsoy, Emre & Rajasekar, S & Sabanovic, Asif & Elitas, Meltem. (2017). Distribution System State Estimation-A step towards Smart Grid. *Renewable and Sustainable Energy Reviews*. 10.1016/j.rser.2017.06.071.

[7] M. E. Baran and A. W. Kelley, A branch-current-based state estimation method for distribution systems, *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 483–491, Feb. 1995.

[8] W. M. Lin, J. H. Teng, and S. J. Chen, A highly efficient algorithm in treating current measurements for the branch-current-based distribution state estimation, *IEEE Trans. Power Del.*, vol. 16, no. 4, pp. 433–439, Jul. 2001.

[9] E. Farantatos, G. K. Stefopoulos, G. J. Cokkinides and A. P. Meliopoulos, ”PMU-based dynamic state estimation for electric power systems,” *2009 IEEE Power & Energy Society General Meeting*, Calgary, AB, 2009, pp. 1-8. doi: 10.1109/PES.2009.5275407

[10] H. Yuan, F. Li, Y. Wei and J. Zhu, ”Novel Linearized Power Flow and Linearized OPF Models for Active Distribution Networks With Application in Distribution LMP,” in *IEEE Transactions on Smart Grid*, vol. 9, no. 1, pp. 438-448, Jan. 2018

[11] J. Zhao, M. Netto and L. Mili, ”A Robust Iterated Extended Kalman Filter for Power System Dynamic State Estimation,” in *IEEE Transactions on Power Systems*, vol. 32, no. 4, pp. 3205-3216, July 2017. doi: 10.1109/TPWRS.2016.2628344

[12] A. J. Wood and B. F. Wollenberg, *Power generation, operation and control*. John Wiley & Sons, 2012

[13] Singh R. et al, Choice of estimator for distribution system state estimation, *IET Generation, Transmission & Distribution(2009)*

[14] H. P. Horisberger, J. C. Richard, and C. Rossier, A fast decoupled static state estimator for electric power systems,*IEEE Transactions on Power Apparatus and Systems*, vol. 95, pp. 208-215, Jan. 1976.

[15] A. Garcia, A. Monticelli, and P. Abreu, Fast Decoupled State Estimation and Bad Data Processing,*IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, pp. 1645-1652, Sept. 1979.

[16] T. E. Dielman, “Least absolute value regression: recent contributions,”*Journal of Statistical Computation and Simulation*, vol. 75, no. 4, pp. 263–286, 2005

[17] A.Monticelli, “Electric power system state estimation,” *Proceedings of the IEEE* , vol. 88, no. 2, pp. 262–282, 2000

[18] Mili L., Cheniae M.G., and Rousseeuw P.J., “Robust State Estimation of Electric Power Systems” *IEEE Transactions on Circuits and Systems*, Vol. 41, No. 5, May 1994, pp.349-358.

[19] L. Mili, V. Phaniraj, and P. J. Rousseeuw, Least median of squares estimation in power systems,” *IEEE Transactions on Power Systems*, vol. 6, pp. 511-523, May 1991.

[20] Chenxi Xu and A. Abur, Robust linear state estimation with equality constraints, *2016 IEEE Power and Energy Society General Meeting (PESGM)*, Boston, MA, 2016, pp. 1-5. doi: 10.1109/PESGM.2016.7741552

[21] Y. Lin and A. Abur, Robust State Estimation Against Measurement and Network Parameter Errors, in *IEEE Transactions on Power Systems*, vol. 33, no. 5, pp. 4751-4759, Sept. 2018

[22] A. S. Dobakhshari, S. Azizi, M. Paolone and V. Terzija, ”Ultra Fast Linear State Estimation Utilizing SCADA Measurements,” in *IEEE Transactions on Power Systems*. doi: 10.1109/TPWRS.2019.2894518

[23] J. J. Grainger, W. D. Stevenson, *Power System Analysis*, McGraw Hill Inc., 1994.

[24] E. Handschin, F. C. Schweppe, J. Kohlas, and A. Fiechter, Bad data analysis for power system state estimation, in *IEEE Transactions on Power Apparatus and Systems*, vol. 94, pp. 329-337, Mar. 1975.

[25] Y. Lin and A. Abur, A Highly Efficient Bad Data Identification Approach for Very Large Scale Power Systems, in *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 5979-5989, Nov. 2018.

[26] A. Monticelli and A. Garcia, Reliable Bad Data Processing for Real-Time State Estimation, in *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-102, no. 5, pp. 1126-1139, May 1983. doi: 10.1109/TPAS.1983.318053

[27] MATLAB version R2015b, The MathWorks Inc.

[28] `https://yalmip.github.io/`

[29] Hampel, Frank R.; Ronchetti, Elvezio M.; Rousseeuw, Peter J.; Stahel, Werner A. (1986), *Robust statistics, Wiley Series in Probability and Mathematical Statistics: Probability and Mathematical Statistics*, New York: John Wiley & Sons, Inc., ISBN 0-471-82921-8, MR 0829458. Republished in paperback, 2005.

[30] T. Zufferey, D. Toffanin, D. Toprak, A. Ulbig and G. Hug, ”Generating Stochastic Residential Load Profiles from Smart Meter Data for an Optimal Power Matching at an Aggregate Level,” *2018 Power Systems Computation Conference (PSCC)*, Dublin, 2018, pp. 1-7. doi: 10.23919/PSCC.2018.8442470

[31] Kreinovich, Vladik & Shary, Sergey. (2016). *Interval methods for data fitting under uncertainty: A probabilistic treatment*. Reliable Computing. Vol. 23. pp. 105-140.