Call H2020-CS2-CFP04-2016-02

Aircraft Electrical Model Simulation Identification and Fitting Toolbox

Project reference:

755332 - *AEMS-IdFit*

Deliverable 1.1

PRELIMINARY IDENTIFICATION & FITTING METHODS APPLICABLE TO ELECTRICAL MODELS

December 2017

PUBLIC

Subject: Deliverable 1.1

Document type: Public

Prepared by: UPC (Universitat Politècnica de Catalunya)
MCIA Research Center

This page is left in blank intentionally

Summary of tasks

1 Summary of tasks

According to the DOW, this deliverable must include the following

- State-of-the art of parameter identification and fitting methods
- Comparison of the abovementioned methods and preliminary selection of the most appropriate ones
- Preliminary selection of the key parameters to take into account in the SaberRD models for power generators, power converters, power cables, power loads and filters

The tasks associated to this Deliverable 1.1 are listed as follows

- Task 1.1. Bibliographical research on the state-of-the-art of identification procedures for electrical equipment in aircrafts including power sources (3 stages variable frequency starter generator, permanent magnet AC generator, AC generator + AC/DC converter and battery), power converters (AC/AC converters such as transformers and autotransformers, AC/DC, DC/AC, DC/DC and DC/AC converters), power loads (synchronous and asynchronous machines), power cables and filters. Participants: UPC-MCIA.
- Task 1.2. Define the key parameters for each type of electrical hardware and the performance of the most suitable parameter identification methods, including test procedures and data post processing from measurements. Comparison of the most appropriate methods. Participants: UPC-MCIA.
- Task 1.3. Bibliographical research on the state-of-the-art of fitting methods. Comparison of the most appropriate methods. Participants: UPC-MCIA.
- Task 1.4. Preliminary selection and validation of the most appropriate parameter identification and fitting methods by using existing experimental data. It will be done according to specified accuracy criteria by comparing simulation results and measurements. Participants: UPC-MCIA.

Specifications agreed in the kick-off meeting

2 Specifications agreed in the kick-off meeting

The next paragraphs summarize the types of electrical and electronic systems to be analyzed in the ***AEMS-IdFit*** project.

1.-Three-stage generator. Generator with voltage regulated control (synchronous machine with excitation) + 6 pulse rectifier (main DC source)+ regulation on DC side.

2.-Three phase inverter + PMSM machine (control type to be confirmed)

3.-Starter-generator (mix of 1 and 2)

4.-All types of DC/DC architecture converter (switching supplies, buckboost converter unit BBCU …) in the range 540VDC/28VDC,540VDC/5VDC or 540VDC/200VDC.

5.-Constant power load. Typically, a constant power load is a computer (150W) with an input DC/DC converter regulating a low voltage (<28VDC).

6.-Passive filters (input filters HVDC standard, output filters for inverters for EMI)

7.- Six-pulse three-phase rectifier.

Problem statement

3 Problem statement

The AEMS-IdFit (Aircraft Electrical Model Simulation Identification and Fitting Toolbox) project addresses the requirement for developers of future MEA aircraft to select technologies and to test, verify and validate electrical system architectures. The limitations of weight and space are important, so there is a need to test, verify and validate reliable, efficient and cost-effective architectures and a wide range of operating conditions.

The general objective of the AEMS-IdFit project is to adapt and develop generic models of aircraft electrical systems to reproduce the response from experimental measurements and manufacturers design data under different operating conditions, giving a full-representation of electrical systems dynamics. These models are based on the right knowledge of a set of parameters in such a way that, based on experimental data, reliable parameter identification and fitting methods will be selected and developed according to its complexity. Taking into account the wide range of the devices operating conditions, mapping algorithms will be developed as well, in order to generalize the parameter calculation taking into account the dependence of the parameters on the operating conditions. On the other hand, methods to optimize the parameter model updating process will also be developed.

In this context, the development to be achieved by AEMS-IdFit, will provide model updating methods for complex electrical simulation models as follows:

1. The adaptation, transformation and updating of a data-base of measured signals for all hardware components to be validated.
2. The adaptation, updating or development of SaberRD generic simulation models with the capacity of reproducing the real behaviour of the hardware components and systems from experimental measurements and design data. The source code of the SaberRD models will be modified to accommodate such models to the requirements of the identification and fitting methods developed and to improve the models to account for some physical effects not included in the original version of such models. These models will include:
   - Electrical power sources (three-stages variable frequency AC starter-generator and permanent magnet generators, AC generator associated to an AC/DC converter or DC battery systems)
   - Power converters (transformers, autotransformers, rectifier, DC/AC converter, DC/DC converters including buck-boost converters)
   - Power loads (synchronous and asynchronous machines)
   - Power cables (single-phase, three-phase, screened and unscreened)
   - Filters used in the above-mentioned equipment since they have an important impact on electrical network stability and power quality.
3. The analysis, selection and performance evaluation of accurate parameter identification and data fitting methods based on an optimization approach.

4. Development of a methodology for generalizing the parameter identification from a wide range of real operating conditions covering the operational spectrum of each device, by developing suitable mapping algorithms taking into account the dependence of such parameters on the operating conditions.
5. Implementation and integration of the parameter identification, fitting and mapping methods developed in the IdFit-Toolbox in the Saber RD platform.
6. Development of a user-friendly SaberRD-GUI (graphical user interface), taking advantage of the IdFit-Toolbox that includes all the functions developed in the project. Different parameter identification, fitting and mapping methods will be available, depending on the imported experimental data files. The identification and fitting tool will be tuned according the level of accuracy required for the simulation, optimizing by this way the time required to calculate the required parameters.

• Comparison of the abovementioned methods and preliminary selection of the most appropriate ones.
• Preliminary selection of the key parameters to take into account in the SaberRD models for power generators, power converters, power cables, power loads and filters.

3.1 Parameter identification methods

Parameter identification is an experimental methodology intended to determine the dynamics of a system by applying algorithms specifically designed to analyze the experimental data. However, the correct identification of parameters to accurately predict in a realistic way the behavior of the electrical systems present in MEA aircrafts is not an easy task due to the complexity of such systems and the variety of operating conditions. Therefore both the measurement system and the resulting identified parameters must have the ability to deal with the typical operating conditions, existing imbalances in the electrical systems involved, saturation effects or presence of harmonics among others.

Parameter identification methods are specially intended for white-box models.

Although in the technical bibliography different parameter identification methods are found, there are three main group of techniques applied can be classified as follows, which are **specially intended for white-box models** (see next section).

- Parameter identification methods based on **manufacturers' design data**
- **Offline** parameter identification methods based on measurements
- **Online** parameter identification methods based on measurements

According to the classification above, the parameters can be obtained from either the electrical system design data or through measurements, the latter ones can be done under both, offline or online operating conditions.

Parameter identification and estimation under dynamic conditions has been effectively applied to identify circuit parameters and machine parameters based on the measurement of

electrical magnitudes such as instantaneous voltages and currents. It is possible to do so using real time operating data[1]. However, **model parameters can depend on the operating conditions** and the identification of such parameters from experimental data can be done both offline or online either in the frequency-domain or in the time-domain.

3.2 Parameter fitting methods

- **Parameter fitting methods are mainly intended for grey-box models.**
- **Parameter fitting methods are applied to fit model parameters to experimental data. They apply optimization methods to find the best values of the parameters.**

The values of the parameters obtained are usually influenced by the objective function dealt with. Different objective functions can be defined, for example, to **minimize the sum of squares** (SoS) of the difference between the model output for a given value of the set of parameters and the experimental data acquired over several experiments[2]**or the root mean square error** (RMSE)[3].

3.3 White-boxes, black-boxes and grey-boxes

There are different approaches for system identification, which can be broadly categorized into **white-box,black-box** and **grey-box** modelling.

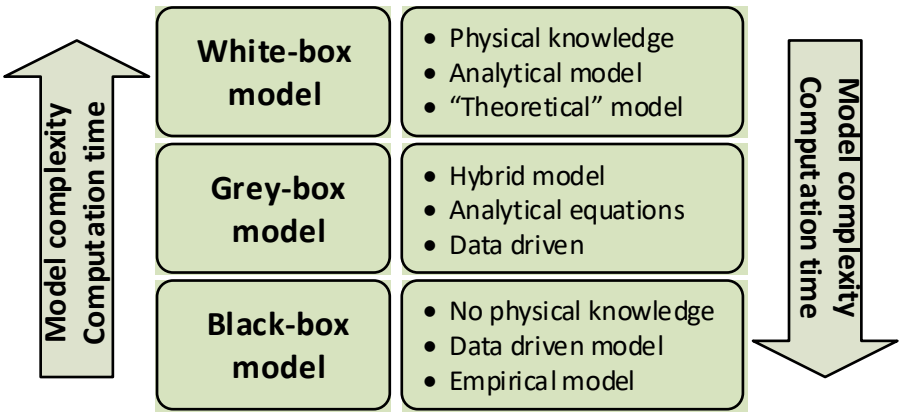


Fig. 3.1. White-box, grey-box and black-box modelling

White-box physical multidimensional models ensures a high genericity of the method by applying a series of algebraic and/or differential equations to characterize the system

[1]Y. Wehbe; L. Fan; Z. Miao, Least squares based estimation of synchronous generator states and parameters with phasor measurement units, North American Power Symposium (NAPS), 2012, pp. 1 – 6.

[2] Drayton Munster, Parameter Identification: A Comparison of Methods, July 19, 2009

[3] F. Ferracuti *et al.*, Data-driven models for short-term thermal behaviour prediction in real buildings, Applied Energy, 2017.

behavior, at the expense of a high computational effort. Contrarily, models directly derived from experimental data are called **black-box** models, and usually allow a great reduction of the computational effort, but have a strong dependence on available experimental data, which can severely limit their genericity. Therefore, **grey-box** approaches combining the advantages of both physical and empirical models might offer an interesting solution[4].

3.3.1 White-box models

- **White-box models are fully derived from known physical laws.**
- **All equations and parameters are determined by theoretical modeling.**
- **Models whose structure is completely derived from first principles fall under the white-box category, even if some parameters are estimated from data.**
- **White-box models do not depend on data (or only to a minor degree), and that their parameters have a direct interpretation in first principles.**[5]

White-box models are usually based on a specific set of equations that define the physical behavior of the analyzed system. White-box models represent theoretical models which require a profound physical "a priori" knowledge of the problem, providing a global structural representation of the system. The equations describing the models tend to be very complex, especially when the models are highly non-linear. Therefore, white-box methods assume a known structure for the electric/electronic system.

Although white-box models can reproduce system's behavior from a theoretical point of view, they sometimes present a limited scope, since they have difficulty in representing unknown noises, nonlinearities and unseen dynamics that often appear as lumped modelling errors.

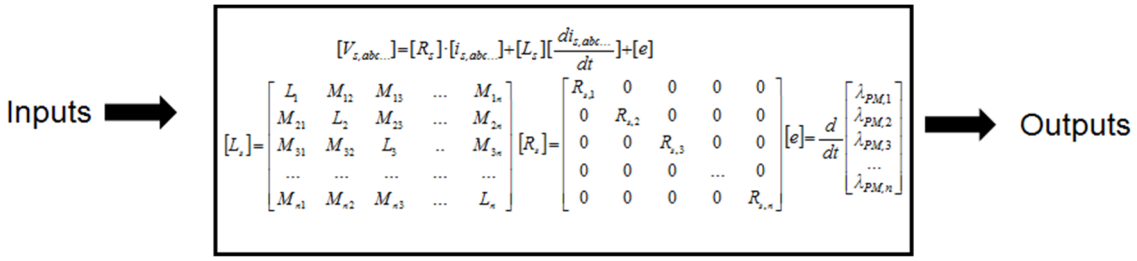


Fig. 3.2. White-box model with **all parameters are known**

[4] R. Petrone *et al.*, A review on model-based diagnosis methodologies for PEMFCs, International Journal of Hydrogen Energy 38 (2013) 7077-7091.

[5] Oliver Nelles, Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models, Springer, 2001.

3.3.2 Black-box models

- **Black-box modeling is useful tool when intending fitting experimental data (inputs and outputs) of the real system regardless of a particular mathematical structure of the model, that is, when such structure is unknown.**
- **Therefore, black-box models do not assume any particular structure of the analyzed system.**
- **System inputs and outputs relationships between are not based on physical equations as for analytical models, since they are deduced from experimental data.**

Different mathematical methods based on linear and nonlinear black-box model structures have been traditionally applied to represent the dynamics of electrical and electronic systems. These methods vary in complexity depending on flexibility and accuracy required to account for the dynamics and noise of the system under analysis.

When applying black-box models, physical significance or structural information of the system is lost, since the mapping between a black-box model and the set of ODEs defining the behavior of a nonlinear system are not bijective or equivalent.

This unclear system representation restricts the analysis of the system to the conditions represented by the data available.

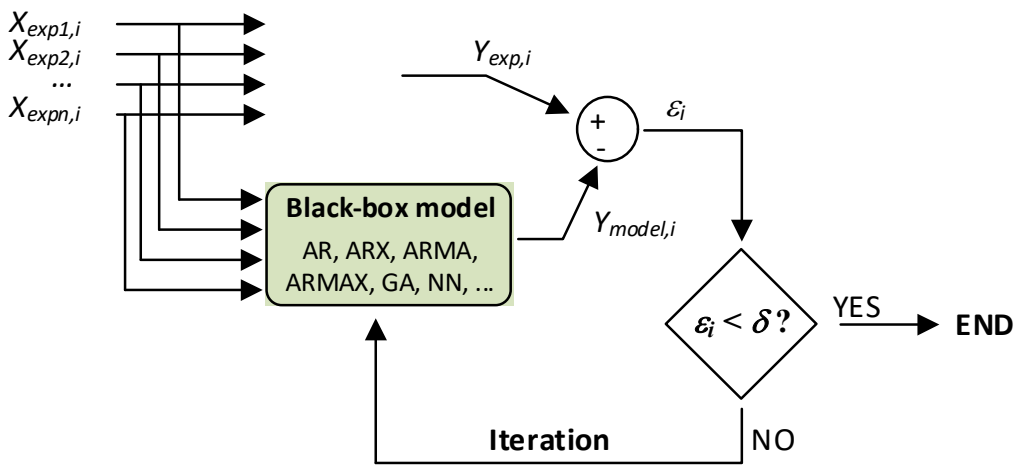


Fig. 3.3. Black-box model (**no model available and no model parameters fitting**)

Mathematical tools such as AR (auto-regressive), ARX (auto-regressive with exogenous inputs) models for **linear predictions** or ARMA(nonlinear auto-regressive moving average) and ANN (artificial neural network) models for **nonlinear predictions** can be used to model or approximate the behavior both linear and nonlinear systems.

In particular linear regression techniques are often more efficient than most non-linear techniques.

Experimental data sets are often split in two different sets:

- The **calibration set**, which is dedicated to calibrate or train the identification procedure.
- The remaining data constitute the **validation set**, which is dedicated to validate the identification procedure using data different that the used during the calibration stage[6].

3.3.3 Grey-box models

- Grey-box models combine a partial theoretical structure of the system with experimental data to complete the model.
- Grey-box models use prior physical knowledge although the model is not completely described by physical equations, but equations and the parameters are physically interpretable.
- Grey-box models are founded on physical laws supported by experimental data, thus replacing some mathematical equations with empirical formulas or map tables.
- Gray-box models are developed by using the white-box models whose parameters are estimated by using the measured system inputs and outputs.
- A grey-box model is a composition of a white-box (physically-based model) and a black-box model (data-based model), which exhibits the advantages of both approaches[7].
- Grey-box models allows easily accommodating missing data.
- Grey-box models allow reducing the computational burden when comparing with white-box models.

The general case to be represented by a grey-box model is a linear or non-linear model with a partially known theoretical structure and some unknown parts derived from experimental data.

Grey-box models search the optimal values of the undetermined parameters applying a linear or nonlinear optimization process. To evaluate the fitness between predicted and measured data, an objective function is defined during the optimization process. A simple choice is to **minimize the integrated root-mean-square error.**

Different optimization methods can be applied, such as the **trust region algorithm** (TRA), or evolutionary solvers including **genetic algorithms** (GA) or **particle swarm optimization**

[6]R. Petrone et al., A review on model-based diagnosis methodologies for PEMFCs, International Journal of Hydrogen Energy 38 (2013) 7077-7091.

[7]R. Kicsiny, Grey-box model for pipe temperature based on linear regression, International Journal of Heat and Mass Transfer 107 (2017) 13–20.

(PSO).According to Hu *et al.*[8] the three solvers provide similar results, although the TRA algorithm is the fastest and the GA the slowest.

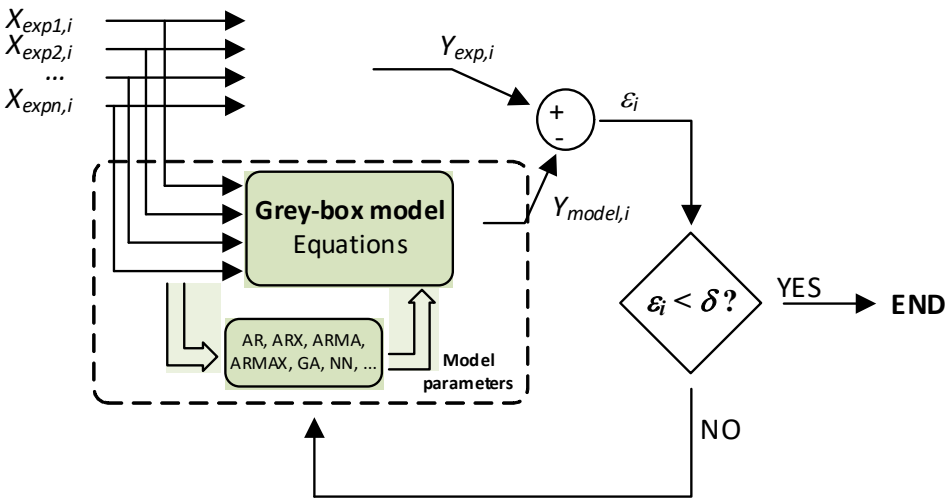


Fig. 3.4. Grey-box model (parameter fitting)

Next figure shows the parameter fitting process,

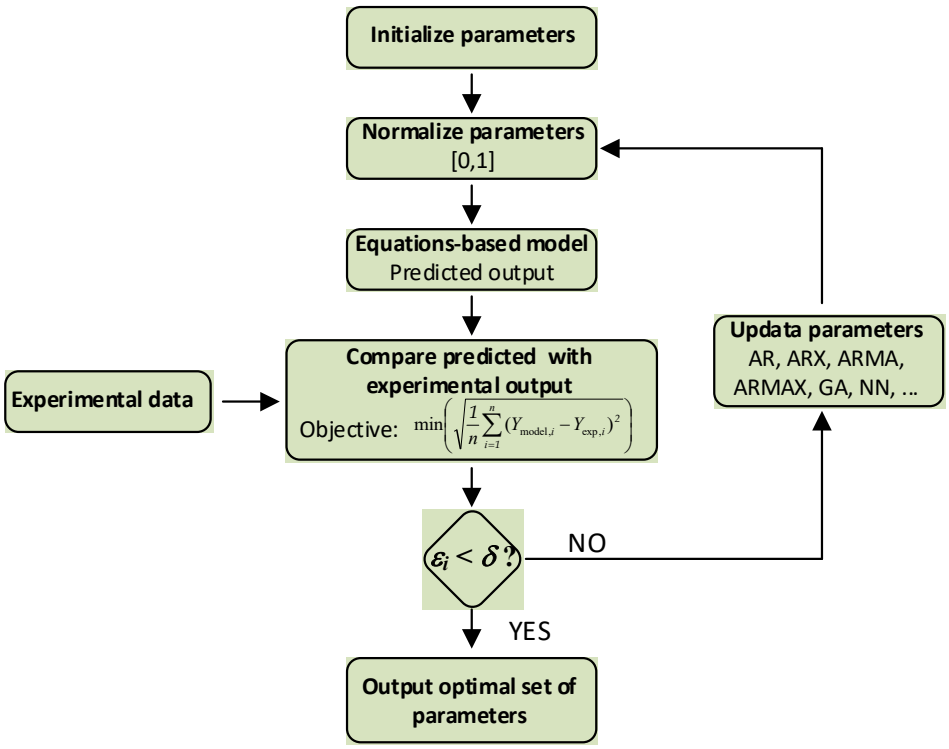


Fig. 3.5. Parameter fitting process in grey-box models.

[8] M. Hu, F. Xiao, L. Wang, Investigation of demand response potentials of residential air conditioners in smart grids using grey-box room thermal model, Applied Energy, 2017.

According to the literature, grey-box models can be classified as[9],

- Parameter identification based
- Observed-based
- Parity space methods

Grey-box models, identify the unknown parameters of the full nonlinear model through a part of experimental data. It is noted that experimental data is often split in two sets, that is, the **calibration and validation sets of data**. The data included in the **calibration set** is used to estimate the unknown parameters, whereas the remaining data (**validation data**) is used to validate the model. Parameter identification of the model through grey-box approach can be performed by using the ***idnlgrey*** function of MATLAB[10].

Lumped element grey-box models (of first, second and third order) are among the most interesting on the basis of a literature review conducted in [11].
A grey-box model consists of a set of continuous stochastic differential equations formulated in a state space form together with an output equation as follows:

$$d\boldsymbol{X}(t) = \boldsymbol{A}(\boldsymbol{\theta})\boldsymbol{X}(t)\,dt + \boldsymbol{B}(\boldsymbol{\theta})\boldsymbol{U}(t)\,dt + \boldsymbol{\sigma}(\boldsymbol{\theta})\,d\omega$$
$$\boldsymbol{Y}(t) = \boldsymbol{C}(\boldsymbol{\theta})\boldsymbol{X}(t) + \boldsymbol{D}(\boldsymbol{\theta})\boldsymbol{U}(t) + \boldsymbol{\varepsilon}$$

where X(t) is the state vector of the dynamic system, U(t) is a vector containing the measured inputs of the system, σ(t) is the process noise matrix which describes how the disturbances affect the system and ω is a Wiener process. The measured output of the system Y(y) is given as a function of the states X(t) and the inputs U(t). ε is the measurement error. θ are the parameters to be estimated.

[9]R. Petrone et al., A review on model-based diagnosis methodologies for PEMFCs, International Journal of Hydrogen Energy 38 (2013) 7077-7091.
[10]M. M. Barzegari, E. Alizadeh, A. H. Pahnabi, Grey-box modeling and model predictive control for cascade-type PEMFC, Energy 127 (2017), pp. 611-622.
[11] F. Ferracuti *et al.*, Data-driven models for short-term thermal behaviour prediction in real buildings, Applied Energy, 2017.

SaberRD models provided by Airbus

4 Saber RD models provided by Airbus

4.1 SaberRD models related to electrical systems

4.1.1 AAASP library/Three-stage generator

SABER RD Component:

Three-stage generator is not available in the first version of the AAASP library. Only wound rotor synchronous machine was found in SABER library.

Parameters to be included to interact with the synchronous machine

Parameter	Description	Default value
r_s	Resistance of the stator windings	
L_{ls}	Stator leakage inductance	
L_{md}	Direct-axis magnetizing inductance viewed from stator	
L_{mq}	Quadrature-axis magnetizing inductance viewed from stator	
r_{fd}'	Field resistance referred to the stator	
L_{lfd}'	Field leakage inductance referred to the stator	
r_{kd}'	Direct-axis damping winding resistance referred to the stator	
L_{lkd}'	Direct-axis damping winding leakage inductance referred to the stator	
r_{kq1}'	Resistance of the first quadrature-axis damping winding referred to the stator	
L_{lkq1}'	Field leakage inductance of the first quadrature-axis damping winding referred to the stator	
r_{kq2}'	Resistance of the second quadrature-axis damping winding referred to the stator	
L_{lkq2}'	Field leakage inductance of the second quadrature-axis damping winding referred to the stator	
J	Machine inertia	
B	Machine damping constant	
theta0	Initial angle	
p	Number of poles	

4.1.1.1 Equations of the three-phase salient-pole synchronous machine

The model uses a dq0 transformation in the rotor's reference frame.
Features:

1. Stator windings are temperature sensitive (static thermal).
2. Stator windings are assumed sinusoidally distributed.
3. Balanced 3-phase winding configuration assumed.
4. Torque output attenuation due to rotor inertia and friction.
5. Consideration of losses (windage).
6. Can be implemented as generator or motor.

Limitations:

1. Sinusoidal field distribution ignores spatial harmonics in the air gap.
2. Torque ripple and cogging torque effects neglected.
3. Saturation effects Ignored.
4. Iron losses neglected.

The a-phase stator reference axis agrees with the maximum mmf direction created by a positive a-phase current. d-axis rotor reference agrees with the permanent magnet flux direction (Fig. 1). The electrical angle between a-axis and q-axis in degrees is defined as θ. The electrical angular velocity is defined as $\omega = d\theta/dt$[12].

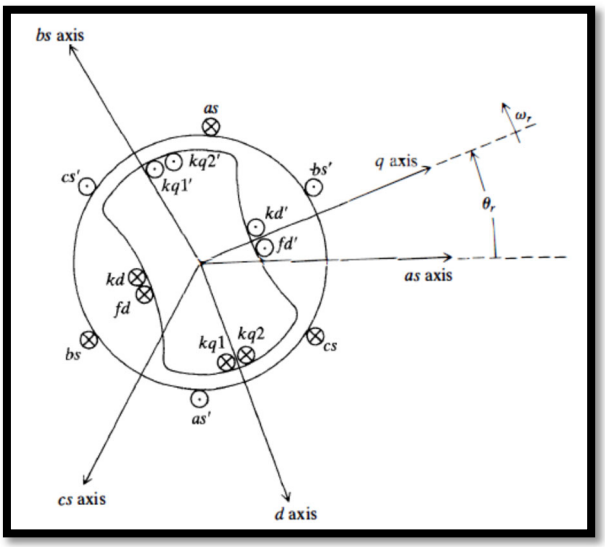


Fig. 4.1. abc- and dq-reference frames for the salient pole SM.

Park transformation:

[12] P. Krause, O. Wasynczuk, S. D. Sudhoff, S. Pekarek, Analysis of Electric Machinery and Drive Systems, 3rd Edition, Wiley-IEEE Press, 2013.

$$\boldsymbol{T}(\boldsymbol{\theta}) = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Inverse Park transformation:

$$\boldsymbol{T}^{-1}(\boldsymbol{\theta}) = \begin{bmatrix} \cos\theta & \sin\theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix}$$

Equations expressed according to the generator criterion for stator and motor criterion for rotor.

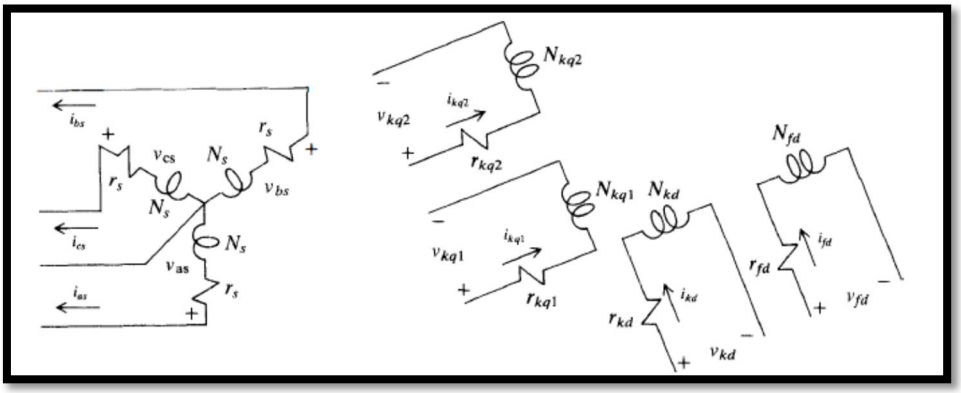


Fig. 4.2. Two-pole, 3-phase, wye-connected, salient-pole synchronous machine.

Electrical dynamic equation in terms of phase variables:

$$\boldsymbol{v_s^{abc}} = -\boldsymbol{r_s} \cdot \boldsymbol{i_s^{abc}} + \frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\psi_s^{abc}}$$

$$\boldsymbol{v_r^{qd}} = \boldsymbol{r_r} \cdot \boldsymbol{i_r^{qd}} + \frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\psi_r^{qd}}$$

$$\boldsymbol{v_s^{abc}} = \begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} \quad \boldsymbol{i_s^{abc}} = \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \end{bmatrix} \quad \boldsymbol{\psi_s^{abc}} = \begin{bmatrix} \psi_{as} \\ \psi_{bs} \\ \psi_{cs} \end{bmatrix} \quad \boldsymbol{r_s} = \begin{bmatrix} r_s & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix}$$

$$\boldsymbol{v_r^{qd}} = \begin{bmatrix} v_{kq1} \\ v_{kq2} \\ v_{fd} \\ v_{kd} \end{bmatrix} \quad \boldsymbol{i_r^{qd}} = \begin{bmatrix} i_{kq1} \\ i_{kq2} \\ i_{fd} \\ i_{kd} \end{bmatrix} \quad \boldsymbol{\psi_r^{qd}} = \begin{bmatrix} \psi_{kq1} \\ \psi_{kq2} \\ \psi_{fd} \\ \psi_{kd} \end{bmatrix} \quad \boldsymbol{r_r} = \begin{bmatrix} r_{kq1} & 0 & 0 & 0 \\ 0 & r_{kq2} & 0 & 0 \\ 0 & 0 & r_{fd} & 0 \\ 0 & 0 & 0 & r_{kd} \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{\psi_s^{abc}} \\ \boldsymbol{\psi_r^{qd}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L_s} & \boldsymbol{L_{sr}} \\ (\boldsymbol{L_{sr}})^T & \boldsymbol{L_r} \end{bmatrix} \begin{bmatrix} -\boldsymbol{i_s^{abc}} \\ \boldsymbol{i_r^{qd}} \end{bmatrix}$$

$$\boldsymbol{L_s} = \begin{bmatrix} L_A + L_{ls} - L_B \cdot \cos 2\theta & -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta - \frac{2\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta + \frac{2\pi}{3}\right) \\ -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta - \frac{2\pi}{3}\right) & L_A + L_{ls} - L_B \cdot \cos\left(2\theta - \frac{4\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos(2\theta + 2\pi) \\ -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta + \frac{2\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos(2\theta + 2\pi) & L_A + L_{ls} - L_B \cdot \cos\left(2\theta + \frac{4\pi}{3}\right) \end{bmatrix}$$

$$\boldsymbol{L_{sr}} = \begin{bmatrix} L_{skq1} \cdot \cos\theta & L_{skq2} \cdot \cos\theta & L_{sfd} \cdot \sin\theta & L_{skd} \cdot \sin\theta \\ L_{skq1} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) & L_{skq2} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) & L_{sfd} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) & L_{skd} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) \\ L_{skq1} \cdot \cos\left(\theta + \frac{2\pi}{3}\right) & L_{skq2} \cdot \cos\left(\theta + \frac{2\pi}{3}\right) & L_{sfd} \cdot \sin\left(\theta + \frac{2\pi}{3}\right) & L_{skd} \cdot \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix}$$

$$\boldsymbol{L_r} = \begin{bmatrix} L_{lkq1} + L_{mkq1} & L_{kq1kq2} & 0 & 0 \\ L_{kq1kq2} & L_{lkq2} + L_{mkq2} & 0 & 0 \\ 0 & 0 & L_{lfd} + L_{mfd} & L_{fdkd} \\ 0 & 0 & L_{fdkd} & L_{lkd} + L_{mkd} \end{bmatrix}$$

$$L_{mq} = \frac{3}{2}(L_A - L_B)$$

$$L_{md} = \frac{3}{2}(L_A + L_B)$$

$$L_{skq1} = \frac{2}{3}\left(\frac{N_{kq1}}{N_s}\right) L_{mq}$$

$$L_{skq2} = \frac{2}{3}\left(\frac{N_{kq2}}{N_s}\right) L_{mq}$$

$$L_{sfd} = \frac{2}{3}\left(\frac{N_{fd}}{N_s}\right) L_{md}$$

$$L_{skd} = \frac{2}{3}\left(\frac{N_{kd}}{N_s}\right) L_{md}$$

$$L_{mkq1} = \frac{2}{3}\left(\frac{N_{kq1}}{N_s}\right)^2 L_{mq}$$

$$L_{mkq2} = \frac{2}{3}\left(\frac{N_{kq2}}{N_s}\right)^2 L_{mq}$$

$$L_{mfd} = \frac{2}{3}\left(\frac{N_{fd}}{N_s}\right)^2 L_{md}$$

$$L_{mkd} = \frac{2}{3}\left(\frac{N_{kd}}{N_s}\right)^2 L_{md}$$

$$L_{kq1kq2} = \left(\frac{N_{kq2}}{N_{kq1}}\right) L_{mk1} = \left(\frac{N_{kq1}}{N_{kq2}}\right) L_{mk2}$$

$$L_{fdkd} = \left(\frac{N_{kd}}{N_{fd}}\right) L_{mfd} = \left(\frac{N_{fd}}{N_{kd}}\right) L_{mkd}$$

Rotor variables referred to the stator windings:

$$i_j{}' = \frac{2}{3}\left(\frac{N_j}{N_s}\right) i_j$$

$$v_j{}' = \left(\frac{N_s}{N_j}\right) v_j$$

$$\psi_j{}' = \left(\frac{N_s}{N_j}\right) \psi_j$$

Where "j" may be kq1, kq2, fd or kd.

$$\begin{bmatrix} \boldsymbol{\psi_s^{abc}} \\ \boldsymbol{\psi_r^{qd}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{L_s} & \boldsymbol{L'_{sr}} \\ \frac{2}{3}(\boldsymbol{L'_{sr}})^{\boldsymbol{T}} & \boldsymbol{L'_r} \end{bmatrix} \begin{bmatrix} -\boldsymbol{i_s^{abc}} \\ \boldsymbol{i'^{qd}_r} \end{bmatrix}$$

$$\boldsymbol{L'_{sr}} = \begin{bmatrix} L_{mq} \cdot \cos\theta & L_{mq} \cdot \cos\theta & L_{md} \cdot \sin\theta & L_{md} \cdot \sin\theta \\ L_{mq} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) & L_{mq} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) & L_{md} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) & L_{md} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) \\ L_{mq} \cdot \cos\left(\theta + \frac{2\pi}{3}\right) & L_{mq} \cdot \cos\left(\theta + \frac{2\pi}{3}\right) & L_{md} \cdot \sin\left(\theta + \frac{2\pi}{3}\right) & L_{md} \cdot \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix}$$

$$\boldsymbol{L'_r} = \begin{bmatrix} L'_{lkq1} + L_{mq} & L_{mq} & 0 & 0 \\ L_{mq} & L'_{lkq2} + L_{mq} & 0 & 0 \\ 0 & 0 & L'_{lfd} + L_{md} & L_{md} \\ 0 & 0 & L_{md} & L'_{lkd} + L_{md} \end{bmatrix}$$

$$\boldsymbol{v_s^{abc}} = \boldsymbol{r_s} \cdot \left(-\boldsymbol{i_s^{abc}}\right) + \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{L_s} \cdot \left(-\boldsymbol{i_s^{abc}}\right) + \boldsymbol{L'_{sr}} \cdot \boldsymbol{i'^{qd}_r}\right)$$

$$\begin{aligned} \left(\boldsymbol{T(\theta)} \cdot \boldsymbol{v_s^{abc}}\right) = {} & \left(\boldsymbol{T(\theta)} \cdot \boldsymbol{r_s} \cdot \boldsymbol{T^{-1}(\theta)}\right) \cdot \left(\boldsymbol{T(\theta)} \cdot \left(-\boldsymbol{i_s^{abc}}\right)\right) + \boldsymbol{T(\theta)} \\ & \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T^{-1}(\theta)} \cdot \left(\boldsymbol{T(\theta)} \cdot \boldsymbol{L_s} \cdot \boldsymbol{T^{-1}(\theta)}\right) \cdot \left(\boldsymbol{T(\theta)} \cdot \left(-\boldsymbol{i_s^{abc}}\right)\right) + \boldsymbol{T^{-1}(\theta)} \right. \\ & \left. \cdot \left(\boldsymbol{T(\theta)} \cdot \boldsymbol{L'_{sr}}\right) \cdot \left(\boldsymbol{i'^{qd}_r}\right)\right\} \end{aligned}$$

$$\left(\boldsymbol{v_s^{qd0}}\right) = \boldsymbol{r_s} \cdot \left(-\boldsymbol{i_s^{qd0}}\right) + \boldsymbol{T(\theta)} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T^{-1}(\theta)} \cdot \boldsymbol{L_s^r} \cdot \left(-\boldsymbol{i_s^{qd0}}\right) + \boldsymbol{T^{-1}(\theta)} \cdot \boldsymbol{L'^r_{sr}} \cdot \left(\boldsymbol{i'^{qd}_r}\right)\right\}$$

$$\begin{aligned} \left(\boldsymbol{v_s^{qd0}}\right) = {} & \boldsymbol{r_s} \cdot \left(-\boldsymbol{i_s^{qd0}}\right) + \boldsymbol{T(\theta)} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T^{-1}(\theta)}\right\} \cdot \left\{\boldsymbol{L_s^r} \cdot \left(-\boldsymbol{i_s^{qd0}}\right) + \boldsymbol{L'^r_{sr}} \cdot \left(\boldsymbol{i'^{qd}_r}\right)\right\} + \boldsymbol{T(\theta)} \cdot \boldsymbol{T^{-1}(\theta)} \\ & \cdot \left\{\boldsymbol{L_s^r} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(-\boldsymbol{i_s^{qd0}}\right) + \boldsymbol{L'^r_{sr}} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{i'^{qd}_r}\right)\right\} \end{aligned}$$

$$\left(\boldsymbol{v}_s^{qd0}\right) = \boldsymbol{r}_s \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \frac{\mathrm{d}}{\mathrm{dt}}\{\boldsymbol{T}^{-1}(\boldsymbol{\theta})\} \cdot \left\{\boldsymbol{L}_s^r \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{L}'^r_{sr} \cdot \left(\boldsymbol{i}'^{qd}_r\right)\right\} + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{T}^{-1}(\boldsymbol{\theta}) \cdot \left\{\boldsymbol{L}_s^r \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{L}'^r_{sr} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{i}'^{qd}_r\right)\right\}$$

$$\boldsymbol{v}_s^{qd0} = \boldsymbol{r}_s \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{\omega} \cdot \boldsymbol{L}_s^r \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{\omega} \cdot \boldsymbol{L}'^r_{sr} \cdot \left(\boldsymbol{i}'^{qd}_r\right) + \boldsymbol{L}_s^r \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{L}'^r_{sr} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{i}'^{qd}_r\right)$$

$$\boldsymbol{v}_s^{qd0} = \begin{bmatrix} v_{qs} \\ v_{ds} \\ v_{0s} \end{bmatrix} \quad \boldsymbol{i}_s^{qd0} = \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \end{bmatrix} \quad \boldsymbol{i}_r^{\prime qd} = \begin{bmatrix} i'_{kq1} \\ i'_{kq2} \\ i'_{fd} \\ i'_{kd} \end{bmatrix} \quad \boldsymbol{\omega} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{L}_s^r = \begin{bmatrix} L_{mq} + L_{ls} & 0 & 0 \\ 0 & L_{md} + L_{ls} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix}$$

$$\boldsymbol{L}'^r_{sr} = \begin{bmatrix} L_{mq} & L_{mq} & 0 & 0 \\ 0 & 0 & L_{md} & L_{md} \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{v}_r^{\prime qd} = \boldsymbol{r}'_r \cdot \boldsymbol{i}_r^{\prime qd} + \frac{\mathrm{d}}{\mathrm{dt}}\left(\frac{2}{3}\left(\boldsymbol{L}'_{sr}\right)^T \cdot \left(-\boldsymbol{i}_s^{abc}\right) + \boldsymbol{L}'_r \cdot \boldsymbol{i}_r^{\prime qd}\right)$$

$$\boldsymbol{v}_r^{\prime qd} = \boldsymbol{r}'_r \cdot \boldsymbol{i}_r^{\prime qd} + \frac{\mathrm{d}}{\mathrm{dt}}\left(\left(\frac{2}{3}\left(\boldsymbol{L}'_{sr}\right)^T \cdot \boldsymbol{T}^{-1}(\boldsymbol{\theta})\right) \cdot \left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \left(-\boldsymbol{i}_s^{abc}\right)\right) + \boldsymbol{L}'_r \cdot \boldsymbol{i}_r^{\prime qd}\right)$$

$$\boldsymbol{v}_r^{\prime qd} = \boldsymbol{r}'_r \cdot \boldsymbol{i}_r^{\prime qd} + \left(\boldsymbol{L}'^r_{sr}\right)^T \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{T}^{-1}(\boldsymbol{\theta})\right) \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{L}'_r \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{i}_r^{\prime qd}\right)$$

$$\boldsymbol{v}_r^{\prime qd} = \boldsymbol{r}'_r \cdot \boldsymbol{i}_r^{\prime qd} + \left(\boldsymbol{L}'^r_{sr}\right)^T \cdot \boldsymbol{\omega} \cdot \left(-\boldsymbol{i}_s^{qd0}\right) + \boldsymbol{L}'_r \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left(\boldsymbol{i}_r^{\prime qd}\right)$$

$$\boldsymbol{v}_r^{\prime qd} = \begin{bmatrix} v'_{kq1} \\ v'_{kq2} \\ v'_{fd} \\ v'_{kd} \end{bmatrix} \quad \boldsymbol{i}_s^{qd0} = \begin{bmatrix} i_{qs} \\ i_{ds} \\ i_{0s} \end{bmatrix} \quad \boldsymbol{i}_r^{\prime qd} = \begin{bmatrix} i'_{kq1} \\ i'_{kq2} \\ i'_{fd} \\ i'_{kd} \end{bmatrix} \quad \boldsymbol{\omega} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Then:

$$v_{qs} = \frac{2}{3}\left(v_{as} \cdot \cos\theta + v_{bs} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) + v_{cs} \cdot \cos\left(\theta + \frac{2\pi}{3}\right)\right)$$

$$v_{ds} = \frac{2}{3}\left(v_{as} \cdot \sin\theta + v_{bs} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) + v_{cs} \cdot \sin\left(\theta + \frac{2\pi}{3}\right)\right)$$

$$v_{0s} = \frac{1}{3}\left(v_{as} + v_{bs} + v_{cs}\right)$$

$$v_{qs} = -r_s \cdot i_{qs} - \omega \cdot \left(L_{md} + L_{ls}\right) \cdot i_{ds} + \omega \cdot L_{md} \cdot \left(i'_{fd} + i'_{kd}\right) - \left(L_{mq} + L_{ls}\right) \cdot \frac{di_{qs}}{dt} + L_{mq} \cdot \frac{d\left(i'_{kq1} + i'_{kq2}\right)}{dt}$$

$$v_{ds} = -r_s \cdot i_{ds} + \omega \cdot (L_{mq} + L_{ls}) \cdot i_{qs} - \omega \cdot L_{mq} \cdot (i'_{kq1} + i'_{kq2}) - (L_{md} + L_{ls}) \cdot \frac{di_{ds}}{dt} + L_{md} \cdot \frac{d(i'_{fd} + i'_{kd})}{dt}$$

$$v_{0s} = -r_s \cdot i_{0s} - L_{ls} \cdot \frac{di_{0s}}{dt}$$

$$v'_{kq1} = r'_{kq1} \cdot i'_{kq1} + (L_{mq} + L'_{lkq1}) \cdot \frac{di'_{kq1}}{dt} + L_{mq} \cdot \frac{d(i'_{kq2} - i_{qs})}{dt}$$

$$v'_{kq2} = r'_{kq2} \cdot i'_{kq2} + (L_{mq} + L'_{lkq2}) \cdot \frac{di'_{kq2}}{dt} + L_{mq} \cdot \frac{d(i'_{kq1} - i_{qs})}{dt}$$

$$v'_{fd} = r'_{fd} \cdot i'_{fd} + (L_{md} + L'_{lfd}) \cdot \frac{di'_{fd}}{dt} + L_{md} \cdot \frac{d(i'_{kd} - i_{ds})}{dt}$$

$$v'_{kd} = r'_{kd} \cdot i'_{kd} + (L_{md} + L'_{lkd}) \cdot \frac{di'_{kd}}{dt} + L_{md} \cdot \frac{d(i'_{fd} - i_{ds})}{dt}$$

$$T_e = \frac{3}{2} \cdot \frac{p}{2} \cdot \left[L_{md}(-i_{ds} + i'_{fd} + i'_{kd}) \cdot i_{qs} - L_{mq}(-i_{qs} + i'_{kq1} + i'_{kq2}) \cdot i_{ds}\right]$$

$$T_e = J \cdot \frac{2}{p} \cdot \frac{d\omega}{dt} + B \cdot \frac{2}{p} \cdot \omega + T_L$$

$$\omega = \frac{d\theta}{dt}$$

$$i_{as} = i_{qs} \cdot \cos\theta + i_{ds} \cdot \sin\theta + i_{0s}$$

$$i_{bs} = i_{qs} \cdot \cos\left(\theta - \frac{2\pi}{3}\right) + i_{ds} \cdot \sin\left(\theta - \frac{2\pi}{3}\right) + i_{0s}$$

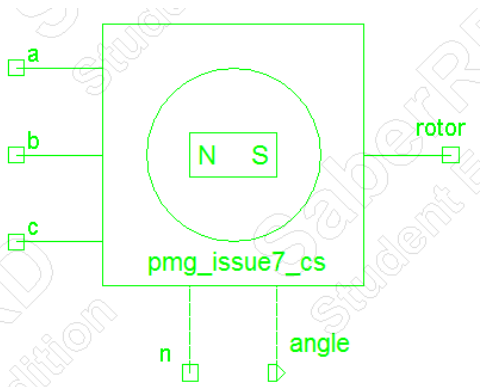
$$i_{cs} = i_{qs} \cdot \cos\left(\theta + \frac{2\pi}{3}\right) + i_{ds} \cdot \sin\left(\theta + \frac{2\pi}{3}\right) + i_{0s}$$

$$p_{in} = i_{as} \cdot v_{as} + i_{bs} \cdot v_{bs} + i_{cs} \cdot v_{cs}$$

4.1.2 AAASP library / Three phase inverter + PMSM machine

SABER RD Component: pmg_issue7_cs

Behavioral model of a permanent-magnet (PM) three-phase synchronous machine with angular position available which is already included in the AAASP library.



Connections / Inputs / Outputs

Electrical: a, b, c, n
Mechanical: rotor, angle

Parameters

Parameter	Description	Default value
rs	Resistance of the windings	*req*
lq	Quadrature axis winding self-inductance	
ld	Direct axis winding self-inductance	
l0	0-axis self-inductance	0
phi_max	Peak flux linkage of permanent magnet	*req*
d	Motor damping constant	0
j	Motor inertia	
theta0	Initial angle	undef
sft	Static friction torque	0
dft	Dynamic friction torque	0
alpha	Temperature coefficient of Cu	0.0039
t0	Initial specific ambient temperature at value of alpha	20.0
p	Number of poles	2

Dependencies

fricls_w.C0

Model

```
{
    external number temp
    number xrs
    number xld
    number xlq
    number xt0
    number xalpha
    number xl0
    number xp
    number xd
    number xj
    number xphi_max
    #
    number remainder = 0
    number p2
    val nu rpm
    val nu w_elec
    val nu w_elec_phid
    val nu vq
    val nu vd
    val nu v0
    val nu ld_dyn
    val nu lq_dyn
```

```
    val nu phi_mag_dyn
    val nu ias
    val nu ibs
    val nu ics
    val nu w_elec_phiq
    var nu id
    var nu phid
    var nu iq
    var nu phiq
    var nu i0
    var nu phi0
    number ang
    val nu ang_elec_mks
    val nu p_as
    val nu p_bs
    val nu p_cs
    val nu p_abcs
    val nu p_in_total
    val nu p_loss_e
    val nu p_loss_m
    val nu p_loss_total
    val nu tq_mks_reaction
    val nu tq_mks_reluctance
    val nu tq_elmag_mks
    #
    var nu thetarm
    number Math_Pi

    equations {
        i(a->n) += (((iq) * cos(ang_elec_mks)) + ((id) *
sin(ang_elec_mks))) + (i0)
    }

    equations {
        i(b->n) += (((iq) * cos((ang_elec_mks) - (ang))) + ((id) *
sin((ang_elec_mks) - (ang)))) + (i0)
    }

    equations {
      # next equation was modified
        i(c->n) += (((iq) * cos((ang_elec_mks) + (ang))) + ((id) *
sin((ang_elec_mks) + (ang)))) + (i0)
    }

    equations {
        # tq_elmag_mks - d_by_dt(j_eff*w_rps) - d_eff*w_rps
        tq_Nm(rotor) += ((tq_elmag_mks) - (d_by_dt(((j) *
(w_radps(rotor)))))) - ((d) * (w_radps(rotor)))
    }
      # next component is redundant if motor damping constant is used
    #Friction on the shaft
    fricls_w.C0 wrm:rotor = dft=dft, sft=sft

    parameters {
```

```
    # Check the resistance values
    if ((rs) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "resistance value")
        xrs = 0
    }
    else
        if ((rs) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "resistance value")
            xrs = 0
        }
    else
        if ((rs) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "resistance", "rs")
            xrs = 0
        }
    else {
        xrs = rs
    }
}

parameters {
    # Check the Ld values
    if ((ld) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "Inductance value")
        xld = 0
    }
    else
        if ((ld) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Inductance value")
            xld = 0
        }
    else
        if ((ld) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Inductance", "ld")
            xld = 0
        }
    else {
        xld = ld
    }
}

parameters {
    # Check the lq values
    if ((lq) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "Inductance value")
        xlq = 0
    }
    else
        if ((lq) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Inductance value")
            xlq = 0
```

```
        }
    else
        if ((lq) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Inductance", "lq")
            xlq = 0
        }
    else {
        xlq = lq
    }
}

parameters {
    # Check the l0 values
    if ((l0) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "Inductance value")
        xl0 = 0
    }
    else
        if ((l0) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Inductance value")
            xl0 = 0
        }
    else
        if ((l0) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Inductance", "l0")
            xl0 = 0
        }
    else {
        xl0 = l0
    }
}

parameters {
    # Check the p values
    if ((p) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "poles value")
        xp = 0
    }
    else
        if ((p) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Poles value")
            xp = 0
        }
    else
        if ((p) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Poles", "p")
            xp = 0
        }
    else {
        xp = p
    }
    # Calculate the number of pole pairs
```

```
    p2 = (xp) / (2)
}

parameters {
    # Check the t0 values
    if ((t0) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "temperature value")
        xt0 = 0
    }
    else
        if ((t0) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "temperature value")
            xt0 = 0
        }
    else
        if ((t0) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "temperature", "t0")
            xt0 = 0
        }
    else {
        xt0 = t0
    }
}

parameters {
    # Check the phi_max values
    if ((phi_max) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "Flux value")
        xphi_max = 0
    }
    else
        if ((phi_max) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Flux value")
            xphi_max = 0
        }
    else
        if ((phi_max) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Flux", "phi_max")
            xphi_max = 0
        }
    else {
        xphi_max = phi_max
    }
}

parameters {
    # Check the d values
    if ((d) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "damping value")
        xd = 0
    }
    else
```

```
        if ((d) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "damping value")
            xd = 0
        }
    else
        if ((d) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "damping", "d")
            xd = 0
        }
    else {
        xd = d
    }
}

parameters {
    # Check the j values
    if ((j) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "inertia value")
        xj = 0
    }
    else
        if ((j) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "inertia value")
            xj = 0
        }
    else
        if ((j) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "inertia", "j")
            xj = 0
        }
    else {
        xj = j
    }
}

parameters {
    # Check the alpha values
    if ((alpha) == (inf)) then {
        saber_message("TMPL_S_RANGE_NE_INF", "Alpha value")
        xalpha = 0
    }
    else
        if ((alpha) < (0)) then {
            # Negative Resistance
            saber_message("TMPL_W_GE_REL_VALUE", "Alpha value")
            xalpha = 0
        }
    else
        if ((alpha) == (undef)) then {
            saber_message("TMPL_S_ALT_SPEC", "Alpha", "alpha")
            xalpha = 0
        }
```

```
        else {
            xalpha = alpha
        }
    }

    parameters {
        # modify the calculated winding resistance based on the
themperature
        # coefficient of the material (alpha) and the current simulation
temperature
        xrs = (xrs) * ((1) + ((xalpha) * ((temp) - (xt0))))
    }

    values {
        # Calculate the electrical angular velocity
        w_elec = ((xp) * (w_radps(rotor))) / (2)
        # Calcuate the electrical flux
      # next equations were modified
        w_elec_phid = (w_elec) * (phid)
        w_elec_phiq = (w_elec) * (phiq)
        # calculate rotor effective electrical angle:
        ang_elec_mks = (thetarm) * (p2)
        # Calculate driect and quadrature voltages
        #
#vq=(2/3)*(vas*cos(ang_elec_mks) + vbs*cos(ang_elec_mks-ang) +
vcs*cos(ang_elec_mks+ang))
#
        vq = ((2) / (3)) * ((((v(a,n)) * cos(ang_elec_mks)) + ((v(b,n)) *
cos((ang_elec_mks) - (ang)))) + ((v(c,n)) * cos((ang_elec_mks) + (ang))))
        vd = ((2) / (3)) * ((((v(a,n)) * sin(ang_elec_mks)) + ((v(b,n)) *
sin((ang_elec_mks) - (ang)))) + ((v(c,n)) * sin((ang_elec_mks) + (ang))))
        v0 = ((1) / (3)) * (((v(a,n)) + (v(b,n))) + (v(c,n)))
        # calculate terminal currents
        ias = (((iq) * cos(ang_elec_mks)) + ((id) * sin(ang_elec_mks))) +
(i0)
        ibs = (((iq) * cos((ang_elec_mks) - (ang))) + ((id) *
sin((ang_elec_mks) - (ang)))) + (i0)
      # next equation was modified
        ics = (((iq) * cos((ang_elec_mks) + (ang))) + ((id) *
sin((ang_elec_mks) + (ang)))) + (i0)
        # calculate dynamic inductances
      # next equations were modified
        ld_dyn = xld
        lq_dyn = xlq
        # calcuate dynamic phi max
        phi_mag_dyn = xphi_max
        # Calculate the instantaneous power
        p_as = (ias) * (v(a,n))
        p_bs = (ibs) * (v(b,n))
        p_cs = (ics) * (v(c,n))
        p_abcs = ((p_as) + (p_bs)) + (p_cs)
        p_in_total = p_abcs
        # Calculate the mechanical losses
        # loss in stator windings
```

```
        p_loss_e = (xrs) * ((((ias) * (ias)) + ((ibs) * (ibs))) + ((ics) *
(ics)))
        #mechnical loss
      # next equation was modified
        p_loss_m = ((d) * abs(w_radps(rotor)))) * abs(w_radps(rotor))
        #total loss power
        p_loss_total = (p_loss_e) + (p_loss_m)
        # Electromagnetic Torque
        tq_mks_reaction = (((p2) * ((3) / (2))) * (phi_mag_dyn)) * (iq)
        tq_mks_reluctance = ((((p2) * ((3) / (2))) * ((ld_dyn) - (lq_dyn)))
* (id)) * (iq)
        tq_elmag_mks = (tq_mks_reluctance) + (tq_mks_reaction)
        # Calculate the rotor speed in RPM
        rpm = (((2) * (Math_Pi)) * (w_radps(rotor))) / (60)
    }

    equations {
      # next equations were modified
        phid : phid = ((id) * (ld_dyn)) + (phi_mag_dyn)
        phiq : phiq = ((iq) * (lq_dyn))
        phi0 : phi0 = (i0) * (xl0)
    }

    equations {
        iq : vq = (((iq) * (xrs)) + (d_by_dt(phiq))) + (w_elec_phid)
        id : vd = (((id) * (xrs)) + (d_by_dt(phid))) - (w_elec_phiq)
        i0 : v0 = ((i0) * (xrs)) + (d_by_dt(phi0))
    }

    parameters {
        # Set up Pi
        Math_Pi = 3.14159
        # set up the angle displacement
        ang = ((2) * (Math_Pi)) / (3)
    }

    equations {
        #
#mirr:mirr * mast_dc_domain+mirr'ddt()==dmirrdt * mast_time_domain
#
#      thetarm: d_by_dt(thetarm)=(1-dc_domain)*w_rps + \
#                case*dc_domain*eps*(thetarm-xtheta0) + \
#                (1-case)*dc_domain*(thetarm-xtheta0)
#
#
#
        thetarm : d_by_dt(thetarm) = (((1) - (dc_domain)) *
(w_radps(rotor))) + ((dc_domain) * ((thetarm) - (theta0)))
        angle : angle = thetarm
    }
}
```

4.1.3.1 Equations of the permanent-magnet three-phase synchronous machine

The model uses a dq0 transformation in the rotor's reference frame.
Features:

1. Stator windings are temperature sensitive (static thermal).
2. Stator windings are assumed sinusoidally distributed.
3. Balanced 3-phase winding configuration assumed.
4. Torque output attenuation due to rotor inertia and friction.
5. Consideration of losses (windage).
6. Can be implemented as generator or motor.

Limitations:

1. Sinusoidal field distribution ignores spatial harmonics in the air gap.
2. Torque ripple and cogging torque effects neglected.
3. Saturation effects Ignored.
4. Iron losses neglected.

The a-phase stator reference axis agrees with the maximum mmf direction created by a positive a-phase current. d-axis rotor reference agrees with the permanent magnet flux direction (Fig. 1). The electrical angle between a-axis and q-axis in degrees is defined as θ. The electrical angular velocity is defined as ω = dθ/dt [13].

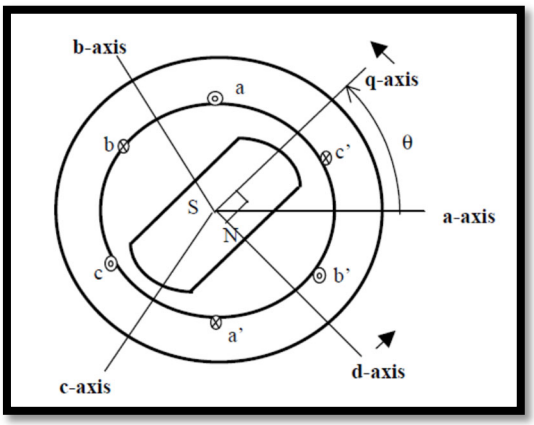


Fig. 4.3. abc- and dq-reference frames for the PMSM.

Park transformation:

$$\boldsymbol{T}(\boldsymbol{\theta}) = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ \sin(\theta) & \sin\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) \\ \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Inverse Park transformation:

[13] P. Krause, O. Wasynczuk, S. D. Sudhoff, S. Pekarek, Analysis of Electric Machinery and Drive Systems, 3rd Edition, Wiley-IEEE Press, 2013.

$$\boldsymbol{T}^{-1}(\boldsymbol{\theta}) = \begin{bmatrix} \cos\theta & \sin\theta & 1 \\ \cos\left(\theta - \frac{2\pi}{3}\right) & \sin\left(\theta - \frac{2\pi}{3}\right) & 1 \\ \cos\left(\theta + \frac{2\pi}{3}\right) & \sin\left(\theta + \frac{2\pi}{3}\right) & 1 \end{bmatrix}$$

Equations expressed according to the motor criterion.
Electrical dynamic equation in terms of phase variables (Krause):

$$\boldsymbol{v}_s^{abc} = \boldsymbol{r}_s \cdot \boldsymbol{i}_s^{abc} + \frac{\mathrm{d}}{\mathrm{dt}}\boldsymbol{\psi}_s^{abc}$$

$$\boldsymbol{v}_s^{abc} = \begin{bmatrix} v_a \\ v_b \\ v_c \end{bmatrix} \quad \boldsymbol{i}_s^{abc} = \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} \quad \boldsymbol{\psi}_s^{abc} = \begin{bmatrix} \psi_a \\ \psi_b \\ \psi_c \end{bmatrix} \quad \boldsymbol{r}_s = \begin{bmatrix} r_s & 0 & 0 \\ 0 & r_s & 0 \\ 0 & 0 & r_s \end{bmatrix}$$

$$\boldsymbol{\psi}_s^{abc} = \boldsymbol{L}_s^{abc} \cdot \boldsymbol{i}_s^{abc} + \boldsymbol{\psi}_M^{abc}$$

$$\boldsymbol{L}_s^{abc} = \begin{bmatrix} L_A + L_l - L_B \cdot \cos 2\theta & -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta - \frac{2\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta + \frac{2\pi}{3}\right) \\ -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta - \frac{2\pi}{3}\right) & L_A + L_l - L_B \cdot \cos\left(2\theta - \frac{4\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos(2\theta + 2\pi) \\ -\frac{1}{2}L_A - L_B \cdot \cos\left(2\theta + \frac{2\pi}{3}\right) & -\frac{1}{2}L_A - L_B \cdot \cos(2\theta + 2\pi) & L_A + L_l - L_B \cdot \cos\left(2\theta + \frac{4\pi}{3}\right) \end{bmatrix}$$

$$\boldsymbol{\psi}_M^{abc} = \begin{bmatrix} \psi_m \cdot \sin\theta \\ \psi_m \cdot \sin\left(\theta - \frac{2\pi}{3}\right) \\ \psi_m \cdot \sin\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix}$$

$$\left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{v}_s^{abc}\right) = \left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{r}_s \cdot \boldsymbol{T}^{-1}(\boldsymbol{\theta})\right) \cdot \left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{i}_s^{abc}\right) + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T}^{-1}(\boldsymbol{\theta}) \cdot \left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{L}_s^{abc} \cdot \boldsymbol{T}^{-1}(\boldsymbol{\theta})\right) \cdot \left(\boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{i}_s^{abc}\right) + \boldsymbol{\psi}_M^{abc}\right\}$$

$$\boldsymbol{v}_s^{qd0} = \boldsymbol{r}_s \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T}^{-1}(\boldsymbol{\theta}) \cdot \boldsymbol{L}_s^{qd0} \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{\psi}_M^{abc}\right\}$$

$$\boldsymbol{v}_s^{qd0} = \boldsymbol{r}_s \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{T}^{-1}(\boldsymbol{\theta})\right\} \cdot \boldsymbol{L}_s^{qd0} \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \boldsymbol{T}^{-1}(\boldsymbol{\theta}) \cdot \boldsymbol{L}_s^{qd0} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{i}_s^{qd0}\right\} + \boldsymbol{T}(\boldsymbol{\theta}) \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{\psi}_M^{abc}\right\}$$

$$\frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{\psi}_M^{abc}\right\} = \begin{bmatrix} \omega \cdot \psi_m \cdot \cos\theta \\ \omega \cdot \psi_m \cdot \cos\left(\theta - \frac{2\pi}{3}\right) \\ \omega \cdot \psi_m \cdot \cos\left(\theta + \frac{2\pi}{3}\right) \end{bmatrix}$$

$$\boldsymbol{v}_s^{qd0} = \boldsymbol{r}_s \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{\omega} \cdot \boldsymbol{L}_s^{qd0} \cdot \boldsymbol{i}_s^{qd0} + \boldsymbol{L}_s^{qd0} \cdot \frac{\mathrm{d}}{\mathrm{dt}}\left\{\boldsymbol{i}_s^{qd0}\right\} + \boldsymbol{\omega}\boldsymbol{\psi}_M^{qd0}$$

$$\boldsymbol{v}_s^{qd0} = \begin{bmatrix} v_q \\ v_d \\ v_0 \end{bmatrix} \quad \boldsymbol{i}_s^{qd0} = \begin{bmatrix} i_q \\ i_d \\ i_0 \end{bmatrix} \quad \boldsymbol{\omega} = \begin{bmatrix} 0 & \omega & 0 \\ -\omega & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\boldsymbol{L_s^{qd0}} = \begin{bmatrix} \frac{3}{2}(L_A - L_B) + L_l & 0 & 0 \\ 0 & \frac{3}{2}(L_A + L_B) + L_l & 0 \\ 0 & 0 & L_l \end{bmatrix} = \begin{bmatrix} L_{mq} + L_l & 0 & 0 \\ 0 & L_{md} + L_l & 0 \\ 0 & 0 & L_l \end{bmatrix} = \begin{bmatrix} L_q & 0 & 0 \\ 0 & L_d & 0 \\ 0 & 0 & L_0 \end{bmatrix}$$

$$\boldsymbol{\omega\psi_M^{qd0}} = \begin{bmatrix} \omega \cdot \psi_m \\ 0 \\ 0 \end{bmatrix}$$

Then:

$$v_q = \frac{2}{3}\left(v_a \cdot \cos\theta + v_b \cdot \cos\left(\theta - \frac{2\pi}{3}\right) + v_c \cdot \cos\left(\theta + \frac{2\pi}{3}\right)\right)$$

$$v_d = \frac{2}{3}\left(v_a \cdot \sin\theta + v_b \cdot \sin\left(\theta - \frac{2\pi}{3}\right) + v_c \cdot \sin\left(\theta + \frac{2\pi}{3}\right)\right)$$

$$v_0 = \frac{1}{3}(v_a + v_b + v_c)$$

$$v_q = r_s \cdot i_q + \omega \cdot L_d \cdot i_d + L_q \cdot \frac{di_q}{dt} + \omega \cdot \psi_m$$

$$v_d = r_s \cdot i_d - \omega \cdot L_q \cdot i_q + L_d \cdot \frac{di_d}{dt}$$

$$v_0 = r_s \cdot i_0 + L_0 \cdot \frac{di_0}{dt}$$

$$T_e = \frac{3}{2} \cdot \frac{p}{2} \cdot \left[\psi_m \cdot i_q + \left(L_d - L_q\right) \cdot i_q \cdot i_d\right]$$

$$T_e = J \cdot \frac{2}{p} \cdot \frac{d\omega}{dt} + B \cdot \frac{2}{p} \cdot \omega + T_L$$

$$\omega = \frac{d\theta}{dt}$$

$$i_a = i_q \cdot \cos\theta + i_d \cdot \sin\theta + i_0$$

$$i_b = i_q \cdot \cos\left(\theta - \frac{2\pi}{3}\right) + i_d \cdot \sin\left(\theta - \frac{2\pi}{3}\right) + i_0$$

$$i_c = i_q \cdot \cos\left(\theta + \frac{2\pi}{3}\right) + i_d \cdot \sin\left(\theta + \frac{2\pi}{3}\right) + i_0$$

$$p_{in} = i_a \cdot v_a + i_b \cdot v_b + i_c \cdot v_c$$

$$p_{loss\_e} = r_s \cdot \left(i_a^2 + i_b^2 + i_c^2\right)$$

$$tq\_mks\_reluctance = \frac{3}{2} \cdot \frac{p}{2} \cdot \left(L_d - L_q\right) \cdot i_q \cdot i_d$$

$$tq\_elmag\_mks = tq\_mks\_reluctance + tq\_mks\_reaction$$

4.1.3 AAASP library / Starter-generator

SABER RD Component:

This component has not been worked yet.

4.2 SaberRD models related to electronic systems

4.2.1 Buck converter

There are two models available for the buck converter the open loop and the closed loop in saber. The parameters in which the value is `*req*` the value must be specified to do the simulation and the values of other parameters are optional

4.2.2 Buck converter in open loop

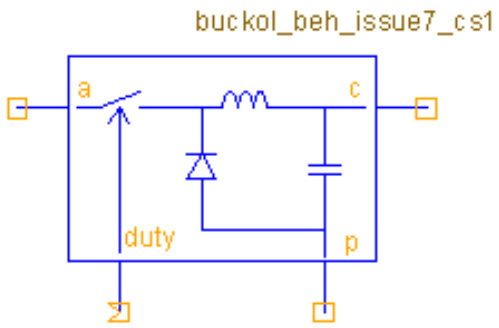


Figure 4.4. Buck converter in open loop

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	
cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*

4.2.3 Buck converter in closed loop

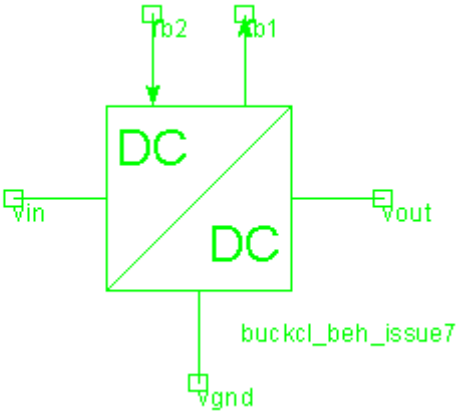


Figure 4.5 Closed loop buck converter

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	
cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*
vref	The reference voltaje for the feedback	*req*
ss	Soft Start Time	
k	The loop gain	*req*
fs	The Switching Frequency	*req*
dutymax	Maximum duty cycle	
dutymin	Minimum Duty cycle	
dt	Delay Time	

4.3 Boost converter

There are two models available for the boost converter the open loop and the closed loop in saber. The parameters in which the value is *req* the value must be specifed to do the simulation and the values of other parameters are optional

4.3.1 Boost converter in open loop

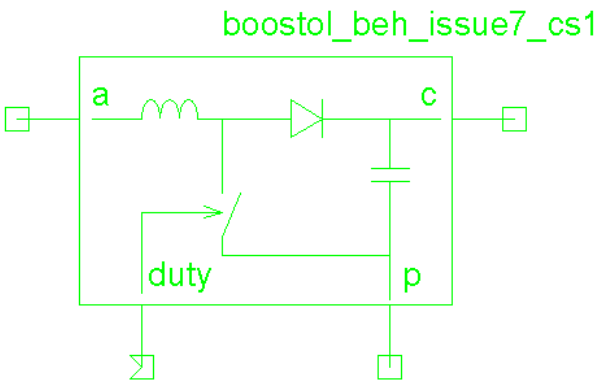


Figure 4.6. Boost converter in open loop

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	
cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	0
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*

4.3.2 Boost converter in closed loop

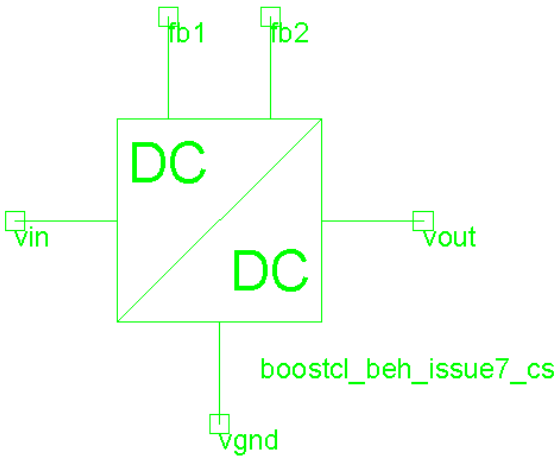


Figure 4.7. Boost converter closed loop model

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	

cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*
vref	The reference voltaje for the feedback	
ss	Soft Start Time	
k	The loop gain	
fs	The Switching Frequency	
dutymax	Maximum duty cycle	
dutymin	Minimum Duty cycle	
dt	Delay Time	

4.4 Buck-boost converter

There are two models available for the buck-boost converter the open loop and the closed loop in saber. The parameters in which the value is *req* the value must be specifed to do the simulation and the values of other parameters are optional

4.4.1 Buck-boost converter in open loop

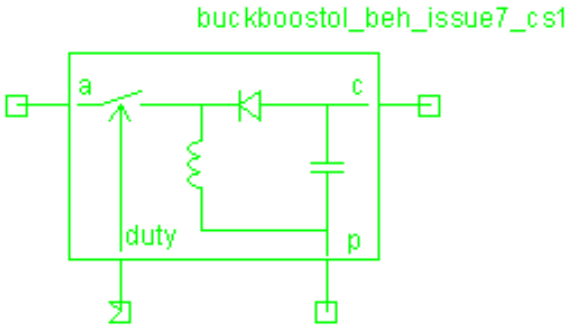


Figure 4.8. Buck-Boost converter open loop model

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	
cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*

4.4.2 Buck-boost converter in closed loop

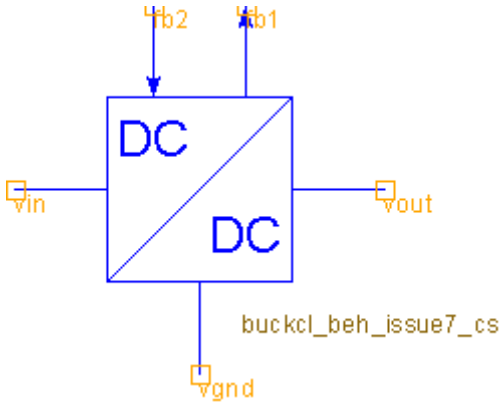


Figure 4.9 Buck boost converter closed loop model

Parameter	Description	Default value
l	Inductance Value	*req*
lr	Inductive resistance Value	
cap	Capacitance Value	*req*
esr	Capacitive resistance value	
ic	Initial current in inductor	
ron	The resistance during rise time	
roff	The resistance during fall time	
ton	The rise time	*req*
toff	The fall time	*req*
vref	The reference voltaje for the feedback	*req*
ss	Soft Start Time	
k	The loop gain	*req*
fs	The Switching Frequency	*req*
dutymax	Maximum duty cycle	
dutymin	Minimum Duty cycle	
dt	Delay Time	

4.5 Six-pulse diode rectifier

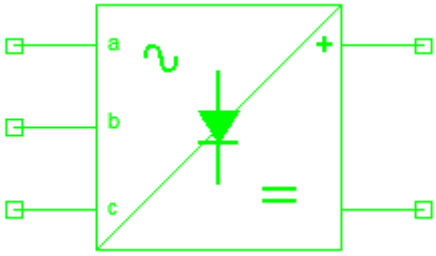


Figure 4.10. Three-phase six-pulse diode rectifier

4.6 Passive filter for inverter

In the filter section only the filter for inverter is available in the saber library the EMI filter is not available.

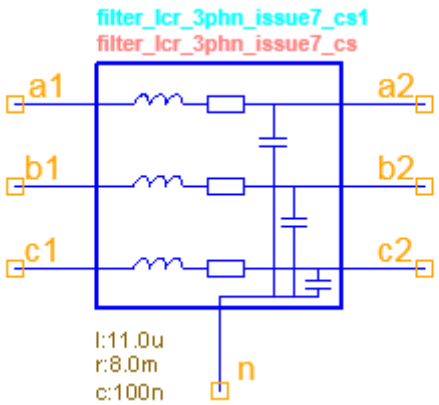


Figure 4.11. Three-phase filter for inverter

Parameter	Description		Default value
l	Inductance Value		*req*
lr	Inductive resistance Value		
c	Capacitance Value		*req*
esr	Capacitive resistance value		
r	The resistance		*req*
cleak	The leakage resistance		
tnom	Nominal Tempreture		
tc1	Proportional Temperature Coefficient		
tc2	Second Order Temperature Coefficient		

State-of-the art of parameter identification methods

5 State-of-the-art of parameter identification methods

System-level simulation and modeling are powerful design tools for system-level integration of distributed electrical and electronic architectures. These tools are applied to evaluate different architecture designs operating under multiple working conditions. They allow finding out optimum architecture designs, and detecting the situations in which the system exhibits poor dynamic response. Modeling and simulation can accelerate the design process while reducing experimental tests, thus reducing development costs and accelerating time-to-market[14].

However, the mathematical models are often based on parameters, whose values must be fully characterized in order to ensure an accurate response or output.

The accurate assessment of the parameters of the electrical systems and networks present in modern aircrafts is of paramount importance for the in situ non-intrusive and accurate evaluation and execution of powerful simulation methods to assess the performance of electrical systems and networks under different operating conditions[15]. By this way the behavior of such devices can be assessed, thus allowing ensuring and demonstrating significant performance progress, compliment with the specifications, and reliable and safe operation of the electrical systems involved in aircraft systems. This strategy also allows applying further optimization procedures while ensuring operating limits will not be exceed, hence minimizing the risk of incorrect operating conditions and additional or premature failure.

Parameter identification is an experimental methodology intended to determine the dynamics of a system by applying algorithms specifically designed to analyze the experimental data. However, the correct identification of parameters to accurately predict in a realistic way the behavior of the electrical systems present in MEA aircrafts is not an easy task due to the complexity of such systems and the variety of operating conditions. Therefore both the measurement system and the resulting identified parameters must have the ability to deal with the typical operating conditions, existing imbalances in the electrical systems involved, saturation effects or presence of harmonics among others.

Although in the technical bibliography different **parameter identification methods** are found, there are three main group of techniques applied can be classified as follows, which **are specially intended for white-box models** (see next section).

- Parameter identification methods based on **manufacturers' design data**
- **Offline** parameter identification methods based on measurements

[14]Virgilio Valdivia Guerrero, Behavioral Modeling and Identification of Power Electronics Converters and Subsystems Based on Transient Response, PhD Thesis, Universidad Carlos III, Madrid, 2013.

[15]N. Elkhattabi ; A. Sevigny ; P. Sicard ; A. Charette, Non-linear identification of the parameters of the ABC/abc model of the induction motor, Canadian Conference on Electrical and Computer Engineering, 2-5 May 2004

- **Online** parameter identification methods based on measurements

According to the classification above, the parameters can be obtained from either the electrical system design data or through measurements, the latter ones can be done under both, offline or online operating conditions.

Parameter identification and estimation under dynamic conditions has been effectively applied to identify circuit parameters and machine parameters based on the measurement of electrical magnitudes such as instantaneous voltages and currents. It is possible to do so using real time operating data[16]. However, **model parameters can depend on the operating conditions** and the identification of such parameters from experimental data can be done both offline or online either in the frequency-domain or in the time-domain.

5.1 Parameter identification methods based on design (manufacturer) data

Parameter identification methods based on design data require disposing of detailed design data sheets. For example, synchronous generator parameters have been obtained traditionally by using manufacturer's data sheets and further verified and improved by applying offline tests[17], some of which are described in IEEE Standards[18]. In addition, detailed design data sets are not always available since most manufacturers only provided basic information about their products. When analyzing the behavior and performance of a new electrical system which is installed in a specific application, it is appealing to dispose of a suitable and automatic parameter identification method. This strategy is also more efficient in terms of time requirements[19].

Even when the manufacturer's design data sheet is available, it often does not take into account the fact that **some parameters change under different operating conditions** and does not consider different effects including presence of harmonics, saturation effects, vibrations or existing imbalances in the electrical systems analyzed among others. Therefore, this approach has a limited scope.

[16]Y. Wehbe; L. Fan; Z. Miao, Least squares based estimation of synchronous generator states and parameters with phasor measurement units, North American Power Symposium (NAPS), 2012, pp. 1 – 6.

[17]E. Kyriakides, G. T. Heydt, and V. Vittal, Online parameter estimation of round rotor synchronous generators including saturation, IEEE Trans. Energy Conversion, vol. 20, no. 3, pp. 529–537, September 2005.

[18]IEEE Guide for Test Procedures for Synchronous Machines. Part I. Acceptance and Performance Testing. Part II. Test Procedures and Parameter Determination for Dynamic Analysis, IEEE Std 115-2009 (Revision of IEEE Std 115-1995), 2010, pp. 1 - 219.

[19]Henrik Neugebauer, Parameter Identification of a Permanent Magnet Synchronous Motor, Master's Thesis, Chalmers University of Technology, Gothenburg, Sweden 2012.

5.2 Parameter identification methods based on offline tests

Offline parameter identification methods are based on specific and dedicated experimental tests. For example, offline techniques to determine motor and generator parameters are often based on standstill operation (or rotor blocking condition) or when the machine rotates without any load. Off-PIM methods can further be classified as *self-commission* and *commission* methods[20]. Offline tests can efficiently provide information to identify the parameters of electrical and electronic systems. However, off-line tests present some drawbacks, since they require removing the system from normal service, thus being sometimes impractical and uneconomical. In addition, under different operating conditions, certain parameters may change slightly, so offline methods may not satisfactorily characterize some electrical machines and systems when a high degree of accuracy is pursued. For example, when dealing with electrical machines such as generators and thus motors, some effects such as saturation are not reasonably accounted in offline measurements.

5.3 Parameter identification methods based on online tests (white-box)

Online parameter identification methods are usually based on applying a small disturbance to the equipment under analysis during normal service, thus do not affecting the normal operation of the system.

Online parameter identification methods can be broadly classified into spectral analysis techniques, observer-based techniques, or techniques based on real-time operating data.

Spectral Analysis Techniques. Spectral analysis techniques are based on the measured experimental response to an intentionally injected test signal or an already present specific harmonic in the current/voltage spectrum. Therefore voltages and/or currents in the analyzed electrical system are acquired and sampled. The model parameters are derived from the spectral analysis of such samples[21]. This technique can lead to some inaccuracies, especially when the perturbation signal is applied I under no-load conditions.

Extended Kalman filter (EKF) based techniques have been effectively applied without injection of any test signal for parameter estimation during normal operating conditions, or even by using start-up transient current and voltage measurements[22], specifically in electrical motors and generators. Although no external signals are injected, the wide-band harmonics

[20]Ivan Jadric, Modeling and control of a synchronous generator with electronic load, Master Thesis, Virginia Polytechnic Institute and State University, 1998.

[21]H. A. Toliyat; E. Levi; M. Raina, A review of RFO induction motor parameter estimation techniques, IEEE Transactions on Energy Conversion, 2003, Vol. 18 (2), pp. 271 – 283

[22]P. Huynh; H. Zhu; D. Aliprantis, Non-intrusive parameter estimation for single-phase induction motors using transient data, 2015 IEEE Power and Energy Conference at Illinois (PECI), 2015, pp. 1 – 8.

of the output voltage due to the switching of the PWM inverter act as the excitation[23]. The EKF is recursive estimation method since the estimated value of the current state is inferred from the previous estimated value[24].

***Based on real time operating data*.** Standard parameters used to generate accurate models of electrical systems although can be obtained from specific offline tests or from manufacturer data, it is known they can experiment some changes throughout the useful life of such devices due to materials aging, wear or due to repairs[25].

Parameter identification can be based from online data measurements of electrical variables done at the terminals of the electrical system. To this end data sets recorded under a wide range of operating conditions are required to ensure the consistency parameter identification method.

- Since these data have been acquired at the electrical system terminals under real operating conditions, they include effects such as harmonics, unbalances, switching effects or saturation among others.
- Noise filtering and rejection of outlier data can be employed to increase the reliability and accuracy of the estimates.
- Observers can also be applied to infer electrical magnitudes that otherwise are unknown since they are not measured.

The estimated parameters can be compared to available manufacturer data to assess the accuracy of the parameter identification method

[23]H. A. Toliyat; E. Levi; M. Raina, A review of RFO induction motor parameter estimation techniques, IEEE Transactions on Energy Conversion, 2003, Vol. 18 (2), pp. 271 – 283

[24]X. Zhan, G. Zeng, J. Liu, Q. Wang, S. Ou, A Review on Parameters Identification Methods for Asynchronous Motor, International Journal of Advanced Computer Science and Applications, Vol. 6(1), 2015

[25]E. Kyriakides, G. T. Heydt, and V. Vittal, Online parameter estimation of round rotor synchronous generators including saturation, IEEE Trans. Energy Conversion, vol. 20, no. 3, pp. 529–537, September 2005.

5.4 Parameter identification based on design data. Three-stage electrical generator

1. Final model. Equations description
2. Parameters involved
3. How to measure each parameter

Manufacturer data:
Rated voltages (delta/star): 13800 V (line-to-line), Y-connected[26]
Rated currents (delta/star): 7823.51 A, Y-connected
Rated speed: 60*60/(four-pole/2) = 180 rpm
Rated excitation current: 1087 A
Rated power: 187 MVA
Rated frequency: 60 Hz

1. Open circuit test at rated speed (field currents – line generated voltages) [27].
The generator is rotated at the rated speed, all the terminals are disconnected. The field current is increased in steps and measured and the line-to-line voltage is measured too.

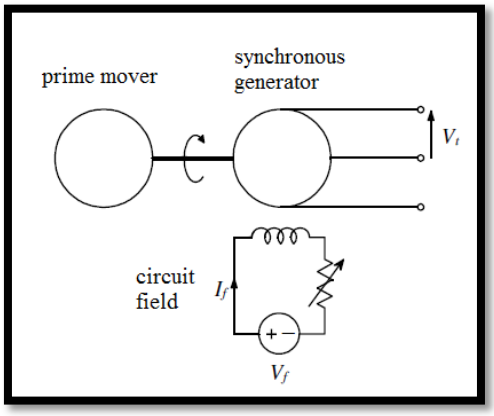


Figure 5.1. Open circuit test

Table 5.1. Open circuit test data

Inf[%]	5	10	15	20	25	30	35	40	45	50	55	60
Vf	3.52	7.04	10.56	14.08	17.60	21.125	24.64	28.16	31.69	35.21	38.73	42.25
If	54.5	108.7	163.1	217.4	271.6	326.3	380.5	434.8	489.3	543.6	597.9	652.2
Vt	755	1509	2265	3015	3673	4531	5281	6035	6793	7549	8302	9056

[26] P. Kundur, Power System Stability and Control, McGraw-Hill, 1994.

[27] IEC 60034-4. Rotating electrical machines – Part 4: Methods for determining synchronous machine quantities from tests

Inf[%]	65	70	75	85	95	105	110	120	140
Vf	45.74	49.29	52.81	59.86	66.90	73.84	77.36	84.40	98.48
If	706.4	761.3	815.9	925.1	1034	1141	1196	1305	1524
Vt	9785	10463	11089	12320	13347	14261	14562	15240	16299

2. Short circuit test at rated speed (field current – armature current).

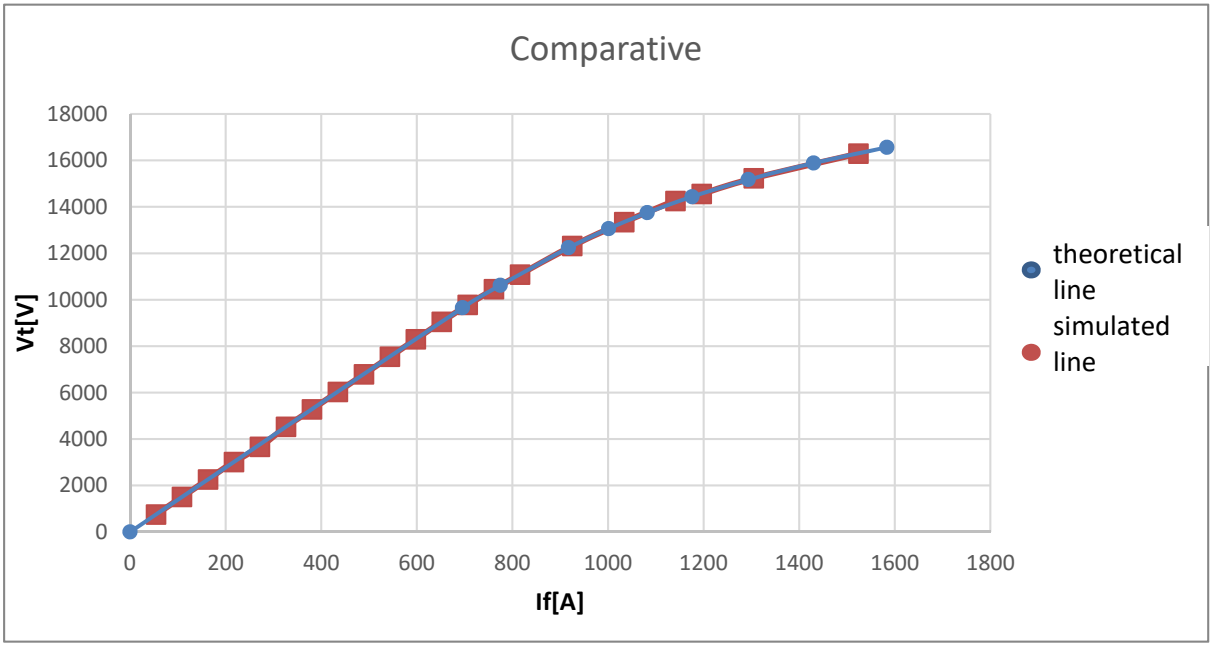


Figure 5.2. Saturation curve

The generator is rotated at the rated speed, with all terminals in short-circuit. The field current is increased in steps and measured, and the armature current is also measured.

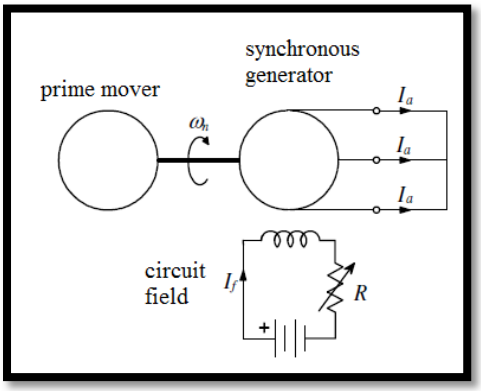


Figure 5.3. Short circuit test

Table 5.2. Short circuit test data

Vn %	20 higher	15 higher	10 higher	9 higher	5 higher	Vn higher	50 lower	75 inf lower	90 inf lower	99 inf lower
Vf[V]	84.43	80.914	77.40	76.88	73.89	70.36	35.18	17.59	7.036	0.7036
If [A]	1305	1251	1196	1188	1142	1088	544	272	109	11
Ipp [A]	11130	10675	10210	10138	9744	9279	4639	2319	930	93
Ia RMS[A]	7870.1	7548.4	7219.6	7168.6	6890.4	6561.2	3280.6	1639.8	657.6	65.8

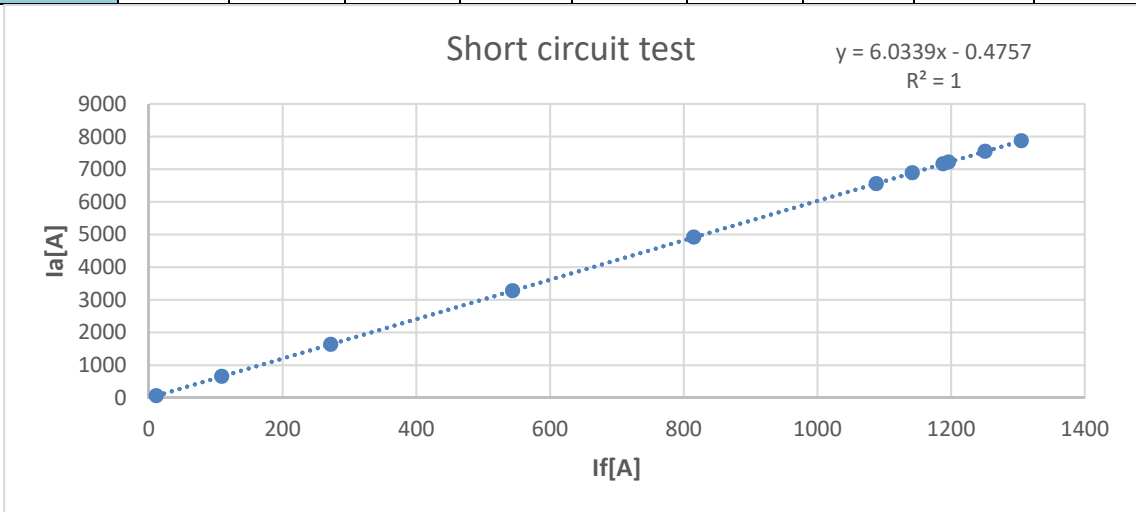


Figure 5.4. Short circuit test data

3. Armature resistance (dc voltage – armature current).

A DC voltage is applied to two of the terminals, the armature current and dc voltage are measured.

Table 5.3. Armature resistance test data

V DC[V]	5
Ia[A]	860.1

$$\text{Armature resistance} = \frac{\left(\frac{\text{V DC}}{\text{Ia}}\right)}{2} = 2.93066\ m\Omega$$

- **Matlab parameter's calculations.**

```
% synchronous machine
% rated voltage
VN=13800; % V
% rated power
SN=187e6; % VA
% rate current
IN=SN/sqrt(3)/VN; %  7823.51 A
```

```
% pair of poles
p=20; %
% Rated field current
IfN=1087; % A
% Rated frequency
fN=60 ; % Hz
% Open circuit test at rated speed (field currents – line generated voltages):
field_current_oc=[54.5 380.5 652.2 706.4  815.9 925.1 1034 1196 1305 1524]; % A
terminal_voltage_oc=[755 5281 9056 9785  11089  12320 13347 14562 15240 16299]; %V
% Short circuit test at rated speed (field current – armature current):
field_current_sc=[1305 1251 1196 1188 1142 1088 544 272 109 11]; % A
armature_current_sc=[7870.1 7548.4  7219.6 7168.6 6890.4 6561.2 3280.6 1639.8 657.6
65.8]; %A
%
Iacc1=IN;
Ifcc1=interp1(armature_current_sc,field_current_sc,Iacc1,'spline'); % 1297.5
Ifoc1=Ifcc1;
Voc1=interp1(field_current_oc,terminal_voltage_oc,Ifoc1,'spline'); % 15197
Voc2=VN;
Ifoc2=interp1(terminal_voltage_oc,field_current_oc,Voc2,'spline'); % 1809.9
Ifcc2=Ifoc2;
% cftool
%  Linear model Poly1:
 %     f(x) = p1*x + p2
% Coefficients (with 95% confidence bounds):
     %  p1 =       6.034  (6.031, 6.037)
      % p2 =     -0.8376  (-3.923, 2.248)
%
% Goodness of fit:
 %  SSE: 38.43
%   R-square: 1
  % Adjusted R-square: 1
 % RMSE: 2.192
Iacc2 = 6.034*Ifcc2 -0.8376; % 6575.8

%%%%%%%%% Parameters %%%%%%%%%
% Rated impedance
ZN=VN^2/SN; % 1.0184 ohm
% Unsaturated value of the direct-axis synchronous impedance (air-gap line)
Zsu=(Voc1/sqrt(3))/Iacc1; % 1.1215ohm
Zsu_pu=Zsu/ZN; % 1.1012 pu
% Saturated value of thedirect-axis synchronous impedance:
```

```
Zs=(Voc2/sqrt(3))/Iacc2; %  1.2116 ohm
Zs_pu=Zs/ZN; %  1.1897 pu
% Armature resistance (dc voltage – armature current):
Vdc=5; % V applied to two of the tenninals
Iadc=860.1; % A
Rs=Vdc/(2*Iadc); % 0.0029 ohm
Rs_pu=Rs/ZN; % 0.0029 pu
% Unsaturated value of the direct-axis synchronous reactance (air-gap line)
Xdu=sqrt(Zsu^2-Rs^2); % 1.1215 ohm
Xdu_pu=Xdu/ZN; % 1.1012 pu
% Saturated value of the synchronous reactance:
Xd=sqrt(Zs^2-Rs^2); % 1.2116 ohm
Xd_pu=Xd/ZN; % 1.1897 pu
%
```

Per-phase equivalent circuit of the cylindrical rotor synchronous generator:

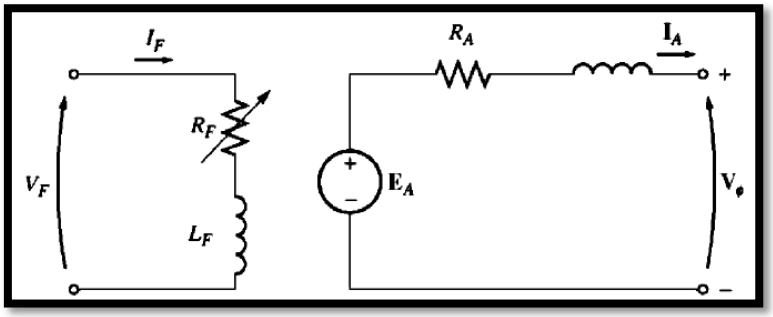


Figure 5.5 equivalent circuit of the cylindrical rotor synchronous generator

$$E_A = R_A \cdot I_A + jX_s \cdot I_A + V_s$$

5.5 Parameter identification based offline data. Three-stage electrical generator

4. Low slip test.

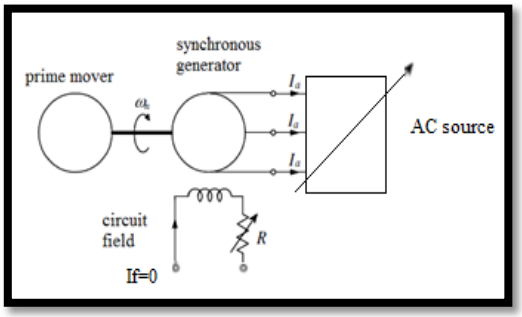


Figure 5.6. Low slip test

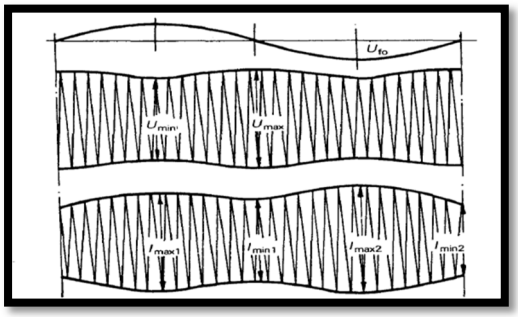


Figure 5.7. Low slip test data

The generator is driven by means of a prime mover at a speed slightly different from the synchronous speed, less than 1%, and the field windings on the rotor must be kept open-circuited. After, the armature winding is connected to a rated frequency symmetrical three-phase low-voltage supply (1% to 20% of rated voltage). During switching on and off of the supply, the excitation winding should be closed (short-circuited or through a discharge resistance) to avoid possible damage. Armature current and voltage and the slip-ring voltage and slip are measured and recorded. If the residual voltage measured before the test is larger than 0.3 of the supply test voltage, the rotor should be demagnetized. Demagnetizing might be done, for example, by connecting the field winding to a low-frequency source with current about 0.5 of the no-load rated voltage excitation current of the tested machine and gradually decreasing its amplitude and frequency (the latter if possible).

Table 5.4. Low slip test data

V slip min [V]	380
V slip max[V]	770
I slip min [A]	1000
I slip max [A]	1070

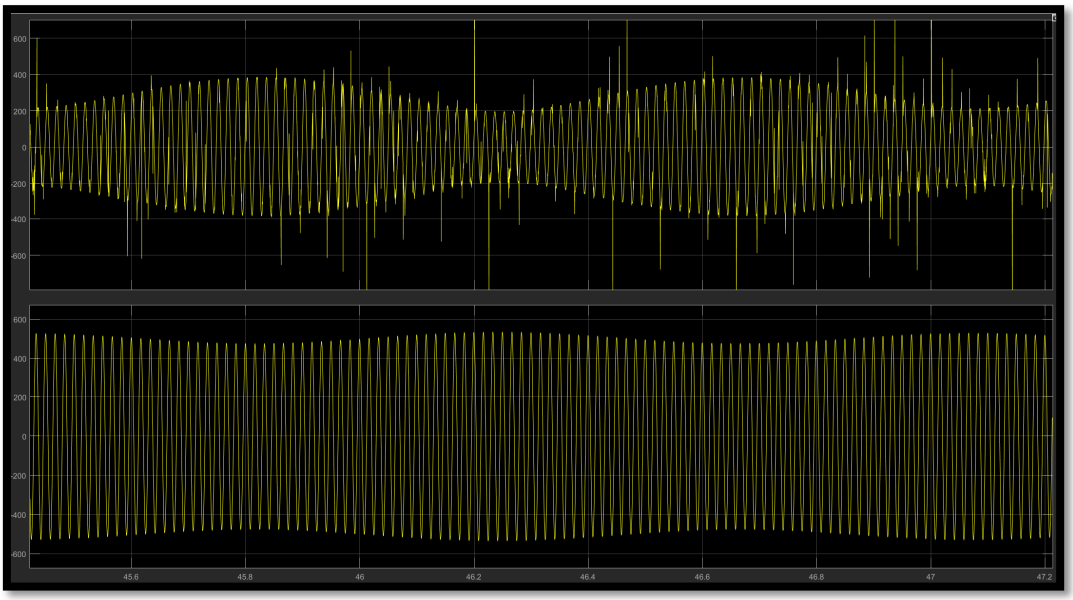


Figure 5.8. Low slip test simulation results

- **Matlab parameter's calculations.**

```
% synchronous machine
% rated voltage
VN=13800; % V
% rated power
SN=187e6; % VA
% rate current
```

```
IN=SN/sqrt(3)/VN; %  7823.51 A
% pair of poles
p=20; %
% Rated field current
IfN=1087; % A
% Rated frequency
fN=60 ; % Hz
% Open circuit test at rated speed (field currents – line generated voltages):
field_current_oc=[54.5 380.5 652.2 706.4  815.9 925.1 1034 1196 1305 1524]; % A
terminal_voltage_oc=[755 5281 9056 9785  11089  12320 13347 14562 15240 16299]; %V
% Short circuit test at rated speed (field current – armature current):
field_current_sc=[1305 1251 1196 1188 1142 1088 544 272 109 11]; % A
armature_current_sc=[7870.1 7548.4  7219.6 7168.6 6890.4 6561.2 3280.6 1639.8 657.6
65.8]; %A
%
Iacc1=IN;
Ifcc1=interp1(armature_current_sc,field_current_sc,Iacc1,'spline'); % 1297.5
Ifoc1=Ifcc1;
Voc1=interp1(field_current_oc,terminal_voltage_oc,Ifoc1,'spline'); % 15197
Voc2=VN;
Ifoc2=interp1(terminal_voltage_oc,field_current_oc,Voc2,'spline'); % 1809.9
Ifcc2=Ifoc2;
% cftool
%  Linear model Poly1:
 %     f(x) = p1*x + p2
% Coefficients (with 95% confidence bounds):
     %  p1 =       6.034  (6.031, 6.037)
      % p2 =     -0.8376  (-3.923, 2.248)
%
% Goodness of fit:
 %  SSE: 38.43
%   R-square: 1
  % Adjusted R-square: 1
 % RMSE: 2.192
Iacc2 = 6.034*Ifcc2 -0.8376; % 6575.8

%%%%%%%%% Parameters %%%%%%%%%
% Rated impedance
ZN=VN^2/SN; % 1.0184 ohm
% Unsaturated value of the direct-axis synchronous impedance (air-gap line)
Zsu=(Voc1/sqrt(3))/Iacc1; % 1.1215ohm
Zsu_pu=Zsu/ZN; % 1.1012 pu
```

```
% Saturated value of thedirect-axis synchronous impedance:
Zs=(Voc2/sqrt(3))/Iacc2; %  1.2116 ohm
Zs_pu=Zs/ZN; %  1.1897 pu
% Armature resistance (dc voltage – armature current):
Vdc=5; % V applied to two of the tenninals
Iadc=860.1; % A
Rs=Vdc/(2*Iadc); % 0.0029 ohm
Rs_pu=Rs/ZN; % 0.0029 pu
% Unsaturated value of the direct-axis synchronous reactance (air-gap line)
Xdu=sqrt(Zsu^2-Rs^2); % 1.1215 ohm
Xdu_pu=Xdu/ZN; % 1.1012 pu
% Saturated value of the synchronous reactance:
Xd=sqrt(Zs^2-Rs^2); % 1.2116 ohm
Xd_pu=Xd/ZN; % 1.1897 pu
%

% Low slip test
% d-axis
V_slip_max=770;
Ia_slip_min=1000;
% q-axis
V_slip_min=380;
Ia_slip_max=1070;
%
Xd_slip=V_slip_max/sqrt(3)/Ia_slip_min; % 0.4446
Xq_slip=V_slip_min/sqrt(3)/Ia_slip_max; % 0.2050
% Saliency ratio
Xq_Xd_slip=Xq_slip/Xd_slip; % 0.4612
% Unsaturated value of the quadrature-axis synchronous reactance (air-gap line)
Xqu=Xdu*Xq_Xd_slip; %  0.5172 ohm
Xqu_pu=Xqu/ZN; % 0.5079 pu
% Saturated value of the quadrature-axis synchronous reactance
Xq=Xd*Xq_Xd_slip; %   0.5588 ohm
Xq_pu=Xq/ZN; % 0.5487 pu
```

Per-phase equivalent circuit of the salient-pole rotor synchronous generator:

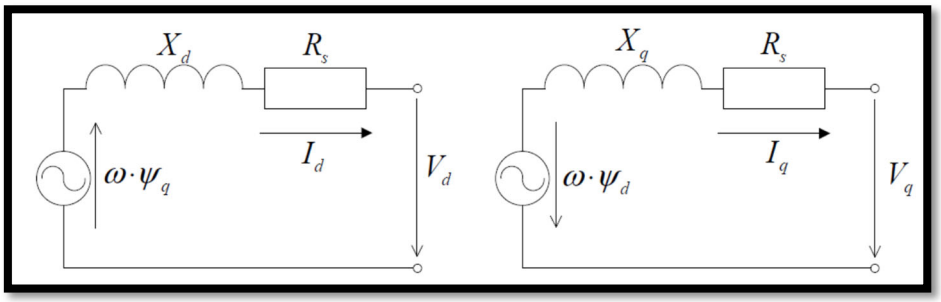


Figure 5.9. equivalent circuit of the salient-pole rotor synchronous generator

$$V_d = -R_s \cdot I_d - jX_d \cdot I_d - \omega \cdot \psi_q$$
$$V_q = -R_s \cdot I_q - jX_q \cdot I_q + \omega \cdot \psi_d$$
$$\omega \cdot \psi_d = -X_d \cdot I_d$$
$$\omega \cdot \psi_q = -X_q \cdot I_q$$

5. Zero power-factor test.

A synchronous generator driven at rated speed and with constant armature current. Several inductive loads are connected in order to operate with zero lagging power factor. By a proper adjustment to the field excitation and the load, both at the same time, readings of the terminal voltage variation with the field current when a constant rated armature current is drawn can be measured in steps. From those readings, the zero power-factor characteristic is plotted together on one graph with the open circuit characteristic.

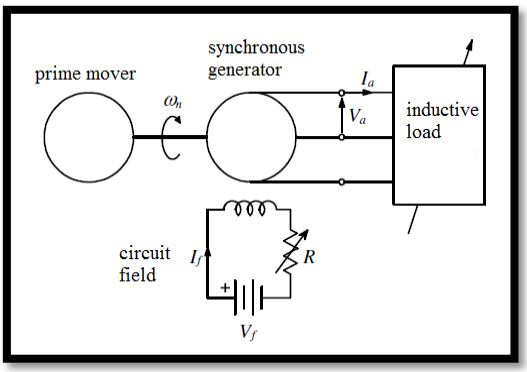


Figure 5.10. Zero Power-factor test

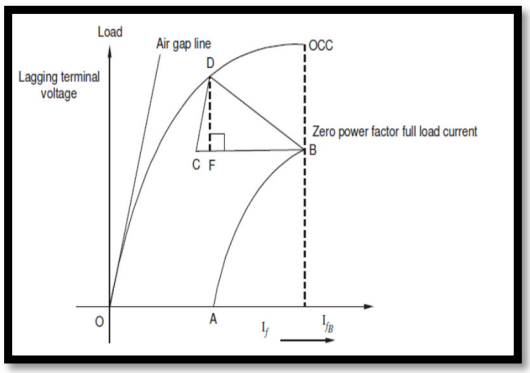


Figure 5.11. Potier tirangle

Table 5.5. Zero power factor test data

Vf [V]	84.43	98.50	105.54	112.57	168.86	175.90
L [mΩ]	0.1e-7	6.2e-4	0.89e-3	1.7e-3	2.2e-3	2.9e-3
If[A]	1314	1523	1626	1735.8	2115	2710
Vt[V]	134.35	3065.7	4553.8	6123.5	11313.7	14707.8

- **Matlab parameter's calculations.**

This code is a continuation of the previous one:

The results are the following:

```
% Zero power-factor test at rated speed (field current – terminal voltage):
field_current_zpf=[1314 1523 1626 1735.8 2019 2115 2710 2844 2967 3321 ]; % A
terminal_voltage_zpf=[134.35 3065.7 4553.8 6123.5 9758.1 11313.7 14707.8 15309.1 15843.3
17402.8]; % V
%
Vzpf_point_A=0;
Ifzpf_point_A=interp1(terminal_voltage_zpf,field_current_zpf,Vzpf_point_A,'spline'); %  1304
%
Vzpf_point_B=VN; % 13800
Ifzpf_point_B=interp1(terminal_voltage_zpf,field_current_zpf,Vzpf_point_B,'spline'); % 2514.2
%
Vzpf_point_C=VN; % 13800
Ifzpf_point_C=Ifzpf_point_B-Ifzpf_point_A; % 1200.1
%
Voc_VN_div_2=VN/2; % 6900
Ifoc_VN_div_2=interp1(terminal_voltage_oc,field_current_oc,Voc_VN_div_2,'spline'); % 496.7118
Voc_VN_div_10=VN/10; % 1380
Ifoc_VN_div_10=interp1(terminal_voltage_oc,field_current_oc,Voc_VN_div_10,'spline'); %99.8272
m=(Voc_VN_div_2-Voc_VN_div_10)/(Ifoc_VN_div_2-Ifoc_VN_div_10); %13.9083
% cftool
% Linear model Poly3:
%     f(x) = p1*x^3 + p2*x^2 + p3*x + p4
% Coefficients (with 95% confidence bounds):
%       p1 =  -2.048e-06  (-3.042e-06, -1.054e-06)
%       p2 =   0.0006564  (-0.001727, 0.00304)
%       p3 =       14.45  (12.87, 16.04)
%       p4 =      -75.69  (-363.8, 212.4)
%
% Goodness of fit:
%   SSE: 5.53e+04
%   R-square: 0.9997
%   Adjusted R-square: 0.9996
%   RMSE: 96
point_D=solve('y = -2.048e-06*x^3 + 0.0006564*x^2 + 14.45*x + -75.69','y=13.9083*(x-
1200)+13800')
Ioc_point_D=eval(point_D.x(1)); % 1.8025e+03
Voc_point_D=eval(point_D.y(1)); % 1.6109e+04
%
% stator leakage reactance (Potier reactance)
```

```
Xl=((Voc_point_D-Vzpf_point_B)/sqrt(3))/IN; % 0.1704
Xl_pu=Xl/ZN; % 0.1165
Ll=Xl/(2*pi*60) % 3.1461e-04
%
% d-axis magnetizing reactance
Xmd=Xd-Xl; % 1.0412
Xmd_pu=Xmd/ZN; % 1.0224
% q-axis magnetizing reactance
Xmq=Xq-Xl; % 0.3884
Xmq_pu=Xmq/ZN; % 0.3814

Lmd=Xmd/(2*pi*60) % 0.0029
Lmq=Xmq/(2*pi*60) % 0.0012
```

6. Sudden three-phase short-circuit test.

Many of the transient and subtransient reactance parameters can be computed from a suitable oscillogram of a short-circuit current suddenly applied to the generator terminals. The interest here is to calculate the direct-axis transient and subtransient synchronous reactance. The quadrature-axis transient and subtransient synchronous reactances are often of little significance during the short-circuit conditions.

This test is performed up to 70% of rated terminal voltage and with no load operation. The synchronous generator is driven at rated speed. Therefore, a three-phase short-circuit is suddenly applied at the generator terminals. The excitation current and the short-circuit current are measured.

The rms amplitude of the total ac component of the short-circuit current in one phase at any instant of time can be defined as follows:[28]

[28] J. P. Martin, C. E. Tindall and D. J. Morrow, Synchronous machine parameter determination using the sudden short-circuit axis currents, IEEE Transactions on Energy Conversion, vol. 14, no. 3, pp. 454-459, Sep 1999.

$$i_a = V_m\left[\underbrace{\frac{1}{X_d}}_{\text{steady state}} + \underbrace{\left(\frac{1}{X_d'}-\frac{1}{X_d}\right)\exp\left(-\frac{t}{\tau_d'}\right)}_{\text{transient}} + \underbrace{\left(\frac{1}{X_d''}-\frac{1}{X_d'}\right)\exp\left(-\frac{t}{\tau_d''}\right)}_{\text{sub-transient}}\right]cos(w_o t+\lambda)$$

$$\underbrace{-\frac{V_m}{2}\left(\frac{1}{X_d''}+\frac{1}{X_q''}\right)exp\left(-\frac{t}{\tau_\alpha}\right)cos(\lambda)}_{\text{aperiodic}} \quad \underbrace{-\frac{V_m}{2}\left(\frac{1}{X_d''}-\frac{1}{X_q''}\right)exp\left(-\frac{t}{\tau_\alpha}\right)cos(2w_o t+\lambda)}_{\text{double frequency}}$$

The decomposition of the axis currents is:

$$i_d = V_m\left[\underbrace{\frac{1}{X_d}}_{\text{steady state}} + \underbrace{\left(\frac{1}{X_d'}-\frac{1}{X_d}\right)\exp\left(-\frac{t}{\tau_d'}\right)}_{\text{transient}} + \underbrace{\left(\frac{1}{X_d''}-\frac{1}{X_d'}\right)\exp\left(-\frac{t}{\tau_d''}\right)}_{\text{sub-transient}}\right]$$

$$\underbrace{-\frac{V_m}{X_d''}exp\left(-\frac{t}{\tau_\alpha}\right)cos(w_o t)}_{\text{armature}}$$

$$i_q = \underbrace{-\frac{V_m}{X_q''}exp\left(-\frac{t}{\tau_\alpha}\right)sin(w_o t)}_{\text{armature}}$$

- Iq study:

- **Matlab code:**

```
c=1;
for k=3:1:length(iqmes)
   if (iqmes(k) < iqmes(k-1)) & (iqmes(k-1) > iqmes(k-2))
      punt(c)=iqmes(k-1);
      c=c+1;
   end
end
figure (1)
hold on
for k=1:1:length(punt)
   pos(k)=find(iqmes==punt(k));
   tfit(k)=tmes(pos(k));
   iqfit(k)=iqmes(pos(k));
   plot(tfit(k),iqfit(k),'*');
end
plot(tmes,iqmes);
hold off
```

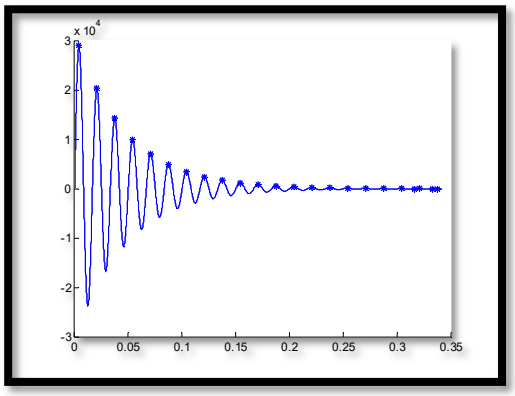


Figure 5.12. Iq measured

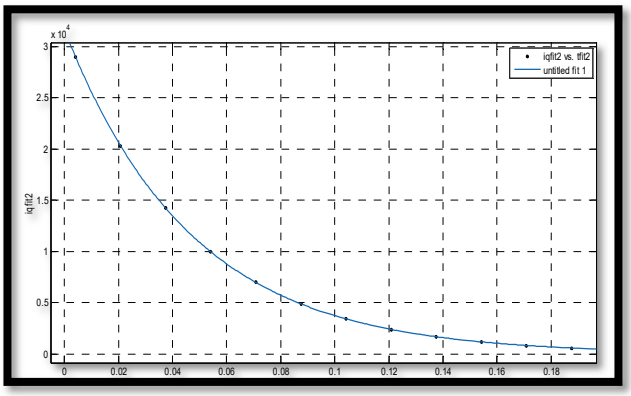


Figure 5.13. Iq fitting

- **Matlab code:**

```
tfit2=[tfit(1:1:round(length(tfit)/2))];
iqfit2=[iqfit(1:1:round(length(iqfit)/2))];
a = 3.162e+04;
b = 21.33;
tcalc=tmes;
for k=1:1:length(tcalc)
 iqcalc(k)=a*exp(-b*tcalc(k))*sin(2*pi*60*tcalc(k));
end
plot(tmes,iqmes,'r',tcalc,iqcalc,'b')
```

Vm=1.665e4
Xq_pu''=1.665e4/3.162e+04/ZN = 0.517053885399516
Tq''=1/21.33 = 0.046882325363338

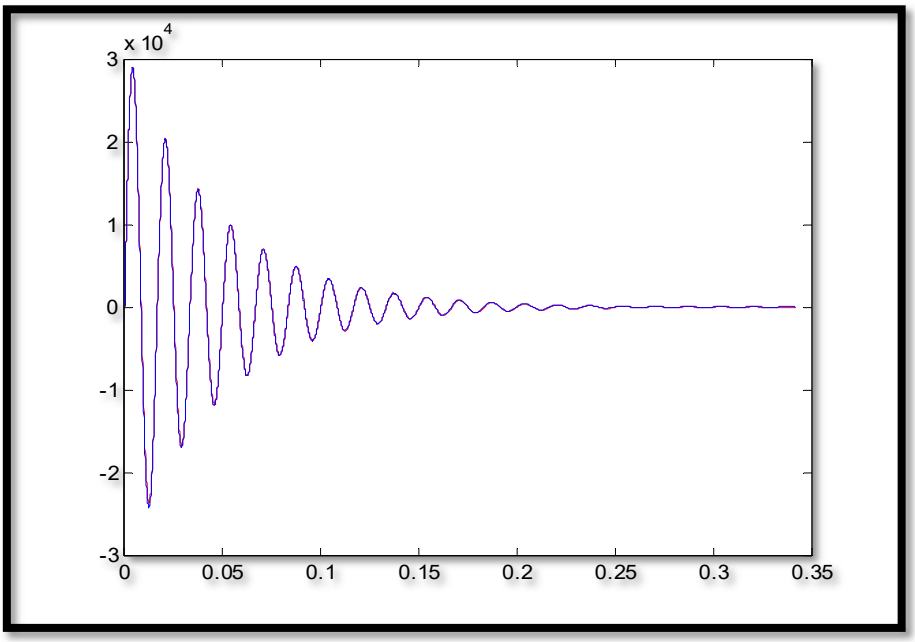


Figure 5.14. Iq measured and Iq calculated

- Id study:

```
tmesid=dades_id_v1(:,1);
idmes=dades_id_v1(:,2);

c=1;
for k=3:1:length(idmes)
   if (idmes(k) < idmes(k-1)) & (idmes(k-1) > idmes(k-2))
      punt(c)=idmes(k-1);
      c=c+1;
   end
end
figure (1)
hold on
for k=1:1:length(punt)
   pos(k)=find(idmes==punt(k));
   tidfit1(k)=tmesid(pos(k));
   idfit1(k)=idmes(pos(k));
   plot(tidfit1(k),idfit1(k),'*');
end
plot(tmesid,idmes);
```

```
c=1;
for k=4000:10000:length(tmesid)
   tidfit2(c)=tmesid(k);
   idfit2(c)=idmes(k);
   plot(tidfit2(c),idfit2(c),'*');
   c=c+1;
end
hold off
```

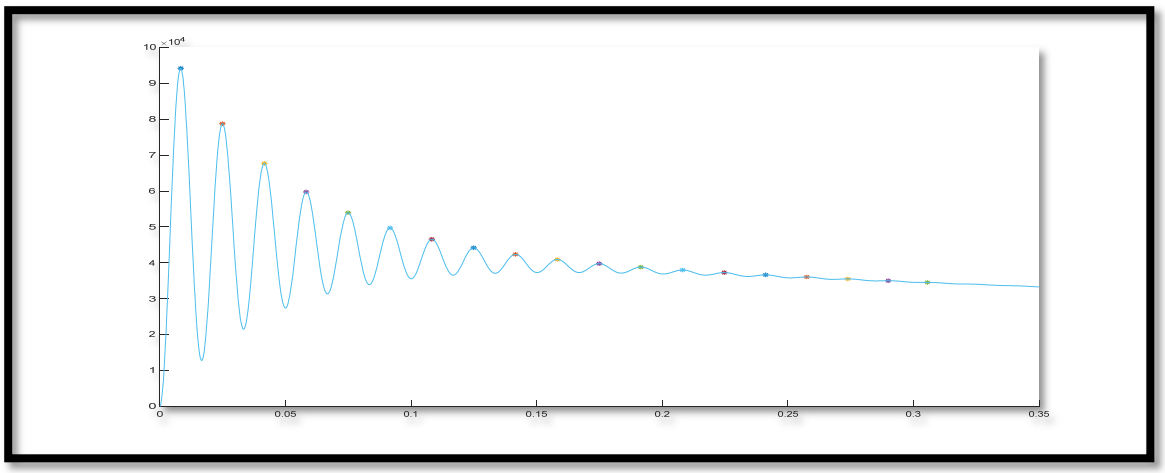


Figure 5.15. Id measured

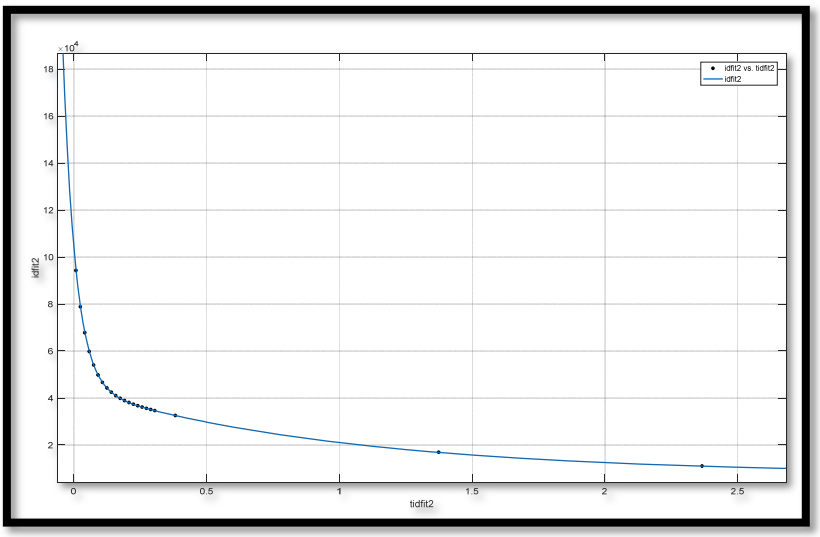


Figure 5.16. Id fitting

```
tidfit=[tidfit1,tidfit2];
idfit=[idfit1,idfit2];

tidfit2=[tidfit(1:1:round(length(tidfit)/1))];
idfit2=[idfit(1:1:round(length(idfit)/1))];
a = 7.46033e3;
b = 1e4;
c = 3.65e4;
d = 5.371e4;
tau1=0.04286;
tau2=1.01;
tau3=0.0456;


tcalc=tmesid;
for k=1:1:length(tcalc)
idcalc(k)=a+b*exp(-tcalc(k)/tau1)+c*exp(-tcalc(k)/tau2)-d*exp(-
tcalc(k)/tau3)*cos(2*pi*60*tcalc(k));

end
plot(tmesid,idmes,'r',tcalc,idcalc,'b')
```

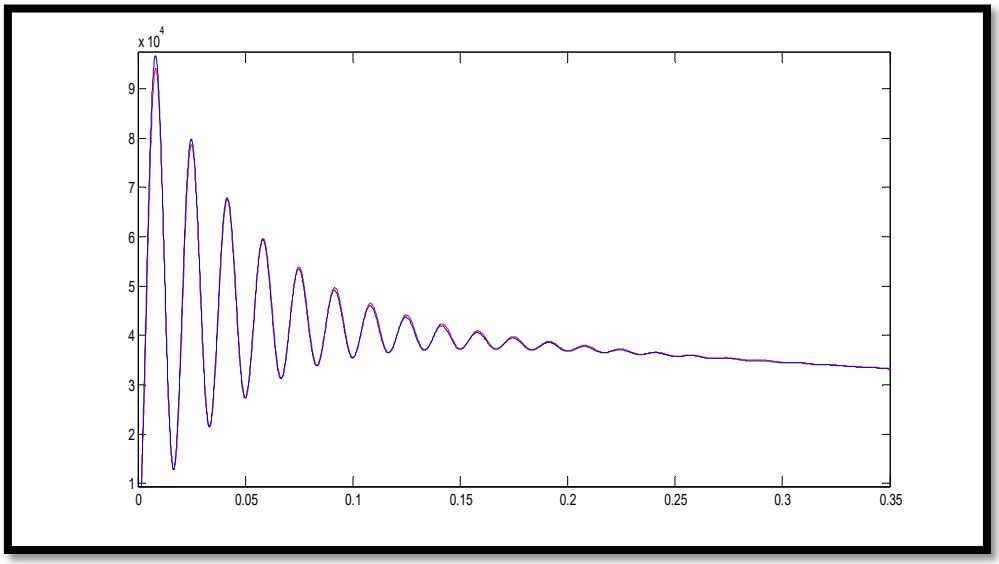


Figure 5.17. Id measured and Id calculated

- **Calculation of Xd, Xd ', Xd' '**

$$i_d = A + Be^{\left(\frac{-t}{\tau_1}\right)} + Ce^{\left(\frac{-t}{\tau_2}\right)} - De^{\left(\frac{-t}{\tau_3}\right)} \cdot \cos(2 \cdot \pi \cdot 60 \cdot t)$$

$$y = 7.46033 \cdot 10^3 + 3.65 \cdot 10^4 e^{(-x/1.01)} + 1 \cdot 10^4 e^{(-x/0.04286)} - 5.371 \cdot 10^4 e^{(-x/0.0456)} \cdot \cos(2 \cdot \pi \cdot 60 \cdot x)$$

Xd:

$$A = \frac{V_m}{X_d} \Rightarrow \quad 7.46033 \cdot 10^3 = \frac{9558}{X_d} = \frac{\frac{16553}{\sqrt{3}}}{X_d} \quad \Rightarrow X_d = 1.2810$$

Xd':

$$B = V_m\left(\frac{1}{X_d{}'} - \frac{1}{X_d}\right) \Rightarrow \quad 3.65 \cdot 10^4 = \frac{16553}{\sqrt{3}}\left(\frac{1}{X_d{}'} - \frac{1}{X_d}\right) \quad \Rightarrow \left(\frac{1}{X_d{}'} - \frac{1}{X_d}\right) = \frac{3.65 \cdot 10^4}{\frac{16553}{\sqrt{3}}} = 3.8192$$

$$\Rightarrow \frac{1}{X_d{}'} = 3.8192 + \frac{1}{1.2810} \Rightarrow X_d' = 0.2174$$

Xd'':

$$C = V_m\left(\frac{1}{X_d{}''} - \frac{1}{X_d{}'}\right) \Rightarrow \quad 1 \cdot 10^4 = \frac{16553}{\sqrt{3}}\left(\frac{1}{X_d{}''} - \frac{1}{X_d{}'}\right) \quad \Rightarrow \left(\frac{1}{X_d{}''} - \frac{1}{X_d{}'}\right) = \frac{1 \cdot 10^4}{\frac{16553}{\sqrt{3}}} = 1.0464$$

$$\Rightarrow \frac{1}{X_d{}''} = 1.0464 + \frac{1}{0.2174} \Rightarrow X_d'' = 0.1771$$

Time constants:

Td'= 1.01
Td''= 0.04286

6.2. Checking the time constants obtained from the sudden three-phase short-circuit test.

The time constants obtained from the sudden three-phase short-circuit test are the following:

Td'= 1.01 *Td''= 0.04286* *Tq''=0.04688*

To make sure that the values obtained are correct, they will be calculated again by an analytical method. [1]

- **Direct axis.**

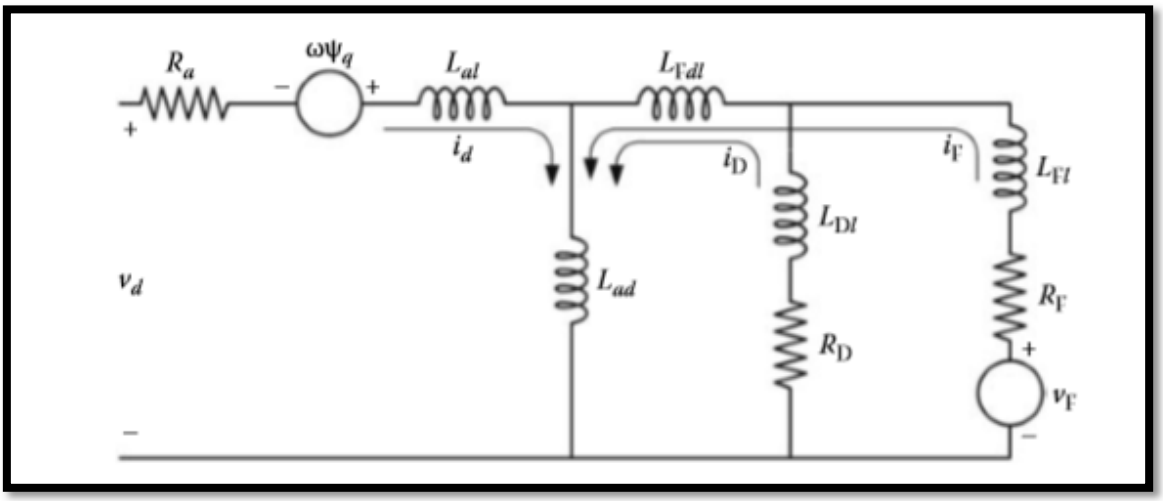


Figure 5.18. Equivalent circuit of work[1].

The equations are the following:

$$Td'' + Td' = \frac{L_{sF}}{R_F} + \frac{L_{sd}}{R_D}$$

$$= \frac{1}{R_F}\left(L_{FF} - \frac{(L_{ad})^2}{L_d}\right) + \frac{1}{R_D}\left(L_{DD} - \frac{(L_{ad})^2}{L_d}\right) \qquad (5.76a)$$

$$Td'' \cdot Td' = \frac{L_{sF}.L_{sD} - (L_{MD})^2}{R_F \cdot R_D}$$

$$= \frac{1}{R_F \cdot R_D}\left[\left(L_{FF} - \frac{(L_{ad})^2}{L_d}\right)\cdot\left(L_{DD} - \frac{(L_{ad})^2}{L_d}\right) - \left(L_{FD} - \frac{(L_{ad})^2}{L_d}\right)^2\right]$$

$$= \frac{1}{R_F \cdot R_D}\left(L_{FF} \cdot L_{DD} - (L_{FD})^2 - \frac{(L_{ad})^2(L_{Fl} + L_{Dl})}{L_d}\right) \qquad (5.76b)$$

Juan A. Martinez-Velasco, Power System Transient, Parameter determination. CRC Press, 2010.

Taking into account that:

$$L_d = (L_{ad} + L_{al})$$

$$L_{FF} = (L_{FD} + L_{Fl})$$

$$L_{DD} = (L_{FD} + L_{Dl})$$

$$L_{FD} = (L_{ad} + L_{Fdl})$$

And:

$$L_{sF} = \left(L_{FF} - \frac{(L_{ad})^2}{L_d}\right)$$

$$L_{sD} = \left(L_{DD} - \frac{(L_{ad})^2}{L_d}\right)$$

$$L_{MD} = \left(L_{FD} - \frac{(L_{ad})^2}{L_d}\right)$$

To solve the previous system, we use the parameters of the simulated machine, which are:

- **Machine values (units of the international system).**

Stator:	Damper:	Field:
Rs = $2.9069\cdot 10^{-3}$	Rkd' = $1.1900\cdot 10^{-2}$	Rf' = $5.9013\cdot 10^{-4}$
Ll = $3.0892\cdot 10^{-4}$	Lkd' = $4.9076\cdot 10^{-4}$	Lfd' = $3.0712\cdot 10^{-4}$
Lmd = $3.2164\cdot 10^{-3}$	Rkq_1'= $2.0081\cdot 10^{-2}$	
Lmq = $9.7153\cdot 10^{-3}$	Lkq_1'= $1.0365\cdot 10^{-3}$	

Thus:

$$L_d = (L_{ad} + L_{al}) = (L_l + L_{ad}) = (L_l + L_{md}) = (3.0892\cdot 10^{-4}) + (3.2164\cdot 10^{-3}) = \mathbf{3.5253\cdot 10^{-3}}$$

$$L_{FF} = (L_{FD} + L_{Fl}) = (L_{ad} + L_{fd}) = (L_{md} + L_{fd}') = (3.2164\cdot 10^{-3}) + (3.0712\cdot 10^{-4}) = \mathbf{3.5235\cdot 10^{-3}}$$

$$L_{DD} = (L_{FD} + L_{Dl}) = (L_{ad} + L_{1d}) = (L_{md} + L_{kd}') = (3.2164\cdot 10^{-3}) + (4.9076\cdot 10^{-4}) = \mathbf{3.7072\cdot 10^{-3}}$$

$$L_{FD} = (L_{ad} + L_{Fdl}) = (L_{md}) = \mathbf{3.2164\cdot 10^{-3}}$$

And:

$$L_{sF} = \left(L_{FF} - \frac{(L_{ad})^2}{L_d}\right) = (3.5235\cdot 10^{-3}) - \frac{(3.2164\cdot 10^{-3})^2}{3.5253\cdot 10^{-3}} = \mathbf{5.6772\cdot 10^{-4}}$$

$$L_{sD} = \left(L_{DD} - \frac{(L_{ad})^2}{L_d}\right) = (3.7072\cdot 10^{-3}) - \frac{(3.2164\cdot 10^{-3})^2}{3.5253\cdot 10^{-3}} = \mathbf{7.5142\cdot 10^{-4}}$$

$$L_{MD} = \left(L_{FD} - \frac{(L_{ad})^2}{L_d}\right) = (3.2164\cdot 10^{-3}) - \frac{(3.2164\cdot 10^{-3})^2}{3.5253\cdot 10^{-3}} = \mathbf{2.6062\cdot 10^{-4}}$$

Resolving the previous system:

$$\begin{cases} Td'' + Td' = \dfrac{L_{sF}}{R_F} + \dfrac{L_{sd}}{R_D} = \dfrac{5.6772 \cdot 10^{-4}}{5.9013 \cdot 10^{-4}} + \dfrac{7.5142 \cdot 10^{-4}}{1.1900 \cdot 10^{-2}} = 1.0252 \\ Td'' \cdot Td' = \dfrac{L_{sF} \cdot L_{sD} - (L_{MD})^2}{R_F \cdot R_D} = \dfrac{(5.6772 \cdot 10^{-4} \cdot 7.5142 \cdot 10^{-4}) - (2.6062 \cdot 10^{-4})^2}{(5.9013 \cdot 10^{-4}) \cdot (1.1900 \cdot 10^{-2})} = 5.1075 \cdot 10^{-3} \end{cases}$$

Thus,

$$\begin{cases} Td'' + Td' = 1.0252 \\ Td'' \cdot Td' = 5.1075 \cdot 10^{-3} \end{cases} \Rightarrow \quad \begin{matrix} \mathbf{Td''} = \mathbf{0.0525} \\ \mathbf{Td'} = \mathbf{0.972} \end{matrix}$$

Now, we can compare the values:

Analytical method	Sudden three-phase short-circuit test.
Td"= 0.0525	**Td''= 0.04286**
Td' = 0.972	**Td'= 1.01**

Therefore, these results should be correct, due to their similarity.

Now we must repeat this process to the quadrature axis.

- **Quadrature axis.**

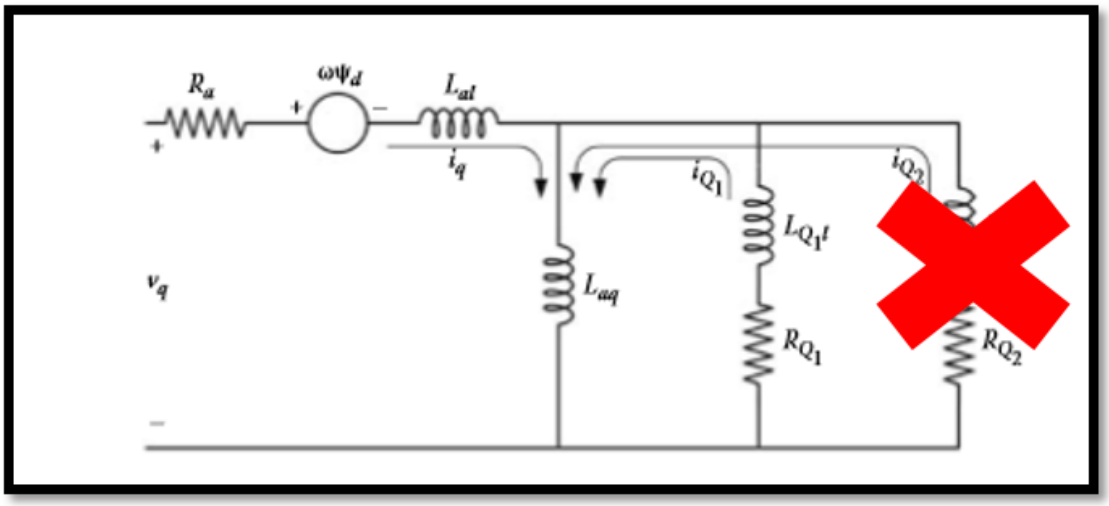


Figure 5.19 : Equivalent circuit quadrature axis.

In this case, we only have one constant, thus, we only need one equation:

The equation for this axis is the following:

$$Tq'' = \frac{1}{R_{Q1}}\left(L_{Q1Q1} - \frac{(L_{aq})^2}{L_q}\right)$$

Where:

$$L_q'' = L_{al} + \frac{L_{aq} \cdot L_{Q1\,l}}{L_{aq} + L_{Q1\,l}}$$

$$L_{Q1Q1} = L_{aq} + L_{Q1}$$

$$L_q = L_l + L_{aq}$$

To solve the previous equation, we use the parameters of the simulated machine,

- **Machine values: (units of the international system).**

Stator:	Damper:	Field:
Rs = $2.9069 \cdot 10^{-3}$	Rkd' = $1.1900 \cdot 10^{-2}$	Rf' = $5.9013 \cdot 10^{-4}$
Ll = $3.0892 \cdot 10^{-4}$	Lkd' = $4.9076 \cdot 10^{-4}$	Lfd' = $3.0712 \cdot 10^{-4}$

Lmd = 3.2164· 10^{-3} Rkq$_1$'= 2.0081· 10^{-2}

Lmq = 9.7153· 10^{-4} Lkq$_1$'= 1.0365· 10^{-3}

Thus:

$$L_{Q1Q1} = L_{aq} + L_{Q1} = L_{mq} + L_{kq1}{}' = (9.7153\cdot 10^{-4}) + (1.0365\cdot 10^{-3}) = \mathbf{2.0080\cdot 10^{-3}}$$

$$L_q = L_l + L_{aq} = L_l + L_{mq} = (3.0892\cdot 10^{-4}) + (9.7153\cdot 10^{-4}) = \mathbf{1.2805\cdot 10^{-3}}$$

And:

$$Tq'' = \frac{1}{R_{Q1}}\left(L_{Q1Q1} - \frac{(L_{aq})^2}{L_q}\right) = \frac{1}{2.0081\cdot 10^{-2}}\left(2.0080\cdot 10^{-3} - \frac{(9.7153\cdot 10^{-4})^2}{1.2805\cdot 10^{-3}}\right) = \mathbf{0.0632}$$

Now, we can compare the values of this axis:

Analytical method	Sudden three-phase short-circuit test.
Tq"= 0.06320	**Tq'' =0.0468**

7. Open circuit tests. (Constantes de tiempo circuito abierto)

7.1. Añadir ensayos antes de esto:

7.2. Checking the parameters obtained from the open circuit test.

Same as in the previous point, to make sure that the values obtained are correct, they will be calculated again by an analytical method. [1]

- **Direct axis.**

In this case, the equations are the following:

$$\begin{cases} Tdo'' + Tdo' = \dfrac{L_{FF}}{R_F} + \dfrac{L_{DD}}{R_D} \\ Tdo''\cdot Tdo' = \dfrac{L_{FF}.L_{DD} - (L_{FD})^2}{R_F\cdot R_D} \end{cases}$$

And remember that:

$$L_{FF} = (L_{FD} + L_{Fl}) = (L_{ad} + L_{fd}) = (L_{md} + L_{fd}') = (3.2164\cdot 10^{-3}) + (3.0712\cdot 10^{-4}) = \mathbf{3.5235\cdot 10^{-3}}$$

$$L_{DD} = (L_{FD} + L_{Dl}) = (L_{ad} + L_{1d}) = (L_{md} + L_{kd}') = (3.2164\cdot 10^{-3}) + (4.9076\cdot 10^{-4}) = \mathbf{3.7072\cdot 10^{-3}}$$

$$L_{FD} = (L_{ad} + L_{Fdl}) = (L_{md}) = \mathbf{3.2164\cdot 10^{-3}}$$

And,

$$R_F = Rf' = \mathbf{5.9013\cdot 10^{-4}}$$

$$R_D = \mathbf{2.0081\cdot 10^{-2}}$$

Resolving the previous system:

$$\begin{cases} Tdo'' + Tdo' = \frac{L_{FF}}{R_F} + \frac{L_{DD}}{R_D} = \frac{3.5235\cdot 10^{-3}}{5.9013\cdot 10^{-4}} + \frac{3.7072\cdot 10^{-3}}{2.0081\cdot 10^{-2}} = 6.2822 \\ Tdo'' \cdot Tdo' = \frac{L_{FF}.L_{DD} - (L_{FD})^2}{R_F\cdot R_D} = \frac{(3.5235\cdot 10^{-3})(3.7072\cdot 10^{-3}) - (3.2164\cdot 10^{-3})^2}{(5.9013\cdot 10^{-4})\cdot(2.0081\cdot 10^{-2})} = 0.3869 \end{cases}$$

Thus,

$$\begin{cases} Tdo'' + Tdo' = 6.2822 \Rightarrow \\ Tdo'' \cdot Tdo' = 0.3869 \end{cases} \qquad \begin{matrix} \mathbf{Tdo''= 0.0622} \\ \mathbf{Tdo' = 5.8953} \end{matrix}$$

Therefore:

Analytical method	Open circuit
Tdo"= 0.0622	**Td''= ?**
Tdo' = 5.8953	**Td'= ?**

- **Quadrature axis.**

In this case, the equation is the following:

$$Tqo'' = \frac{L_{Q1Q1}}{R_{Q1}}$$

Remember that:

$$L_{Q1Q1} = L_{aq} + L_{Q1} = L_{mq} + L_{kq1}' = (9.7153\cdot 10^{-4}) + (1.0365\cdot 10^{-3}) = \mathbf{2.0080\cdot 10^{-3}}$$

And:

$$R_{Q1} = Rkq_1' = \mathbf{2.0081\cdot 10^{-2}}$$

The result is:

$$\mathrm{Tqo''} = \frac{L_{Q1Q1}}{R_{Q1}} = \frac{2.0080 \cdot 10^{-3}}{2.0081 \cdot 10^{-2}} = \mathbf{0.099}$$

Thus,

Analytical method	Open circuit
Tqo''= $\mathbf{0.099}$	**Tqo'' = ?**

7.3. Checking the time constants obtained from the open circuit test with the equations of the standard.

The equations of IEC 60034-4 are the followings:

- **Direct axis:**

$$\begin{cases} L'_d = Ld(0) * \dfrac{T'_d}{T'_{do}} \\ L''_d = Ld(0) * \dfrac{T'_d * T''_d}{T'_{do} * T''_{do}} \end{cases}$$

Where,

$$L'_d = \frac{Xd'}{2\pi \cdot 60}$$

$$L_d\,(0) = \frac{Xd}{2\pi \cdot 60}$$

$$L''_d = \frac{Xd''}{2\pi \cdot 60}$$

The values of Td',Td'' are unknown in the Simlunk machine, thus, we will solve the system with the parameters obtained from sudden three phase short circuit test.

- Values

Xd'' = 0.1771	Td'' = 0.042
Xd' = 0.2174	Td' = 1.01

Xd = 1.2810

Thus,

$$L'_d = \frac{Xd'}{2\pi \cdot 60} = \frac{0.2174}{2\pi \cdot 60} = 5.9443 \cdot 10^{-4}$$

$$L_d(0) = \frac{Xd}{2\pi \cdot 60} = \frac{1.2810}{2\pi \cdot 60} = 3.39 \cdot 10^{-3}$$

$$L''_d = \frac{Xd''}{2\pi \cdot 60} = \frac{0.1771}{2\pi \cdot 60} = 4.69 \cdot 10^{-4}$$

And:

$$\begin{cases} 5.9443 \cdot 10^{-4} = L'_d = Ld(0) * \frac{T'_d}{T'_{do}} = 3.39 \cdot 10^{-3} \cdot \frac{1.01}{T'_{do}} \\ 4.69 \cdot 10^{-4} = L''_d = Ld(0) * \frac{T'_d * T''_d}{T'_{do} * T''_{do}} = 3.39 \cdot 10^{-3} \cdot \frac{1.01 \cdot 0.042}{T'_{do} * T''_{do}} \end{cases}$$

Resolving the previous system:

$$\mathbf{T'_{do} = 5.9443}$$
$$\mathbf{T''_{do} = 0.0526}$$

Therefore,

Analytical method	Open circuit
Tdo"= 0.0526	**Tdo''= ?**
Tdo' = 5.9443	**Td'o = ?**

- **Quadrature axis.**

The equations of IEC 60034-4 is:

$$L''_q = Lq(0) * \frac{T''_q}{T''_{qo}}$$

Where,

$$L''_q = \frac{Xq''}{2\pi \cdot 60}$$

$$L_q\,(0) = \frac{Xq}{2\pi \cdot 60}$$

In this case, the value of Tq' is know in the Simlunk machine, thus, we will solve the system with the machine parameters:

- Values

Xq'' = 0.243 Tq'' = 0.0513

Xq = 0.474

Thus,

$$L''_q = \frac{Xq''}{2\pi \cdot 60} = \frac{0.243}{2\pi \cdot 60} = 6.4458 \cdot 10^{-3}$$

$$L_q\,(0) = \frac{Xq}{2\pi \cdot 60} = \frac{0.474}{2\pi \cdot 60} = 1.2573 \cdot 10^{-3}$$

And,

$$6.4458 \cdot 10^{-3} = L''_q = Lq(0) * \frac{T''_q}{T''_{qo}} = 1.2573 \cdot 10^{-3} * \frac{0.0513}{T''_{qo}}$$

Therefore,

$$\boldsymbol{T''_{qo} = 0.1000}$$

Analytical method

Tqo”= 0.1000

Open circuit test

Tqo’’= ?

In summary, we have obtained this values of time constants:

- Short circuit.

	Simulation	Equations from ref[1]	Simulink machine values	IEC 60034-4 equations (with test values)
Td’	1.01	0.972	Unknown	1.0100
Td’’	0.042	0.0525	Unknown	0.0580
Tq’	0.0468	0.0632	0.0513	-

- Open circuit.

	Simulation	Equations from ref[1]	Simulink machine values	IEC 60034-4 equations (with test values)
Tdo’	?	5.895	4.49	5.9443
Tdo’’	?	0.0622	0.06	0.0526
Tqo’	?	0.099	Unknown	0.1000

7. Standstill frequency response.

To perform this test, voltage at various frequencies is applied to a pair of line terminals of the armature winding. The machine is at standstill. The armature winding is supplied from a single-phase, variable frequency power amplifier. Several tests are made in order to obtain the corresponding transfer functions.

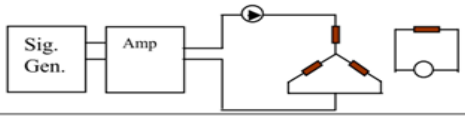
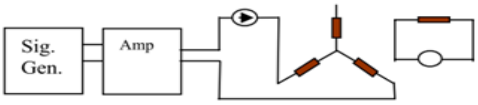
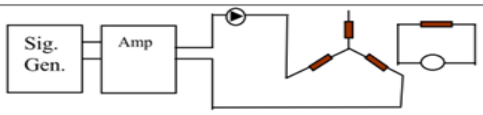
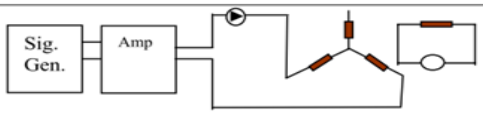
No.	Measurement	Test Diagram	Measured Value	Relationships
1	q-Axis operational Impedance $Z_q(s)$		U_{stator} I_{stator} $U_{rotor(about\ 0)}$	$Z_q(s) = -\frac{\Delta e_q(s)}{\Delta i_q(s)}\Big\|_{\Delta e_{fd}=0}$
2	d-Axis operational Impedance $Z_d(s)$		U_{stator} I_{stator} $U_{rotor(max)}$	$Z_d(s) = -\frac{\Delta e_d(s)}{\Delta i_d(s)}\Big\|_{\Delta e_{fd}=0}$
3	Standstill armature to field transfer function $sG(s)$		U_{stator} I_{stator} I_{rotor}	$sG(s) = -\frac{\Delta i_{fd}(s)}{\Delta i_d(s)}\Big\|_{\Delta e_{fd}=0}$
4	Standstill armature to field transfer impedance Z_{afo}		U_{rotor} I_{stator} $I_{rotor(about-0)}$	$_{fo}(s) = -\frac{\Delta e_{fd}(s)}{\Delta i_d(s)}\Big\|_{\Delta i_{fd}=0}$

Table 5.6. Standstill frequency response test

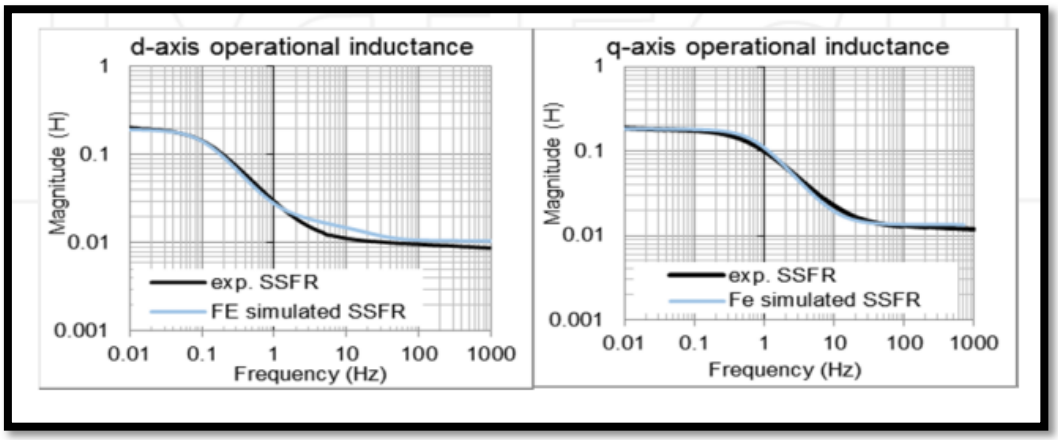


Figure 5.20. Standstill frequency test data

Once all the transfer functions are obtained, the model parameters for d- and q-axes equivalent circuit are determined by curve-fitting technique.

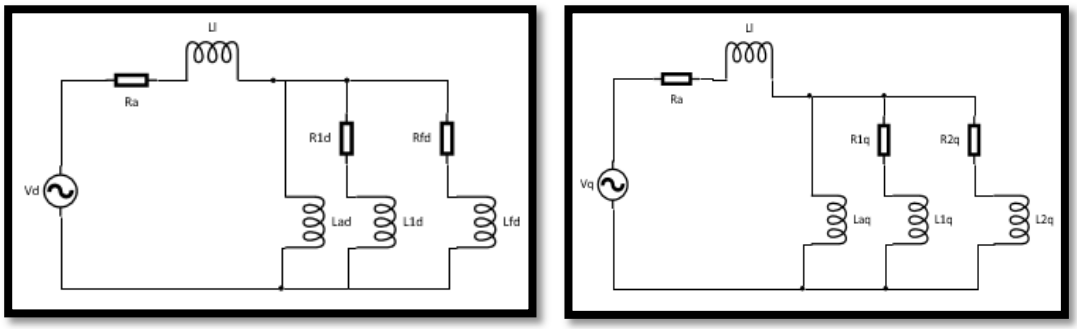


Figure 5.21. d and q-axes equivalent circuits

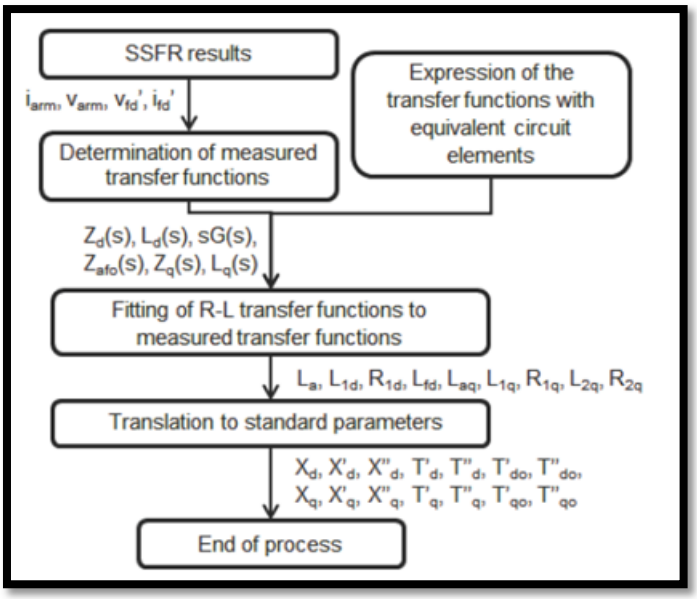


Figure 5.22. Parameter identification flowchart

Positioning the rotor for direct axis test:

Drive the amplifier with approximately 100Hz, and measure the induced voltage in the field winding. Turn the rotor until the induced voltage in the field will be nulled. When this condition is accomplished the magnetic axis of the field winding is aligned with the series connection of phase *a* and *b*. Now the rotor is positioned for the direct axis test.

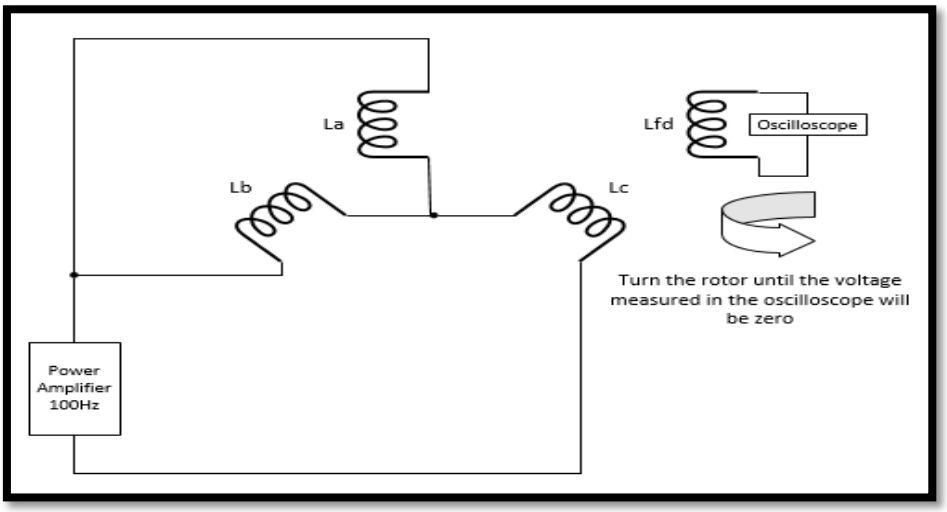


Figure 5.23. Wiring of armature and field winding for rotor positioning for direct axis test

With the proper orientation of direct axis the following tests will be done, according to the standard IEEE-115A-1987.[29]

- Test 1: Direct-axis impedance transfer function.
- Test 2: Standstill armature to field winding transfer impedance.
- Test 3: Standstill armature to field winding transfer function.

Direct axis impedance transfer function test:

Supply the armature phase *a* and *b*, with phase *c* open circuit with a power amplifier of variable frequency. The field circuit will be short circuited with a non-inductive metering shunt. Connect the voltage and current measurements of the stator winding to the measurement instrument. Perform the measurement over a bandwidth from 0,001Hz to 1000Hz.

[29] IEEE Standard Procedures for Obtaining Synchronous Machine Parameters by Standstill Frequency Response Testing. IEEE Std 115A-1987

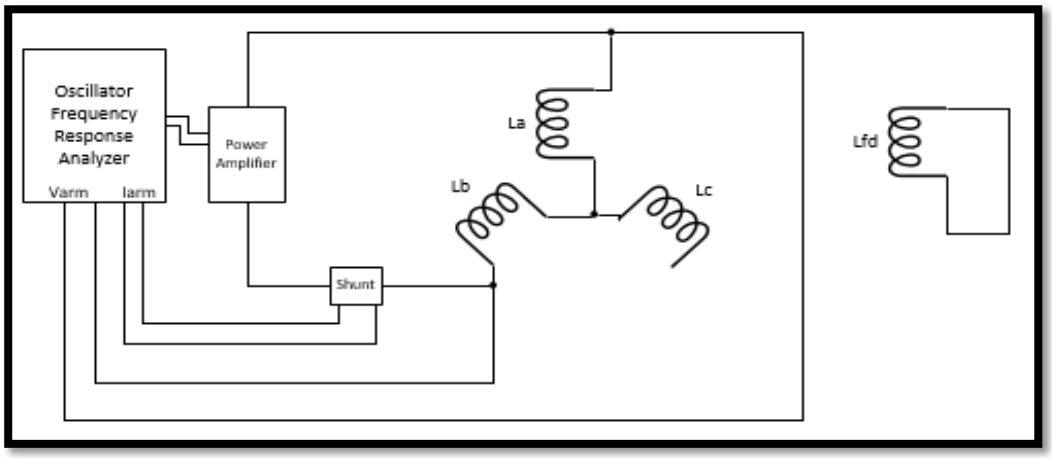


Figure 5.24. Direct axis impedance transfer function test

The impedance transfer function that can be obtained is:

$$Z_d(s) = -\frac{\Delta e_d(s)}{\Delta i_d(s)}\Big|_{\Delta e_{fd}=0}$$

Standstill armature to field winding transfer impedance test:

In this test the transfer function to be sampled and evaluated will be the relation between the voltage field winding and the current in the direct axis.

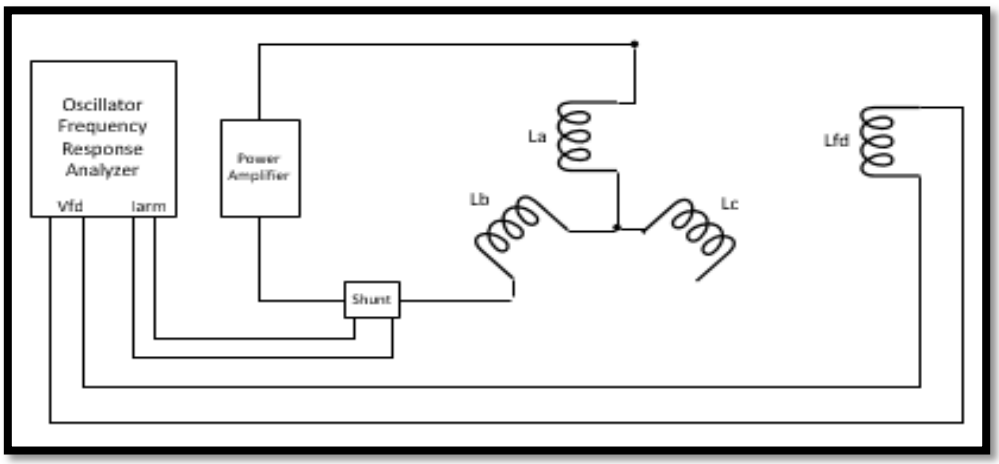


Figure 5.25. Armature to field winding transfer impedance test

The impedance transfer function that can be obtained is:

$$Z_{afo}(s) = -\frac{\Delta e_{fd}(s)}{\Delta i_d(s)}\Big|_{\Delta i_{fd}=0}$$

Standstill armature to field winding transfer function test:

In this test the transfer function to be sampled and evaluated will be the relation between the current field winding and the current in the direct axis.

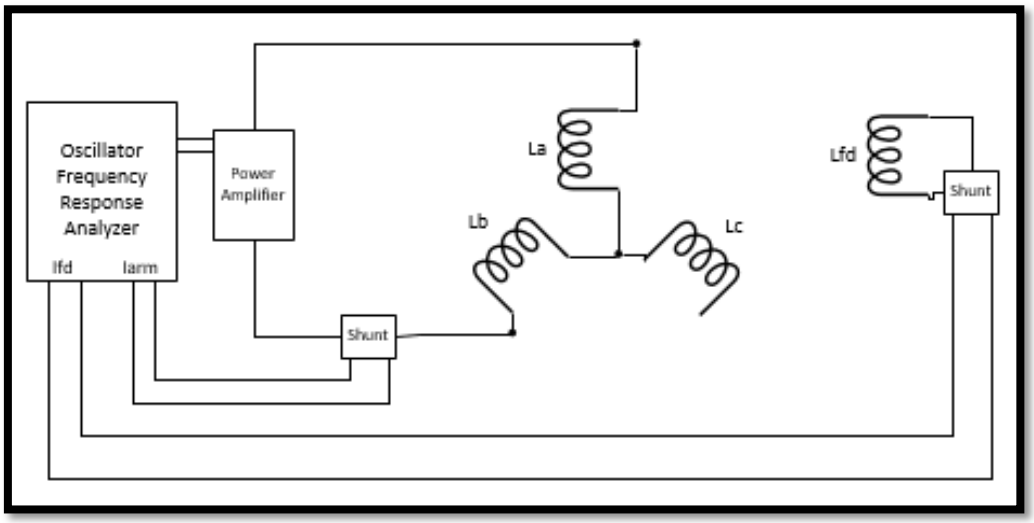


Figure 5.26. Armature to field winding transfer function test

The transfer function that can be obtained is:

$$sG(s) = \frac{\Delta i_{fd}(s)}{\Delta i_d(s)}\Big|_{\Delta e_{fd}=0}$$

Positioning the rotor for quadrature axis test:

Once the essays in the d-Axis have been done, then we will follow for the essay in the q-Axis. First, the rotor must be positioned for the quadrature axis test. To done it, connect the power amplifier across phase *a* and *b*, with phase *c* in open circuit. Remove the field current metering shunt and set the oscillator frequency to 100Hz.

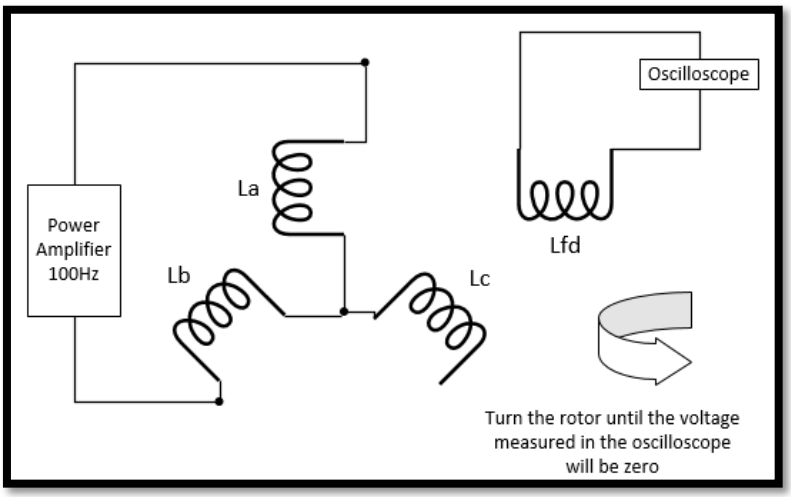


Figure 5.27. Wiring of armature and field winding for rotor positioning for quadrature axis test

Measure the induced voltage in the field winding by the oscilloscope. Turn the rotor until the induced voltage in the field winding will be nulled. Now the rotor is positioned for the quadrature axis test.
Quadrature-axis impedance transfer function test will be done, according to the standard IEEE-115A-1987.

Quadrature axis impedance transfer function test:
Supply the armature phase *a* and *b*, with phase *c* open circuit with a power amplifier of variable frequency. The field circuit will be short circuited with a non-inductive metering shunt. Connect the voltage and current measurements of the stator winding to the measurement instrument. Perform the measurement over a bandwidth from 0,001Hz to 1000Hz.

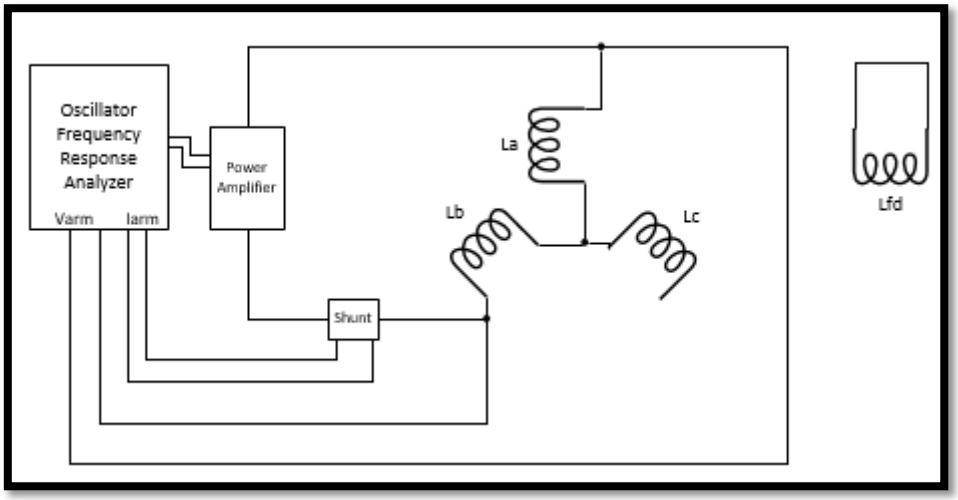


Figure 5.28. Quadrature axis impedance transfer function test

The impedance transfer function that can be obtained is:

$$Z_q(s) = -\frac{\Delta e_q(s)}{\Delta i_q(s)}\Big|_{\Delta e_{fd}=0}$$

Example
Rated values of the synchronous machine:

$$Sn = 192{,}3 \text{ MVA}$$
$$Vn = 18 \text{ kV}$$
$$f = 60\ Hz$$

Base value:

$$Zb = \frac{Vn^2}{Sn} = 1{,}6849\ \Omega$$

$$\text{Lb} = \frac{Zb}{\omega} = 0{,}0045 \text{ H}$$

Leakage inductance Ll, will be a value supplied by the manufacturer or obtained from the zero power-factor test.

$$L_l = 0{,}795\text{mH}$$

The armature impedance measurement for each sampling frequency will be obtained from the direct axis impedance transfer function test:

$$Z_{arm\_d} = \frac{V_{arm\_d}}{I_{arm\_d}}$$

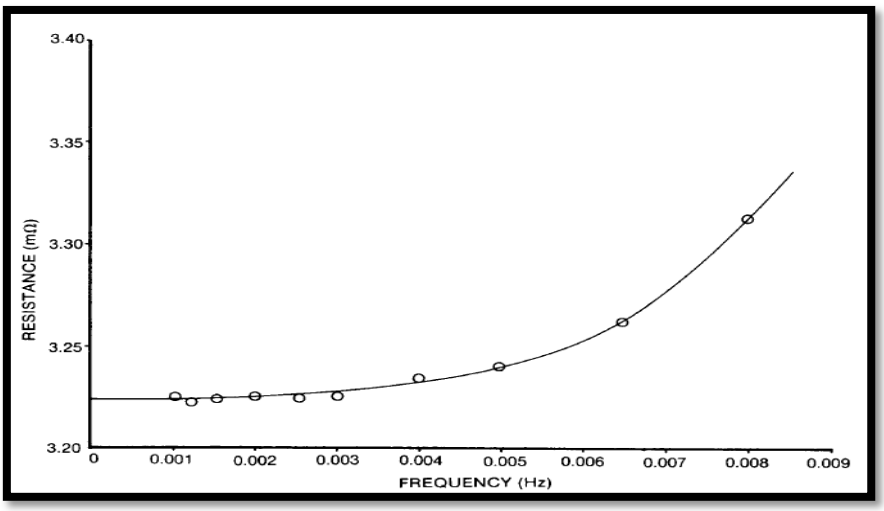


Figure 5.29. Armature resistance obtained from the test

The resistance of the armature (R_arm) is obtained from the real component of the impedance Zarm_d(s) extrapolated to zero frequency.

$$R_{arm} = 0{,}00323\Omega$$

The resistance of one phase of the armature winding extrapolated to zero frequency is:

$$R_a = \frac{R_{arm}}{2} = 0{,}0016\Omega$$

The transfer function impedance in direct axis will be calculated as follows:

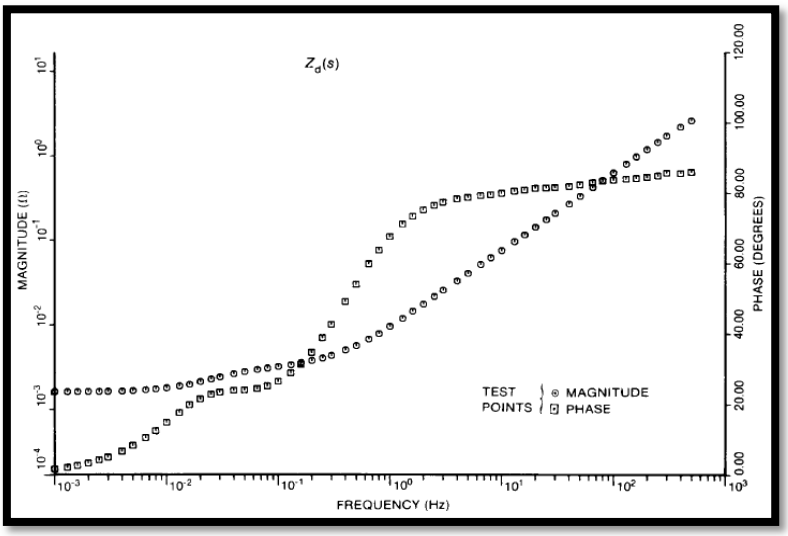


Figure 5.30. Module and argument of direct impedance Zd(s)

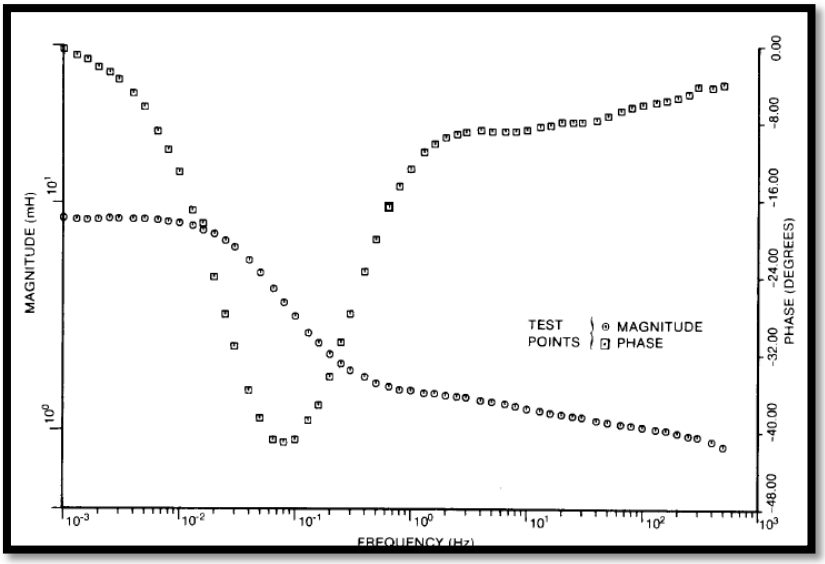


Figure 5.31. Module and argument of the d-axis operational inductance

$$Z_d = \frac{Z_{arm}}{2}$$

The operational inductance in d-axis is:

$$L_d(s) = \frac{Z_d(s) - R_a}{s}$$

The d-axis inductance at zero frequency Ld(0) can be calculated extrapolating Ld(s) to zero frequency.

$$L_d(0) = 0{,}00795H$$

The direct-axis armature to rotor mutual inductance is obtained from the difference between the d-axis inductance at zero frequency and the leakage inductance:

$$L_{ad} = L_d(0) - L_l = 0{,}00795 - 0{,}000795 = 0{,}007155H$$

The data obtained from the armature to field transfer impedance function test are:

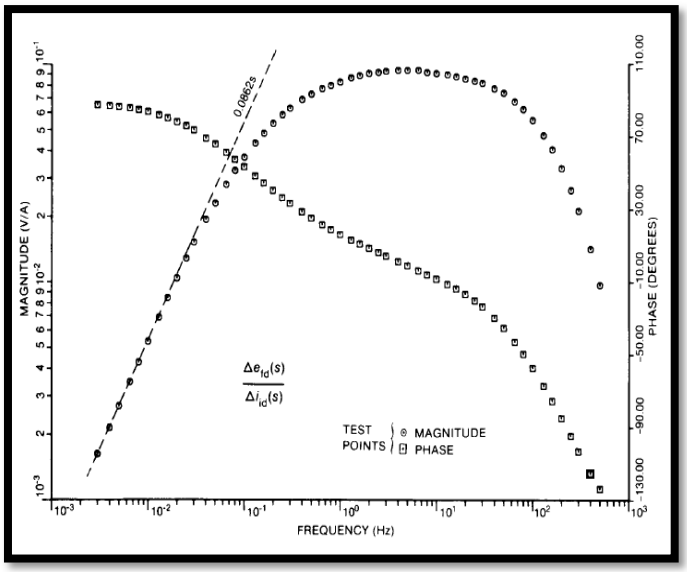


Figure 5.32. Module and argument of armature and field winding transfer function impedance efd(s)/i_d (s)

The slope of the armature and field winding transfer function at zero frequency is

$$\frac{\Delta e_{fd}}{\Delta i_d}(0) = 0{,}0862$$

The armature to field transfer function is obtained from the third test:

$$\frac{i_{fd}(s)}{i_d(s)}$$

The results from this test are:

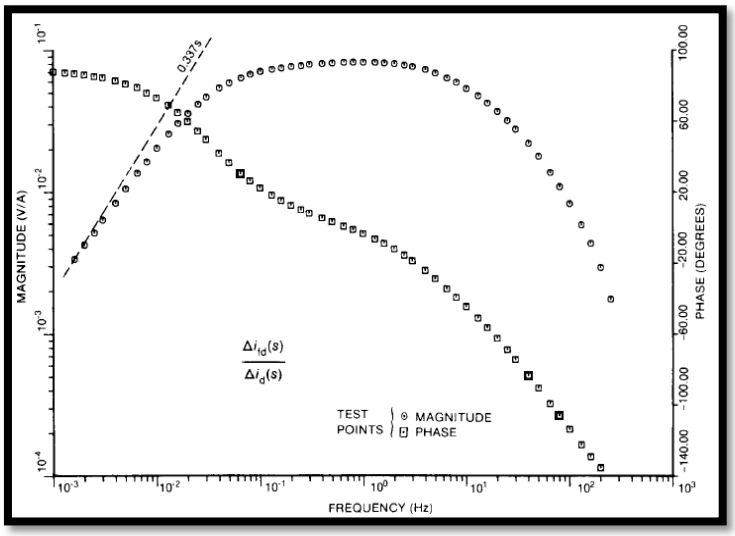


Figure 5.33. Module and argument of armature to field transfer function i_fd (s)/i_d (s)

The slope of the armature to field transfer function at zero frequency is

$$\frac{\Delta i_{fd}}{\Delta i_d}(0) = 0{,}337s$$

The proposed d-axis equivalent circuit is represented in Fig. 31 [30].

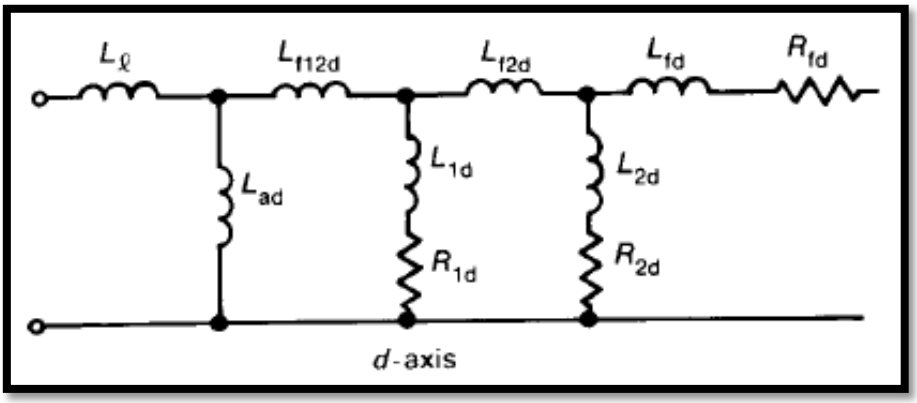


Figure 5.34. d-axis equivalent circuit

The field to armature turn ratio is:

$$\frac{N_{fd}}{N_a} = \frac{\frac{\Delta e_{fd}}{\Delta i_d}(0)}{L_{ad}} = 12{,}05$$

The field resistance referred to the armature winding is:

[30] IEEE Standard Procedures for Obtaining Synchronous Machine Parameters by Standstill Frequency Response Testing. IEEE Std 115A-1987

$$R_{fd} = \frac{L_{ad}}{\frac{\Delta i_{fd}}{\Delta i_d}(0) * \frac{2}{3} * \frac{N_{fd}}{N_a}} = \frac{0{,}007155}{0{,}337 * \frac{2}{3} * 12{,}05} = 0{,}002643\Omega$$

The rest of the model parameters for d-axis equivalent circuit (Lf12d, L1d, R1d, Lf2d, L2d, R2d, Lfd) are determined by curve-fitting technique from the operational Ld(s). The obtained values are:

$$L_{f12d} = 0{,}267\ H$$
$$L_{f2d} = 0\ H$$
$$L_{1d} = 0\ H$$

$$R_{1d} = 0{,}0263\ \Omega$$
$$L_{2d} = 0{,}00282\ H$$
$$R_{2d} = 0{,}00657\ \Omega$$
$$L_{fd} = 0{,}726\ mH$$

The armature impedance measurement for each sampling frequency will be obtained from the quadrature axis impedance transfer function test:

$$Z_{arm\_q} = \frac{V_{arm\_q}}{I_{arm\_q}}$$

The quadrature-axis operational impedance is:

$$Z_q = \frac{Z_{arm\_q}}{2}$$

The quadrature operational inductance is:

$$L_q(s) = \frac{Z_q(s) - R_a}{s}$$

The proposed q-axis equivalent circuit is represented in Fig. 34 [31].

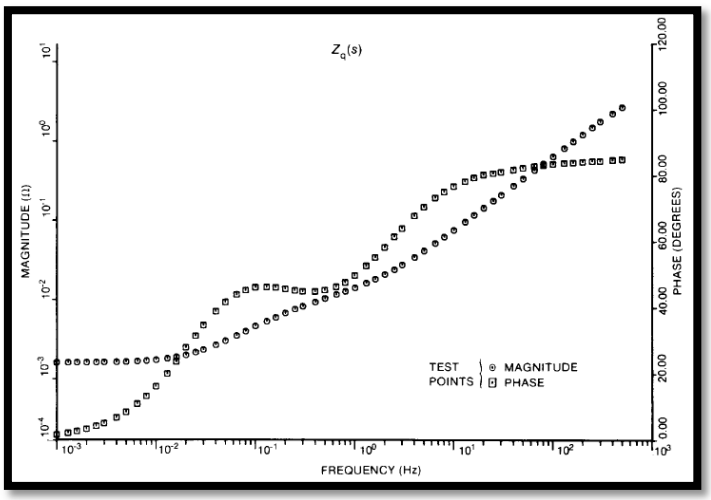


Figure 5.35. Module and argument of quadrature impedance Zq(s)

[31] IEEE Standard Procedures for Obtaining Synchronous Machine Parameters by Standstill Frequency Response Testing. IEEE Std 115A-1987

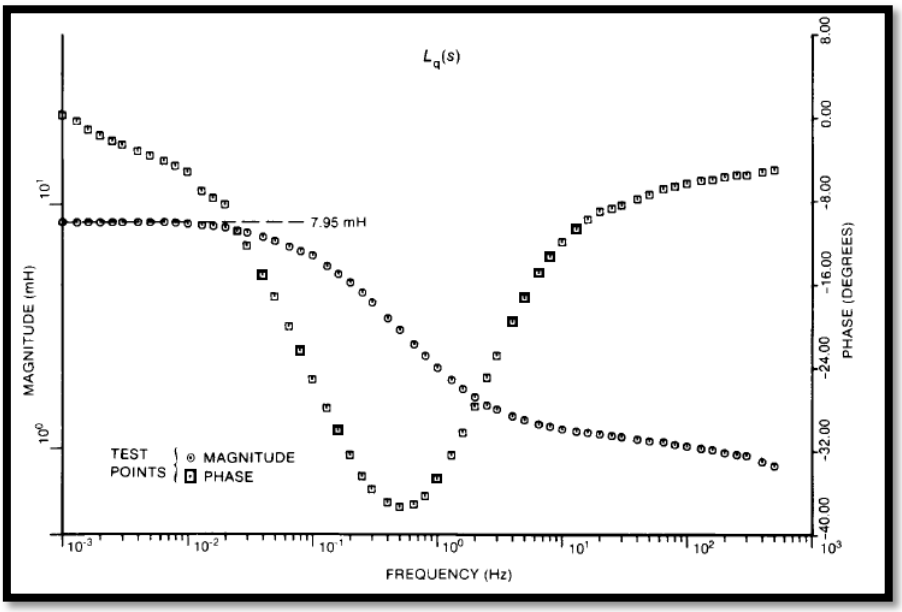


Figure 5.36. Module and argument of the q-axis operational inductance

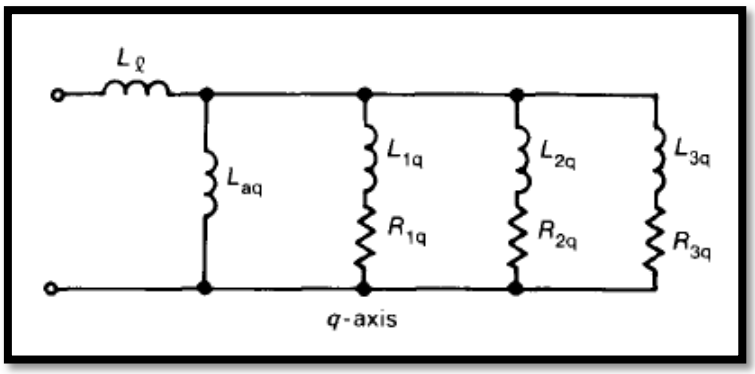


Figure 5.37. q-axis equivalent circuit

The q-axis inductance at zero frequency Lq(0) can be calculated extrapolating Lq(s) to zero frequency.

$$L_q(0) = 0{,}00795H$$

The quadrature-axis armature to rotor mutual inductance is obtained from the difference between the q-axis inductance at zero frequency and the leakage inductance:

$$L_{aq} = L_q(0) - L_l = 0{,}00795 - 0{,}000795 = 0{,}007155H$$

The rest of the model parameters for q-axis equivalent circuit (L1q, R1q, L2q, R2q, L3q, R3q) are determined by curve-fitting technique from the operational Lq(s). The obtained values are:

$$L_{1q} = 6{,}045\ mH$$
$$R_{1q} = 0{,}01355\ \Omega$$
$$L_{2q} = 0{,}735\ mH$$
$$R_{2q} = 0{,}01525\ \Omega$$

$$L_{3q} = 0{,}453\ mH$$
$$R_{3q} = 0{,}1578\ mH$$

Usually, it's not possible to have enough data to work with the previous equivalent circuits and simplified ones are assumed, as the represented in Figs. 35 and 36.

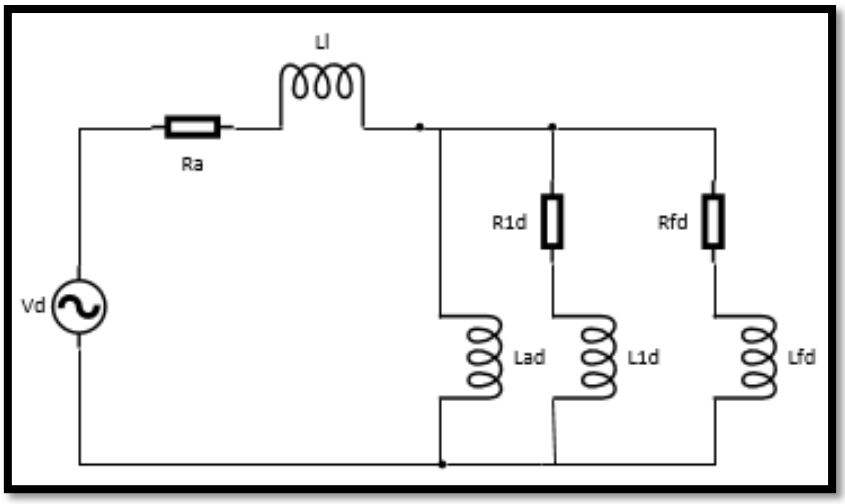


Figure 5.38. Assumed d-axis equivalent circuit

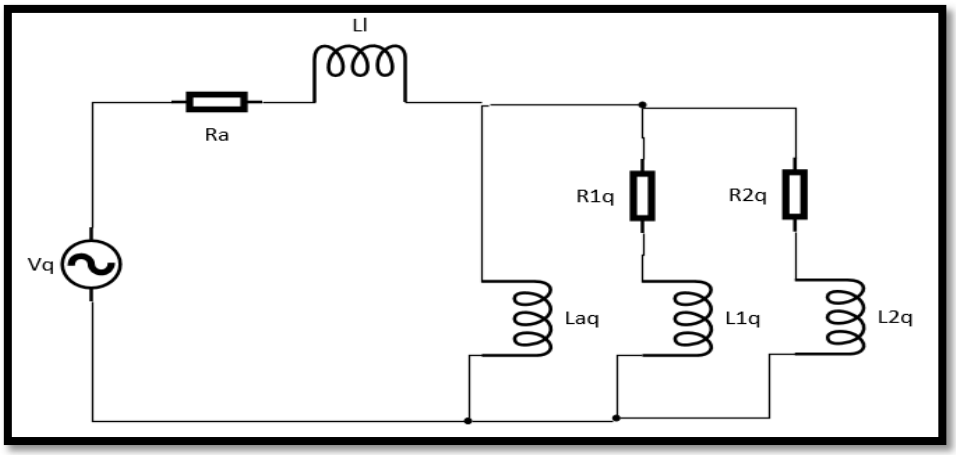


Figure 5.39. Assumed q-axis equivalent circuit

Unknown parameters could be either the standard constants appearing in the operational form:

$$L_d(s) = \frac{L_d(0) * (1 + s * T'd) * (1 + s * T''d)}{(1 + s * T'do) * (1 + s * T''do)}$$

Where:

- Ld(0) is the direct axis inductance in the low frequency bandwidth.
- T'do is the open circuit transient time constant in d-axis.
- T''do is the open circuit sub-transient time constant in d-axis.
- T'd is the short circuit transient time constant in d-axis.

- T''d is the short circuit sub-transient time constant in d-axis.

$$L_q(s) = \frac{L_q(0) * (1 + s * T'q) * (1 + s * T''q)}{(1 + s * T'qo) * (1 + s * T''qo)}$$

Where:

- Lq(0) is the quadrature axis inductance in the low frequency bandwidth.
- T'qo is the open circuit transient time constant in q-axis.
- T''qo is the open circuit sub-transient time constant in q-axis.
- T'q is the short circuit transient time constant in q-axis.
- T''q is the short circuit sub-transient time constant in q-axis.

Previous standard parameters are determined by curve-fitting technique from the operational Ld(s) and Lq(s). For example, for Ld the fitting curve is shown in the next figures:

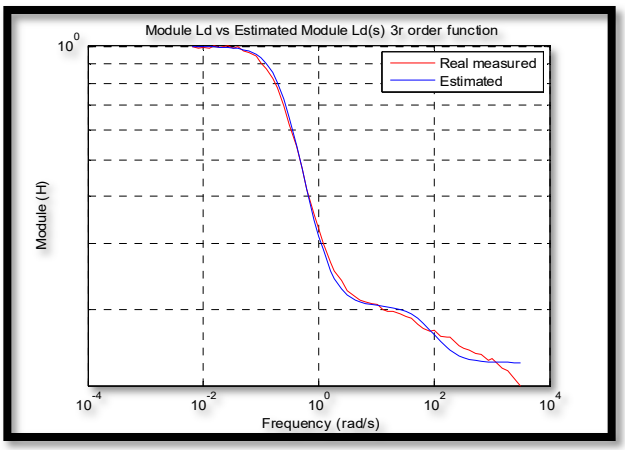


Figure 5.40: Module Ld and module Ld(s)

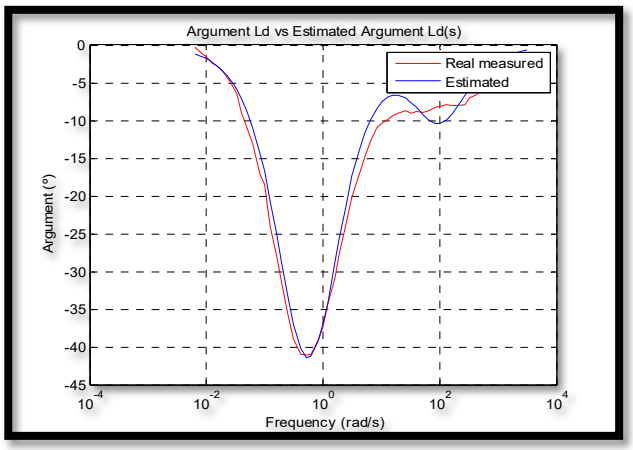


Figure 5.41: Argument Ld and argument Ld(s)

The obtained values for d-axis are:

$$T'_{do} = 3{,}8907\ s$$
$$T''_{do} = 0{,}0156\ s$$
$$T'_d = 0{,}8018\ s$$
$$T''_d = 0{,}011\ s$$

Transient and sub-transient inductances in d-axis are:

$$L'_d = Ld(0) * \frac{T'_d}{T'_{do}} = 0{,}0017H$$

$$L''_d = Ld(0) * \frac{T'_d * T''_d}{T'_{do} * T''_{do}} = 0{,}0012H$$

The obtained values for q-axis are:

$$T'_{qo} = 1{,}83\ s$$
$$T''_{qo} = 0{,}3251\ s$$
$$T'_{q} = 0{,}999\ s$$
$$T''_{q} = 0{,}094\ s$$

Transient and sub-transient inductances in q-axis are:

$$L'_q = Lq(0) * \frac{T'_q}{T'_{qo}} = 0{,}0014H$$
$$L''_q = Lq(0) * \frac{T'_q * T''_q}{T'_{qo} * T''_{qo}} = 0{,}0046H$$

R-L parameters for the d-axis equivalent circuit are calculated from the standard parameters. Taking into account that the known parameters are Ll and Lad, then the remaining parameters (R1d, L1d, Rfd, Lfd) are calculated by using the following equations:

$$L_d(s) = L_d(0) * \frac{1 + (T_{4d} + T_{5d})s + (T_{4d} + T_{6d})s^2}{1 + (T_{1d} + T_{2d})s + (T_{1d} + T_{3d})s^2}$$

Where:

$$T_{1d} = \frac{L_{ad} + L_{fd}}{R_{fd}}$$
$$T_{2d} = \frac{L_{ad} + L_{1d}}{R_{1d}}$$
$$T_{3d} = \frac{1}{R_{1d}} \cdot \left(L_{1d} + \frac{L_{ad} \cdot L_{fd}}{L_{ad} + L_{fd}}\right)$$
$$T_{4d} = \frac{1}{R_{fd}} \cdot \left(L_{fd} + \frac{L_{ad} * L_l}{L_{ad} + L_l}\right)$$
$$T_{5d} = \frac{1}{R_{1d}} \cdot \left(L_{1d} + \frac{L_{ad} * L_l}{L_{ad} + L_l}\right)$$
$$T_{6d} = \frac{1}{R_{1d}} \cdot \left(L_{1d} + \frac{L_{fd} \cdot L_{ad} \cdot L_l}{L_{ad} \cdot L_l + L_{ad} \cdot L_{fd} + L_{fd} \cdot L_l}\right)$$

$$T'_{do} = \frac{2 \cdot T_{1d} \cdot T_{3d}}{T_{1d} + T_{2d} - \sqrt{(T_{1d} + T_{2d})^2 - 4 \cdot T_{1d} \cdot T_{3d}}}$$

$$T''_{do} = \frac{2 \cdot T_{1d} \cdot T_{3d}}{T_{1d} + T_{2d} + \sqrt{(T_{1d} + T_{2d})^2 - 4 \cdot T_{1d} \cdot T_{3d}}}$$

$$L'_d = L_d \cdot \frac{T_{4d} \cdot T_{6d}}{T_{1d} \cdot T_{3d}} \cdot \frac{T_{1d} + T_{2d} - \sqrt{(T_{1d} + T_{2d})^2 - 4 \cdot T_{1d} \cdot T_{3d}}}{T_{4d} + T_{5d} - \sqrt{(T_{4d} + T_{5d})^2 - 4 \cdot T_{4d} \cdot T_{6d}}}$$

$$L''_d = L_d \cdot \frac{T_{4d} \cdot T_{6d}}{T_{1d} \cdot T_{3d}}$$

The parameters obtained are:

$$R_{1d} = 0.0934\ \Omega$$
$$L_{1d} = 0{,}617\ mH$$
$$R_{fd} = 0{,}0021\ \Omega$$
$$L_{fd} = 0{,}985\ mH$$

R-L parameters for the q-axis equivalent circuit are calculated from the standard parameters. Taking into account that the known parameters are Ll and Laq, then the remaining parameters (R1q, L1q, R2q, L2q) are calculated by using the following equations:

$$L_q(s) = L_q(0) * \frac{1 + (T_{4q} + T_{5q})s + (T_{4q} + T_{6q})s^2}{1 + (T_{1q} + T_{2q})s + (T_{1q} + T_{3q})s^2}$$

Where:

$$T_{1q} = \frac{L_{aq} + L_{2q}}{R_{2q}}$$

$$T_{2q} = \frac{L_{aq} + L_{1q}}{R_{1q}}$$

$$T_{3q} = \frac{1}{R_{1q}} \cdot \left(L_{1q} + \frac{L_{aq} \cdot L_{2q}}{L_{aq} + L_{2q}} \right)$$

$$T_{4q} = \frac{1}{R_{2q}} \cdot \left(L_{2q} + \frac{L_{aq} \cdot L_l}{L_{aq} + L_l} \right)$$

$$T_{5q} = \frac{1}{R_{1q}} \cdot \left(L_{1q} + \frac{L_{aq} \cdot L_l}{L_{aq} + L_l} \right)$$

$$T_{6q} = \frac{1}{R_{1q}} \cdot \left(L_{1q} + \frac{L_{aq} \cdot L_{2q} \cdot L_l}{L_{aq} \cdot L_l + L_{aq} \cdot L_{2q} + L_{2q} \cdot L_l} \right)$$

$$T'_{qo} = \frac{2 \cdot T_{1q} \cdot T_{3q}}{T_{1q} + T_{2q} - \sqrt{(T_{1q} + T_{2q})^2 - 4 \cdot T_{1q} \cdot T_{3q}}}$$

$$T''_{qo} = \frac{2 \cdot T_{1q} \cdot T_{3q}}{T_{1q} + T_{2q} + \sqrt{(T_{1q} + T_{2q})^2 - 4 \cdot T_{1q} \cdot T_{3q}}}$$

$$L'_q = L_q \cdot \frac{T_{4q} \cdot T_{6q}}{T_{1q} \cdot T_{3q}} \cdot \frac{T_{1q} + T_{2q} - \sqrt{(T_{1q} + T_{2q})^2 - 4 \cdot T_{1q} \cdot T_{3q}}}{T_{4q} + T_{5q} - \sqrt{(T_{4q} + T_{5q})^2 - 4 \cdot T_{4q} \cdot T_{6q}}}$$

$$L''_q = L_q \cdot \frac{T_{4q} \cdot T_{6q}}{T_{1q} \cdot T_{3q}}$$

The parameters obtained are:

$$L_{1q} = 0{,}522\ mH$$
$$R_{1q} = 0{,}0126\ \Omega$$

$$L_{2q} = 10{,}9\ mH$$
$$R_{2q} = 0.0116\ \Omega$$

- **Matlab code to find the operational transfer function for a 2nd order circuit and their parameters:**

Standard parameters are determined by curve-fitting technique from the operational Ld(s):

```
% Parameter Ld(s)
load 'mod_arg_Ld1.txt'
ld=mod_arg_Ld1;
for k=1:1:length(ld);
   h(k)=(ld(k,2)*cos(ld(k,3)/180*pi)+i*ld(k,2)*sin(ld(k,3)/180*pi))/ld(1,2);
   w(k)=2*pi*ld(k,1);
end

for k=1:1:length(w)/2-10
   Wt(k)=1*0.4;
end
for k=length(w)/2-10+1:1:length(w)/2-10+20
      Wt(k)=1*8;
end
for k=length(w)/2-10+21:1:length(w)
   Wt(k)=1*4;
end

[b,a]=invfreqs(h,w,2,2,Wt,200,1e-7,'trace');
[h2,w2]=freqs(b,a,w);

figure(1)
loglog(w/(2*pi),abs(h),'r',w2/(2*pi),abs(h2),'b')
figure(2)
semilogx(w/(2*pi),angle(h)*180/pi,'r',w2/(2*pi),angle(h2)*180/pi,'b')

sol1=roots(a);     %%solve the poles a
Tdop=-1/sol1(2);
Tdopp=-1/sol1(1);
sol2=roots(b);     %%solve the zeros b
Tdp=-1/sol2(2);
Tdpp=-1/sol2(1);
```

```
Ldp=ld(1,2)*Tdp/Tdop;
Ldpp=ld(1,2)*Tdp*Tdpp/(Tdop*Tdopp);
```

R-L parameters for the d-axis equivalent circuit are calculated from the standard parameters:

```
% Parameter data
xd = 7.950e-3 / lb;
xl = 0.795e-3 / lb;
l12d=0;
%
% Data from Ld fitting
taud0_p = 3.8907 * wb;
taud0_pp = 0.0156 * wb;
taud_p = 0.8018 * wb;
taud_pp = 0.0110 * wb;
%
% Calculations
xd_pp = xd * taud_p * taud_pp / (taud0_p * taud0_pp);
xd_p  = xd * taud_p / taud0_p;
lad = xd - xl;
params = [xd_pp,xd_p,xd,xl,taud0_pp,taud0_p,lad,l12d];
options = optimset('MaxFunEvals',2e4,'MaxIter',2000,'TolFun',1e-8,'TolX',1e-7);
%
% Initializations
lfd0 = (xd_p-xl)*lad / (lad-xd_p+xl);
l1d0 = (xd_pp-xl)*lad*lfd0 / ((lad*lfd0)-((xd_pp-xl)*(lad+lfd0)));
r1d0 = (l1d0 + ((lad*lfd0)/(lad+lfd0))) / taud0_pp;
rfd0 = (lad+lfd0) / taud0_p; %%%
%
x0 = [lfd0; l1d0; r1d0; rfd0];
[x,fval,exitflag] = fsolve('fsol_ejed_r',x0,options,params);
lfd = x(1);    l1d = x(2);    r1d = x(3);  rfd = x(4);
if exitflag == 0,
   h = warndlg('Maximum iteration number has been exceeded','Warning: axe d results');
end


function f = fsol_ejed_r(x,params)
```

```
 xd_pp = params(1);      xd_p = params(2);      xd = params(3);
xl = params(4);         taud0_pp = params(5);  taud0_p = params(6);
lad = params(7);        l12d = params(8);

lfd = x(1);     l1d = x(2);     r1d = x(3);    rfd = x(4);

t1 = (lad + l12d + lfd)/rfd;
t2 = (lad + l12d + l1d)/r1d;
t3 = (l1d + ((lad + l12d)*lfd/(lad+lfd)))/r1d;
t4 = (l12d + lfd + (lad*xl/(lad+xl)))/rfd;
t5 = (l12d + l1d + (lad*xl/(lad+xl)))/r1d;
t6 = (l1d + (lfd*(l12d*lad+l12d*xl+lad*xl)/(lad*xl+lad*(lfd+l12d)+(lfd+l12d)*xl)))/r1d;

f(1) = (taud0_p - 2*t1*t3/((t1+t2)-sqrt((t1+t2)^2-4*t1*t3))) / taud0_p;
f(2) = (taud0_pp - 2*t1*t3/((t1+t2)+sqrt((t1+t2)^2-4*t1*t3))) / taud0_pp;
f(3)  =  (xd_p  -  xd*(t4*t6)/(t1*t3)*((t1+t2)-sqrt((t1+t2)^2-4*t1*t3))/((t4+t5)-sqrt((t4+t5)^2-
4*t4*t6))) / xd_p;
f(4) = (xd_pp - xd*(t4*t6)/(t1*t3)) / xd_pp;
```

Standard parameters are determined by curve-fitting technique from the operational Lq(s):

```
% Parameter Lq(s)
load 'mod_ang_Lq_rev1.txt'
Lq=mod_ang_Lq_rev1;

Frequency_Lq=Lq(:,1);
Module_Lq=Lq(:,2);
Argument_Lq=Lq(:,3);

for k_Lq=1:1:length(Lq);
  h_Lq(k_Lq)=(Lq(k_Lq,2)*cos(Lq(k_Lq,3)/180*pi)+i*Lq(k_Lq,2)*sin(Lq(k_Lq,3)/180*pi));
   w_Lq(k_Lq)=2*pi*Lq(k_Lq,1);
end
for k_Lq=1:1:length(w_Lq)/2-10
    Wt_Lq(k_Lq)=1*2400;
end
for k_Lq=length(w_Lq)/2-10+1:1:length(w_Lq)/2-10+20
    Wt_Lq(k_Lq)=1*8800;
end
for k_Lq=length(w_Lq)/2-10+21:1:length(w_Lq)
    Wt_Lq(k_Lq)=1*7000;
```

```
end

[b_Lq,a_Lq]=invfreqs(h_Lq,w_Lq,2,2,Wt_Lq,1200,1e-19,'trace');
[Estimated_h_Lq,Estimated_w_Lq]=freqs(b_Lq,a_Lq,w_Lq);

figure(1)
loglog(w_Lq,abs(h_Lq),'r',Estimated_w_Lq,abs(Estimated_h_Lq),'b');
title ('Module Lq vs Estimated Module Lq(s)')
xlabel('Frequency (Hz)')
ylabel('Module (H)')
legend('Real measured','Estimated')
grid on

figure(2)
semilogx(w_Lq,angle(h_Lq)*180/pi,'r',Estimated_w_Lq,angle(Estimated_h_Lq)*180/pi,'b');
title ('Argument Lq vs Estimated Argument Lq(s)')
xlabel('Frequency (Hz)')
ylabel('Argument (H)')
legend('Real measured','Estimated')
grid on

sol1=roots(a_Lq);     %%solve the poles a
Tqop=-1/sol1(2);
Tqopp=-1/sol1(1);
sol2=roots(b_Lq);     %%solve the zeros b
Tqp=-1/sol2(2);
Tqpp=-1/sol2(1);

Lqp=Lq(1,2)*Tqp/Tqop;
Lqpp=Lq(1,2)*Tqp*Tqpp/(Tqop*Tqopp);
```

R-L parameters for the q-axis equivalent circuit are calculated from the standard parameters:

```
% Parameter data
xq = 7.950e-3 / lb;
xl = 0.795e-3 / lb;

% Data from Lq fitting
tauq0_p = 1.8346 * wb;
tauq0_pp = 0.3251 * wb;
tauq_p = 0.999 * wb;
```

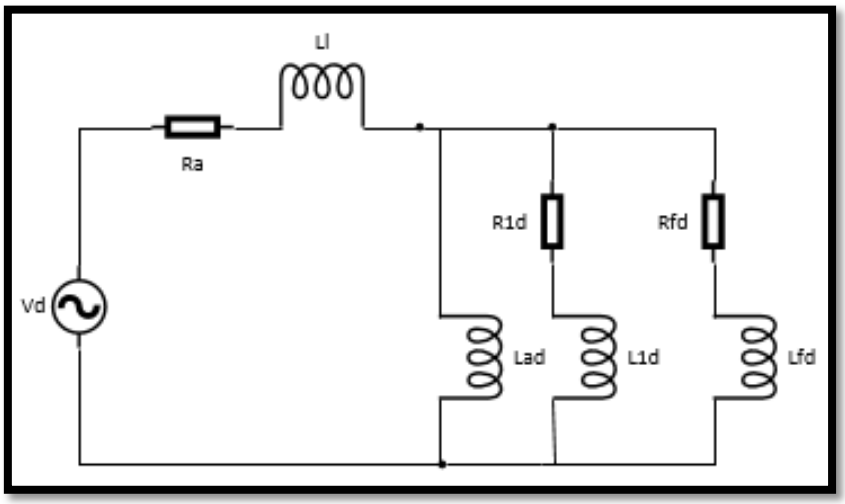
```
tauq_pp = 0.0947* wb;
%
% Calculations
xq_pp = xq * tauq_p * tauq_pp / (tauq0_p * tauq0_pp);
xq_p  = xq * tauq_p / tauq0_p;
laq = xq - xl;
params = [xq_pp,xq_p,xq,xl,tauq0_pp,tauq0_p,laq];
options = optimset('MaxFunEvals',2e4,'MaxIter',2000,'TolFun',1e-8,'TolX',1e-7);
%
% Initializations
l2q0 = (xq_p-xl)*laq / (laq-xq_p+xl);
l1q0 = (xq_pp-xl)*laq*l2q0 / ((laq*l2q0)-((xq_pp-xl)*(laq+l2q0)));
r1q0 = (l1q0 + ((laq*l2q0)/(laq+l2q0))) / tauq0_pp;
r2q0 = (laq+l2q0) / tauq0_p; %%%
%
x0 = [l2q0; l1q0; r1q0; r2q0];
[x,fval,exitflag] = fsolve('fsol_eje_q_r',x0,options,params);
l2q = x(1);    l1q = x(2);    r1q = x(3);  r2q = x(4);
if exitflag == 0,
   h = warndlg('Maximum iteration number has been exceeded','Warning: axe q results');
end


%%
% Parameters in quadrature axis

xq_pp = params(1);      xq_p = params(2);      xq = params(3);
xl = params(4);         tauq0_pp = params(5);  tauq0_p = params(6);
laq = params(7);

l2q = x(1);    l1q = x(2);    r1q = x(3);    r2q = x(4);

t1 = (laq + l2q)/r2q;
t2 = (laq + l1q)/r1q;
t3 = (l1q + ((laq *l2q)/(laq+l2q)))/r1q;

t4 = (l2q + (laq*xl/(laq+xl)))/r2q;
t5 = (l1q + (laq*xl/(laq+xl)))/r1q;

t6 = (l1q + (laq*l2q*xl)/(laq*xl+laq*l2q+l2q*xl))/r1q;

f(1) = (tauq0_p - 2*t1*t3/((t1+t2)-sqrt((t1+t2)^2-4*t1*t3))) / tauq0_p;
f(2) = (tauq0_pp - 2*t1*t3/((t1+t2)+sqrt((t1+t2)^2-4*t1*t3))) / tauq0_pp;
```

f(3) = (xq_p - xq*(t4*t6)/(t1*t3)*((t1+t2)-sqrt((t1+t2)^2-4*t1*t3))/((t4+t5)-sqrt((t4+t5)^2-4*t4*t6))) / xq_p;
f(4) = (xq_pp - xq*(t4*t6)/(t1*t3)) / xq_pp;

- **MATLAB code to determine the transfer functions.**

The transfer function of Ld(s) corresponding to the following equivalent circuit is:

$$L_d(s) = Ld * \frac{(1 + sT'_d) * (1 + sT''_d)}{(1 + sT'_{do}) * (1 + sT''_{do})}$$



5.42. Equivalent circuit

For another equivalent circuit the transfer function can be determinated in a similar way.

For example, if the inductance L12d is added to the circuit, then a 2nd order transfer function is obtained, but if a second dumper winding is added (shown in next figure), then 3rd order transfer function is obtained, been necessary another additional test date from G(s).

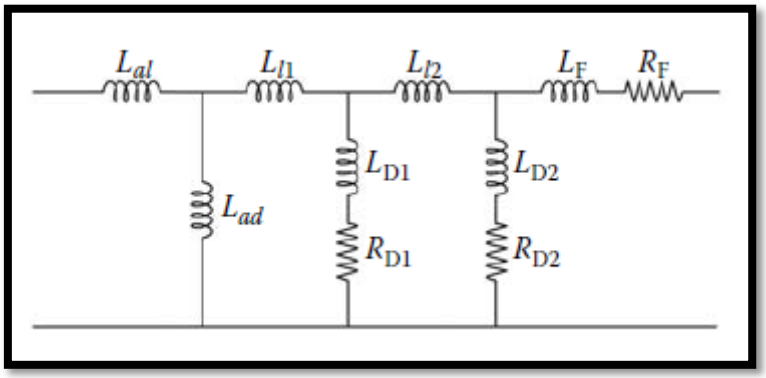


Figure 5.43. Equivalent circuit of 3rd order.

$$G(s) = \frac{L_{ad}}{R_F} \frac{(1+sT_{kd1})(1+sT_{kd2})}{(1+sT'_{d0})(1+sT''_{d0})(1+sT'''_{d0})}$$

$$L_d(s) = L_d \frac{(1+sT'_d)(1+sT''_d)(1+sT'''_d)}{(1+sT'_{d0})(1+sT''_{d0})(1+sT'''_{d0})}$$

Figure 5.44. Transfer function 3rd order.

The first step is define the symbolic parameters that there are represented in the equivalent circuit. The following steps consist on the association of the components in order to obtain the symbolic transfer function.

The calculated coefficients are equal to the corresponding constant time in the transfer function. In this way, a symbolic equation system can be obtained in order to find the parameters of the circuit.

The next example corresponds to the equivalent circuit of the figure.

To develop the MATLAB code, we will work with the following symbolic variables:

Lfd, Rfd, L1d, R1d, Lad, Ll

Where,

Ld = Lad + Ll

The MATLAB code is:

```
% First, we do the parallel of the circuit with the excitation and damper windings.

Zfd=Lfd*s+Rfd;
Z1d=L1d*s+R1d;

Z1=Zfd*Z1d/(Zfd+Z1d);
[n_Z1, d_Z1] = numden(Z1);

% As we will see below, there are two options depending on the coefficients that exist. In the equation, we have a second-order numerator with 3 coefficients. Therefore, we will make the first loop.

% Second order in numerator and first order in denominator
```

```
%Terms of numerator Z1

n_Z1_coefs=children(collect(n_Z1));
n_Z1_order=length(n_Z1_coefs)
if n_Z1_order==3
   for k=length(n_Z1_coefs):-1:1
      n1(length(n_Z1_coefs)-k+1)=n_Z1_coefs(length(n_Z1_coefs)-k+1)/s^(k-1)
   end
else
   for k=length(n_Z1_coefs):-1:2
      n1(length(n_Z1_coefs)-k+1)=n_Z1_coefs(length(n_Z1_coefs)-k+1)/s^(k-1)
   end
end

% Now, we again have two for loops that again depend on the coefficients. In the equation, we have a denominator of first order with 2 coefficients, therefore, we will make the first loop.
%Terms of denominator Z1
d_Z1_coefs=children(collect(d_Z1))
% if d_Z1_order=2 then d_Z1=u1*s+w1
% else c_Z1=u1*s
% end
d_Z1_order=length(d_Z1_coefs)
if d_Z1_order==2
   for k=length(d_Z1_coefs):-1:1
      d1(length(d_Z1_coefs)-k+1)=d_Z1_coefs(length(d_Z1_coefs)-k+1)/s^(k-1)
   end
else
   for k=length(d_Z1_coefs):-1:2
      d1(length(d_Z1_coefs)-k+1)=d_Z1_coefs(length(d_Z1_coefs)-k+1)/s^k
   end
end

% 2nd parallel of the circuit with the first parallel and the magnetizing inductance Lad.

Z2=(Z1*Lad*s)/(Z1+Lad*s)
[n_Z2, d_Z2] = numden(Z2)


% We have the two possible options again. Since we have a third order numerator with 3 coefficients, we will execute the second loop.

% Third order in numerator and second order in denominator
% Terms of numerator Z2
```

```
n_Z2_coefs=children(collect(n_Z2))

n_Z2_order=length(n_Z2_coefs)
if n_Z2_order==4
   for k_Zn2=length(n_Z2_coefs):-1:1
      n2(length(n_Z2_coefs)-k_Zn2+1)=n_Z2_coefs(length(n_Z2_coefs)-k_Zn2+1)/s^(k_Zn2-1)
   end
else
   for k_Zn2=length(n_Z2_coefs):-1:1
      n2(length(n_Z2_coefs)-k_Zn2+1)=n_Z2_coefs(length(n_Z2_coefs)-k_Zn2+1)/s^(k_Zn2)
   end
end



%Now we have in the equation a denominator of second order with 3 coefficients, therefore, the first loop will be executed.
%Terms of denominator Z2
d_Z2_coefs=children(collect(d_Z2))
% if d_Z1_order=2 then d_Z1=u1*s+w1
% else c_Z1=u1*s
% end
d_Z2_order=length(d_Z2_coefs)
if d_Z2_order==3
   for k_Zd2=length(d_Z2_coefs):-1:1
      d2(length(d_Z2_coefs)-k_Zd2+1)=d_Z2_coefs(length(d_Z2_coefs)-k_Zd2+1)/s^(k_Zd2-1)
   end
else
   for k_Zd2=length(d_Z2_coefs):-1:1
      d2(length(d_Z2_coefs)-k_Zd2+1)=d_Z2_coefs(length(d_Z2_coefs)-k_Zd2+1)/s^(k_Zd2)
   end
end



% Now, we add the equivalent circuit that we have before the leakage inductance (Ll)

Z3=Ll*s+Z2
[n_Z3, d_Z3] = numden(Z3)

%In this case, we have a numerator of third order with 3 coefficients, therefore, we execute the second loop.
```

```
% Third order in numerator and second order in denominator
% Terms of numerator Z3

n_Z3_coefs=children(collect(n_Z3))

n_Z3_order=length(n_Z3_coefs)
if n_Z3_order==4
   for k_Zn3=length(n_Z3_coefs):-1:1
      n3(length(n_Z3_coefs)-k_Zn3+1)=n_Z3_coefs(length(n_Z3_coefs)-k_Zn3+1)/s^(k_Zn3-1)
   end
else
   for k_Zn3=length(n_Z3_coefs):-1:1
      n3(length(n_Z3_coefs)-k_Zn3+1)=n_Z3_coefs(length(n_Z3_coefs)-k_Zn3+1)/s^(k_Zn3)
   end
end


% Finally, we have a denominator of second order with 3 coefficients, therefore, the first block
will be executed.

%Terms of denominator Z3
d_Z3_coefs=children(collect(d_Z3))
% if d_Z1_order=2 then d_Z1=u1*s+w1
% else c_Z1=u1*s
% end
d_Z3_order=length(d_Z3_coefs)
if d_Z3_order==3
   for k_Zd3=length(d_Z3_coefs):-1:1
      d3(length(d_Z3_coefs)-k_Zd3+1)=d_Z3_coefs(length(d_Z3_coefs)-k_Zd3+1)/s^(k_Zd3-1)
   end
else
   for k_Zd3=length(d_Z3_coefs):-1:1
      d3(length(d_Z3_coefs)-k_Zd3+1)=d_Z3_coefs(length(d_Z3_coefs)-k_Zd3+1)/s^(k_Zd3-1)
   end
end
```

- **Transfer function developed from Ld (s)**

The transfer function in symbolic variables with all the components of the 2nd order circuit are the following:

```
%%
%Transfer function second order
%Ld(s)=Ld(0)*(1+(T4d+T5d)*s+(T4d*T6d)*s^2)/(1+(T1d+T2d)*s+(T1d*T3d)*s^2)
Ld_0=simplify(n3(1,3)/d3(1,3))

Ld_tr_num_coef1=simplify(n3(1,1)/n3(1,3)) %s^2
Ld_tr_num_coef2=simplify(n3(1,2)/n3(1,3)) %s

Ld_tr_den_coef1=simplify(d3(1,1)/d3(1,3)) %s^2
Ld_tr_den_coef2=simplify(d3(1,2)/d3(1,3)) %s

Ld_tr=Ld_0*((Ld_tr_num_coef1*s^2+Ld_tr_num_coef2*s+1)...
   /(Ld_tr_den_coef1*s^2+Ld_tr_den_coef2*s+1))
```

- **Comparison between the equations of time constants T1d, T2d, T3d, T4d, T5d and T6d given by the bibliography of outstanding authors in the matter and the development with the parameters obtained from the equivalent circuit of 2nd order.**

Despite being different methods, if we calculate the two separately, the difference of both, if they are really equal, must be 0:

```
%Ld(s)=Ld(0)*(1+(T4d+T5d)*s+(T4d*T6d)*s^2)/(1+(T1d+T2d)*s+(T1d*T3d)*s^2)

dif_T4dxT6d=T4d*T6d-T4dxT6d;
dif_T4dxT6d_sim=simplify(dif_T4dxT6d)

dif_T4d_T5d=(T4d+T5d)-T4d_T5d;
dif_T4d_T5d_sim=simplify(dif_T4d_T5d)

dif_T1dxT3d=T1d*T3d-T1dxT3d;
dif_T1dxT3d_sim=simplify(dif_T1dxT3d)

dif_T1d_T2d=T1d+T2d-T1d_T2d;
dif_T1d_T2d_sim=simplify(dif_T1d_T2d)
```

```
dif_T4dxT6d_sim =

0


dif_T4d_T5d_sim =

0


dif_T1dxT3d_sim =

0


dif_T1d_T2d_sim =

0
```

Figure 5.45. Result of comparison.

The difference between the terms calculated and those obtained by the equivalent circuit is 0. Therefore, the equations of the time constants are correct, and the objective has been reached to be able to develop the transfer function of Ld (s).

5.6 Parameter identification based on design data. Three phase inverter + PMSM motor

1. Final model. Equations description
2. Parameters involved
3. How to measure each parameter

Datasheet:

- number of the machine pole pairs
- rated machine current
- rated machine voltage
- stator resistance at 20 ºC *

* Usually the resistance value at 20 °C is listed in the machine's datasheet, if not can be measured as indicated in the following section.

Measurement [32]:

- stator resistance can be measured by an RLC meter
- synchronous inductances:

The synchronous inductances can be measured by using a DC power supply, an oscilloscope and a current and voltage probe.

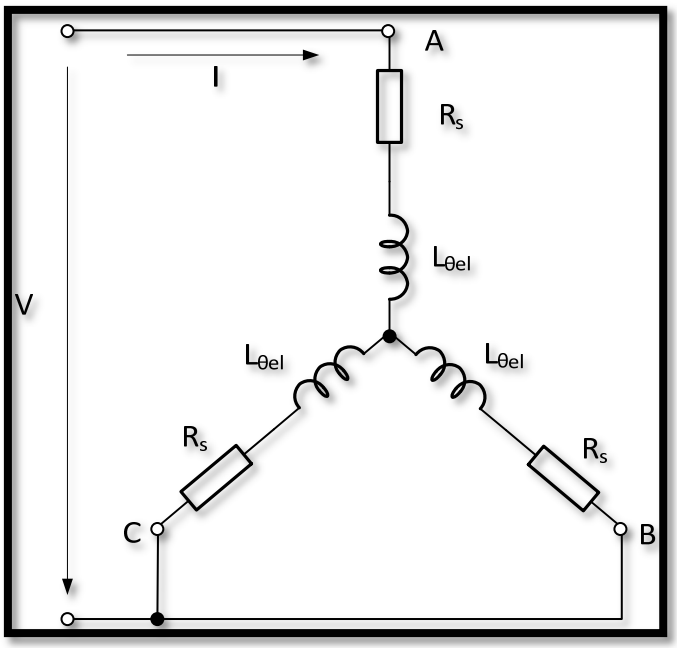


Figure 5.46. Inductance measurement circuit

The alignment into d-axis is done by phase A connected to the positive potential (+) and phase B and C are grounded (-). The rotor shaft must be locked when a voltage step is applied, then the step response of the current is measured and the d-axis inductance can be calculated by using the following equation:

$$L_d = \frac{2}{3}\tau R$$

[32] V Bobek, Pmsm electrical parameters measurement, Freescale Semiconductor, 2013.

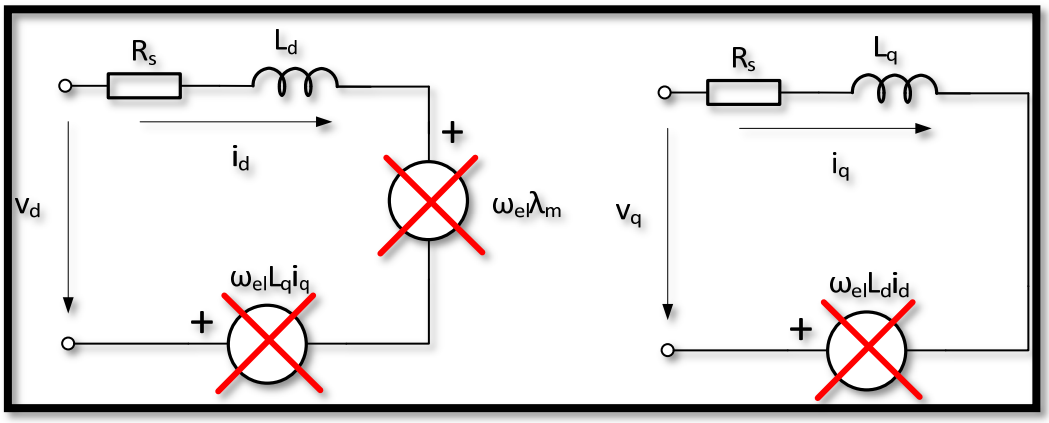


Figure 5.47. Equivalent phase model of PMSM in d/q axis for a locked rotor shaft

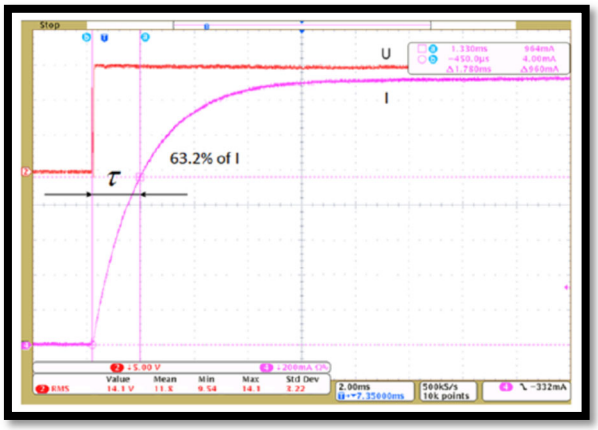


Figure 5.48. Current step response waveform

The alignment into q-axis is done by phase B connected to the positive potential (+), phase C is grounded (-), and phase A is floating (NC). The rotor shaft must be locked when a voltage step is applied between phase A connected to the positive potential (+) of the voltage source and phases B and C are grounded, then the step response of the current is measured and the q-axis inductance can be calculated by using the following equation:

$$L_q = \frac{2}{3}\tau R$$

- back-EMF constant

The back-EMF (BEMF) constant can be obtained by using a driving motor that drives the PMSM at a constant speed, an oscilloscope and voltage probes are used in order to measure the no-load peak phase voltage if the neutral point of the motor is accessible, in other case the no-load peak line-to-line voltage is measured or it's possible to create the artificial neutral point from all three voltage probe clips connected together.

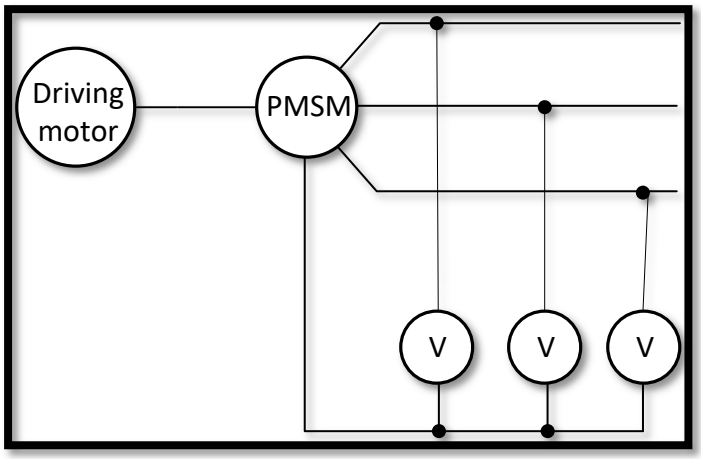


Figure 5.49. Three-phase measurement of the BEMF constant

If the line-to-line voltage is measured then the Back-EMF constant can be calculated according to:

$$K_e = \frac{V_{pk-pk} \cdot T_{el}}{4\pi\sqrt{3}}$$

If the phase voltage is measured then the Back-EMF constant can be calculated according to:

$$K_e = \frac{V_{pk-pk} \cdot T_{el}}{4\pi}$$

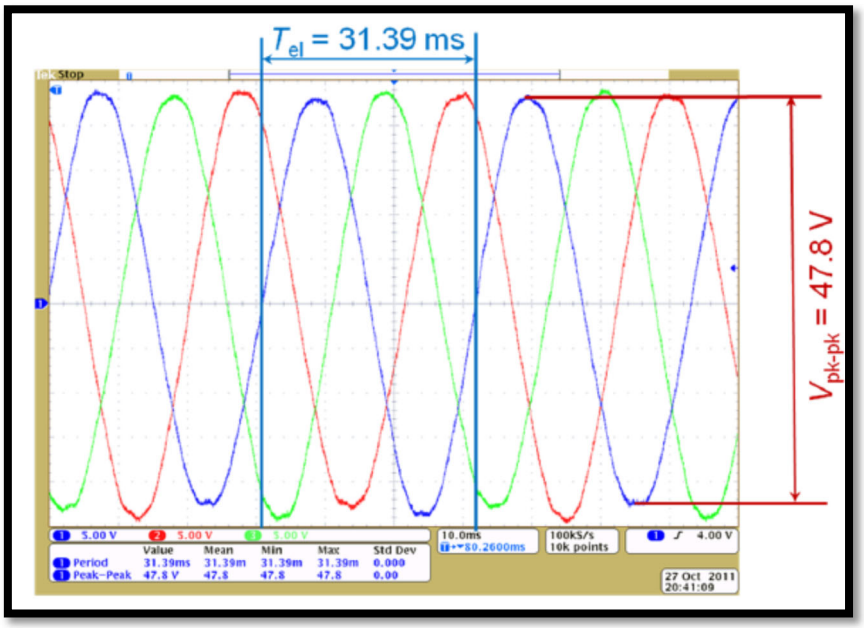


Figure 5.50. Three-phase oscilloscope measurement of the electrical constant

Measurement of frictional coefficient (B) [33]:

<u>Measurement of no-load torque versus speed:</u>

The machine is run on no-load at different speeds in order to find the frictional coefficient. The values of the stator current are measured at steady state at each speed. The electromagnetic torque can be calculated:

$$\omega = \frac{p}{2} \cdot \omega_m$$

$$\theta = \omega \cdot t$$

$$\boldsymbol{i_s^{qd0}} = \boldsymbol{T(\theta)} \cdot \boldsymbol{i_s^{abc}}$$

$$T_e = \frac{3}{2} \cdot \frac{p}{2} \cdot \left[\psi_m \cdot i_q + \left(L_d - L_q\right) \cdot i_q \cdot i_d\right]$$

[33] VSSPK Hari, A Tripathi, G Narayanan, Experimental determination of mechanical parameters in sensorless vector-controlled induction motor drive, Sādhanā, Springer, 2017.

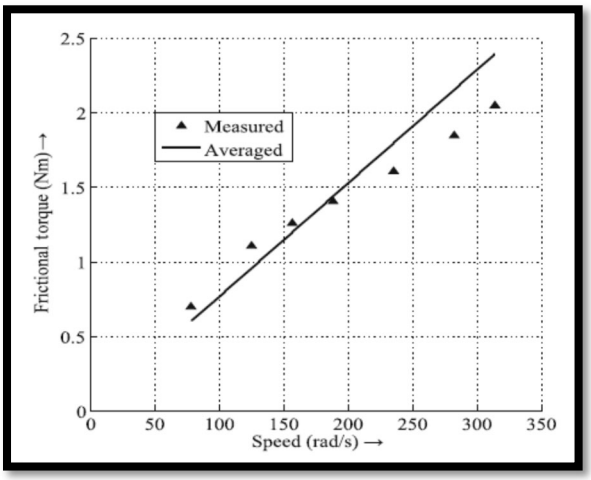


Figure 5.51. Comparison of measured and averaged frictional load torque constant

The variation of the frictional torque can be quite linear with speed as it is shown in previous figure. If the machine is run on no-load at steady state then the electromagnetic torque is equal to the frictional torque, then frictional coefficient can be calculated by using the next equation:

$$B = \frac{T_e}{\omega_m}$$

The average value of B from different measurements can be considered as the measured frictional coefficient.

If the frictional torque is a non-linear function of speed then it is required that the function fit on the measured no-load torque. In this case, usually, it could be assumed a quadratic function:

$$T_f = B_0 + B_1 \cdot \omega_m + B_2 \cdot \omega_m^2$$

Measurement of moment of inertia (J):

The motor drive is operated at no-load with appropriate references in order to achieve a constant torque. Speed and currents are measured. Since torque depends on the currents, then currents should be maintained constant.

If the variation of the frictional torque can be considered linear with speed then the speed response under the influence of a constant torque is:

$$\omega_m(t) = \omega_0 \cdot e^{-\frac{B}{J}t} + \frac{T_e}{B} \cdot \left(1 - e^{-\frac{B}{J}t}\right)$$

Where of ω_0 is the initial speed.

The moment of inertia can be estimated by curve fitting at different torques and the average of these values is taken as the measured of moment of inertia.

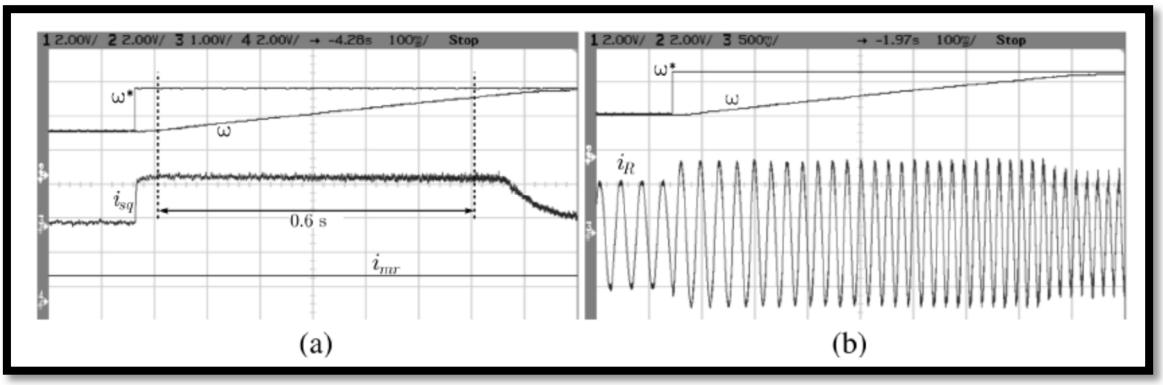


Figure 5.52. Experimental results

Previous figure shows the reference speed, rotor speed and the q-axis current for the case of acceleration under constant torque condition.

If the variation of the frictional torque can be considered a quadratic function with speed then the speed response under the influence of a constant torque is:

$$\omega_m(t) = \frac{K_1 - K_2 \cdot \left(\frac{\omega_0 - K_1}{\omega_0 - K_2}\right) e^{\frac{(K_1 - K_2)B_2}{J}t}}{1 - \left(\frac{\omega_0 - K_1}{\omega_0 - K_2}\right) e^{\frac{(K_1 - K_2)B_2}{J}t}}$$

Where K_1 and K_2 are respectively:

$$K_1 = \frac{-B_1 - \sqrt{B_1^2 + 4 \cdot (T_e - B_0) \cdot B_2}}{2 \cdot B_2}$$

$$K_2 = \frac{-B_1 + \sqrt{B_1^2 + 4 \cdot (T_e - B_0) \cdot B_2}}{2 \cdot B_2}$$

Again, the moment of inertia can be estimated by curve fitting at different torques and the average of these values is taken as the measured of moment of inertia.

5.7 Parameter identification in DC/DC BUCK converters

5.7.1 DC/DC BUCK converters. Parameter identification based on manufacturers' data

A **buck converter** (step-down converter) steps down the voltage (or steps up the current) from the input (supply) to the output (load).

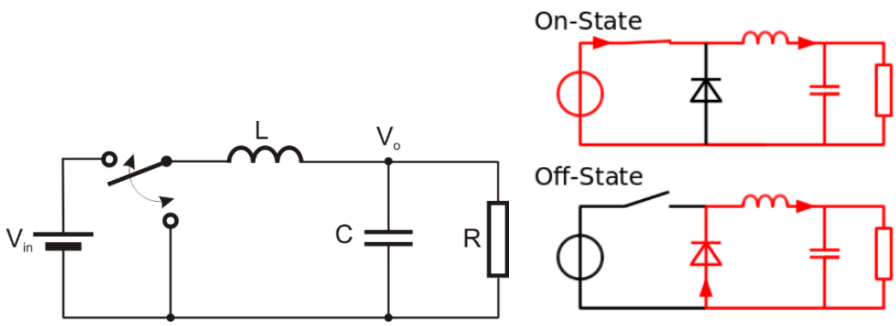


Figure 5.45. DC/DC BUCK Converter.

In general from the manufacturers' data, the available data will be input voltage (V_{input}), output voltage (V_{output}), output current (I_{output}) and sampling frequency (*f*).

The values of L, C will be confidential and the manufacturer will not provide it. For calculating the L and C additional data like output ripple current and ripple voltage is required. With these values the $L_{minimum}$,$C_{minimum}$ required for the converter can be calculated.

During T_{on} we can approximate the value of *L* as,

$V_{output} = D \cdot V_{input}$ (*D*: duty cycle)

$$V_L = V_{input} - V_{output}$$

$$V_L = L\frac{dI_L}{dt}$$

$$L = \frac{V_L \cdot T_{on}}{\Delta I_L}$$

$$L = \frac{(V_{input} - V_{output}) \cdot T_{on}}{\Delta I_L} = \frac{(V_{input} - V_{output}) \cdot D / f}{\Delta I} = \frac{V_{output} \cdot (V_{input} - V_{output})}{\Delta I \cdot f \cdot V_{input}}$$

For calculating the value of *C* the graphs below are used.

Output voltage ripple phenomenon[34] where the output voltage increases during the On-state and decreases during the Off-state.

[34] P.Ned Mohan, power electronics converter and applications,Newyork:John wiley and sons,1995, pages 164 to 171

The output voltage will increase and decrease because of the charging and discharging of the output capacitor.

$$i_C = C\frac{dv_C}{dt}$$

$$\Delta V = \frac{i \cdot \Delta T}{C} \quad \text{(A)}$$

The equation (A) is the general formula for calculating the ripple output voltage.

The output voltage ripple is caused mainly by the capacitor charging and the inductor current during T_{on} and the capacitor discharging during T_{off}

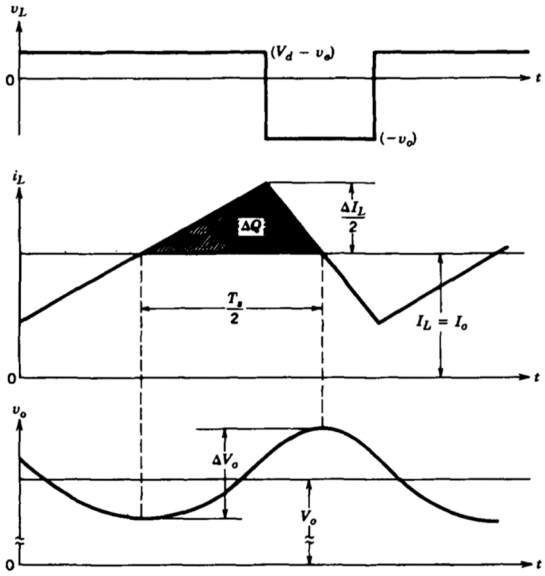


Figure 5.46. Inductor current and output ripple voltage for the buck converter

The output ripple voltage can be calculated as area of the shaded region

$$\Delta V = \frac{\Delta Q}{C}$$

$$C = \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{1}{2} \cdot \frac{\Delta I_L \cdot T}{\Delta V} = \frac{\Delta I_L}{8 \cdot f \cdot \Delta V_{out}}$$

For the data below:

V_{input}	5 V
V_{output}	3.3 V
I_{output}	1.5 A
f	1.25 MHz

From the datasheet, it is not possible to calculate directly the values of L, C, so some assumptions must be done. From the data sheet[35] , by making the assumption that the output current ripple is around the 30% of the maximum I_{output} and the ripple voltage is 2 % of output voltage, it results:

From Simulation, the ripple current is 25% of max current and voltage droop is 3.9% of output voltage

Considering manufacturers' data:

$$L_{\min} = \frac{V_{output} \cdot (V_{input} - V_{output})}{\Delta I \cdot f \cdot V_{input}} = \frac{5 \cdot (5 - 3.3)}{(0.3 \cdot 1.5) \cdot 1.25 \cdot 10^6 \cdot 3.3} = 1.99 \text{ uH}$$

$$C_{\min} = \frac{\Delta I \cdot}{8 \cdot f \cdot \Delta V_{out}} = \frac{(0.3 \cdot 1.5)}{8 \cdot 1.25 \cdot 10^6 \cdot (0.02 \cdot 3.3)} = 0.681 \text{ uF}$$

Considering Simulation Data

$$L_{\min} = \frac{V_{output} \cdot (V_{input} - V_{output})}{\Delta I \cdot f \cdot V_{input}} = \frac{5 \cdot (5 - 3.3)}{(0.25 \cdot 1.5) \cdot 1.25 \cdot 10^6 \cdot 3.3} = 5.49 \text{uH}$$

$$C_{\min} = \frac{\Delta I}{8 \cdot f \cdot \Delta V_{out}} = \frac{(0.3 \cdot 1.5)}{8 \cdot 1.25 \cdot 10^6 \cdot (0.039 \cdot 3.3)} = 0.34965 \text{ uF}$$

Parameters	Simulation data	Manufacturers data
L (H)	5.49 μH	1.99 μH
C (F)	349.65 nF	681 nF

Therefore it is concluded that from manufacturer's data only a rough estimation of the parameters is possible.

5.7.2 DC/DC BUCK converters. Parameter identification based on offline tests.

Open circuit tests are well suited for off line parameter identification (short circuit test are not advisable due to the reduced information it provides and problems related to the electrical protections). Open circuit tests must be split into T_{on} (to obtain *L, R_L, C* and R_C) and T_{off} (to obtain the diode resistance R_d).

5.7.2.1 DC/DC BUCK Converter. Parameter identification based on offline tests. Open circuit test (Ton) during steady state conditions

Next figure shows the equivalent circuit of the buck converter during the open circuit test. It is noted that during steady state conditions there is almost no influence of the PID controller.

[35] http://www.ti.com/lit/ds/slus642a/slus642a.pdf

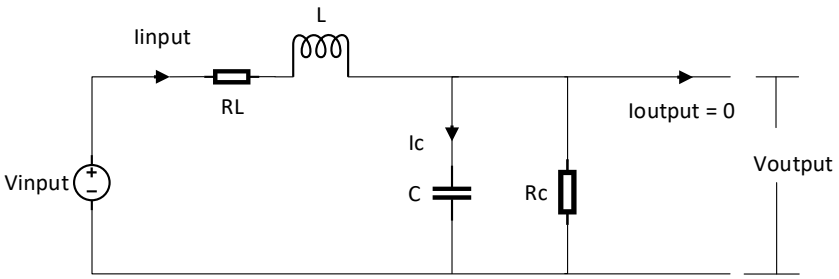


Figure 5.47. Open circuit test. Buck Converter during T_{on}.

The equations describing the T_{on} interval are as follows,

$$I_{input} = I_C + I_{R_C} = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C}$$

$$C\frac{dV_{output}}{dt} = I_{input} - \frac{V_{output}}{R_C} \rightarrow C\int dV_{output} = \int (I_{input} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input} - \frac{V_{output}}{R_c})dt$$

5.7.2.2 DC/DC BUCK converter. Offline tests. Open circuit test (Toff) during steady state conditions

It is noted that during steady state conditions there is almost no influence of the PID controller.

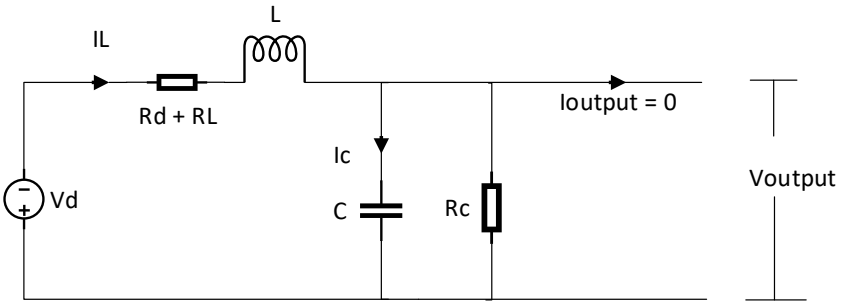


Figure 5.48. Open circuit test. Buck Converter during T_{off}.

The equations describing the T_{on} interval are as follows,

$$-V_d - V_{output} = I_L \cdot (R_d + R_L) + L \cdot \frac{dI_L}{dt} \rightarrow L \cdot \frac{dI_L}{dt} = -V_d - V_{output} - I_L \cdot (R_d + R_L)$$

$$L \cdot \int dI_L = \int -V_d dt - \int V_{output} dt - (R_d + R_L)\int I_L \rightarrow \int dI_L = \int \frac{V_{input} dt}{L} - \int \frac{V_{output} dt}{L} - \frac{R_d + R_L}{L}\int I_L$$

Calculation of I_L during T_{off}

The I_L during T_{off} cannot be measured in a typical real converter, hence the $I_{L,Toff}$ is estimated from the $I_{L,Ton}$.

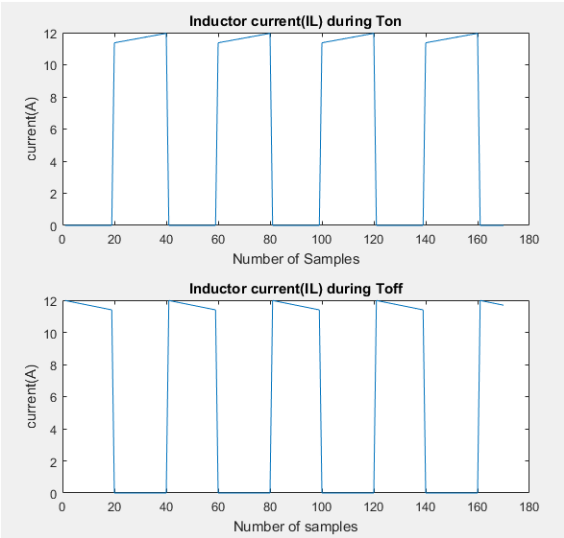


Figure 5.49. The inductor current during T_{off} and T_{on}

The current I_L during T_{off} and T_{on} is represented in Figure 3.

It can be assumed the I_L during T_{off} deceases linearly but actually the current is never linear as shown in Figure 4 hence with using the matlab functions **polyfit** and **polyval** using the data of inductor current(I_L) during Ton the I_L during toff can be estimated.

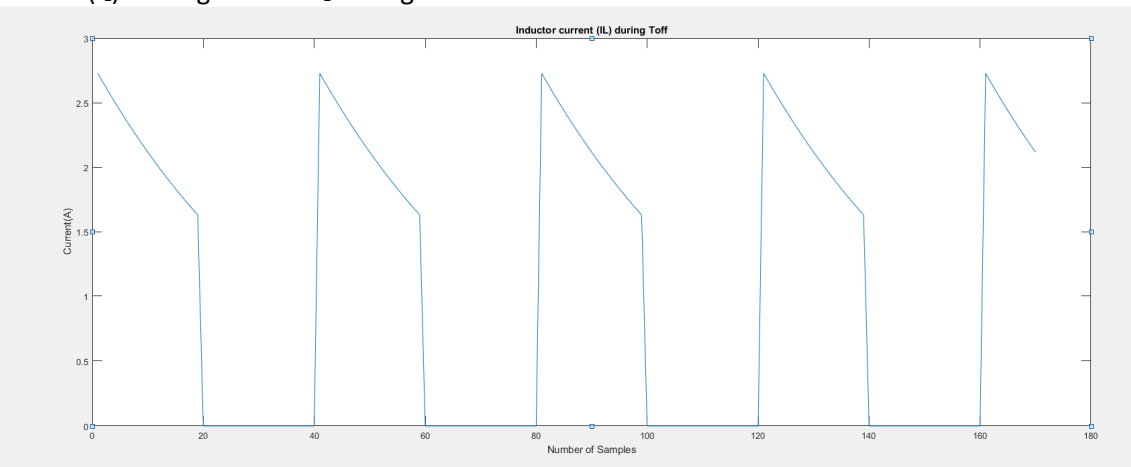


Figure 5.50. Inductor current T_{off} for low load value

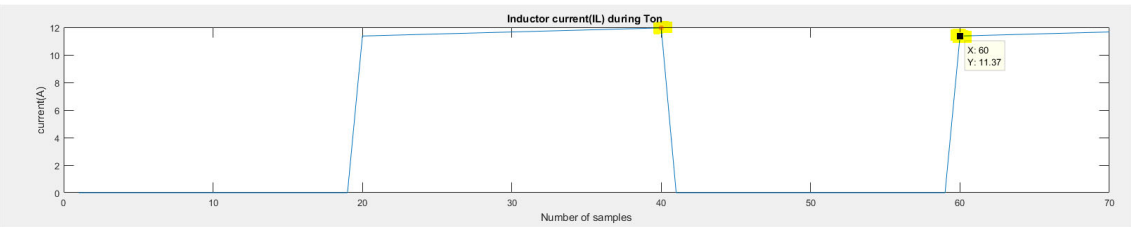


Figure 5.51. The points to calculate inductor current during T_{off} using $I_{L,Ton}$

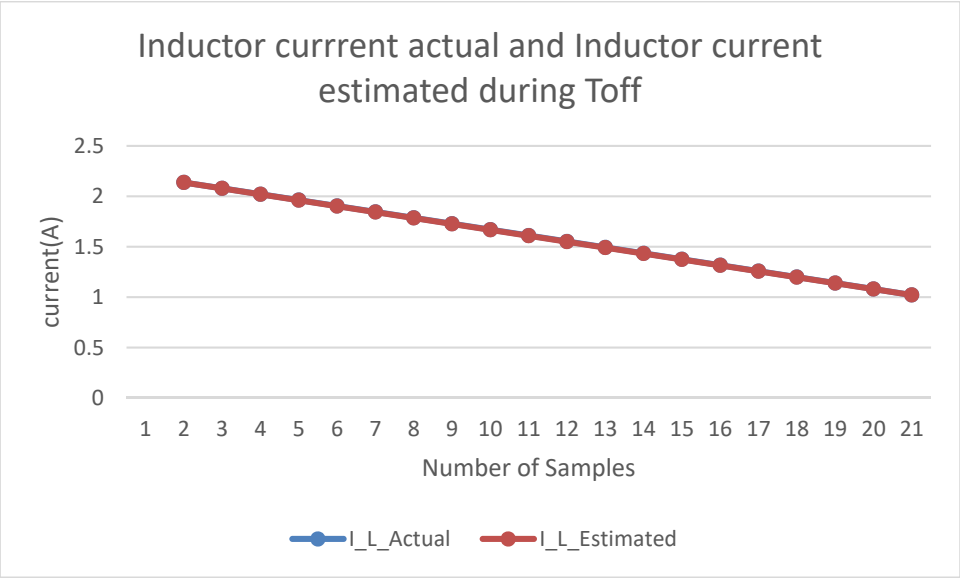


Figure 5.52. Inductor current Actual and Inductor current estimated during T_{off}

Parameters for Open Circuit Test

Parameters	Simulated Value	Calculated Value
L	200 uH	200 uH
C	30 uF	30 uF
R_L	0.06 Ohms	0.06 Ohms
R_c	100 Ohms	100 Ohms
R_d	1e-3 Ohms	0.98e-3 Ohms

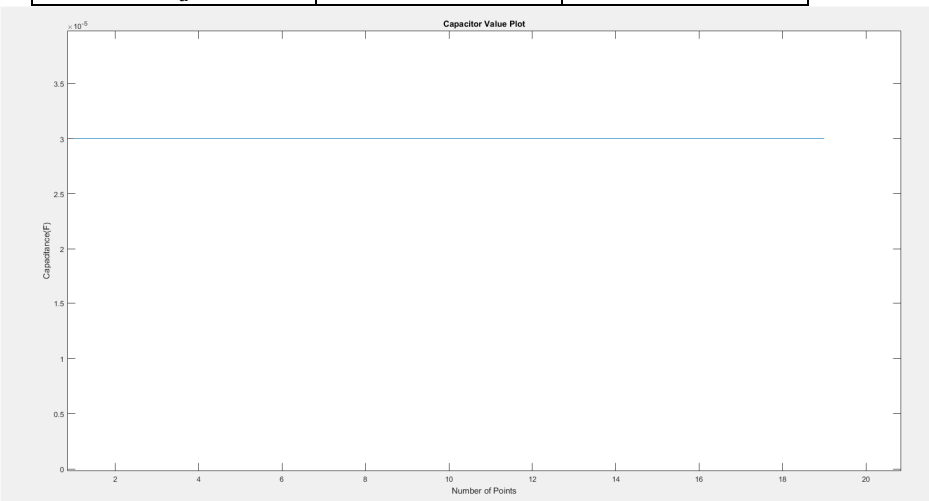


Figure 5.53. The capacitance value plot

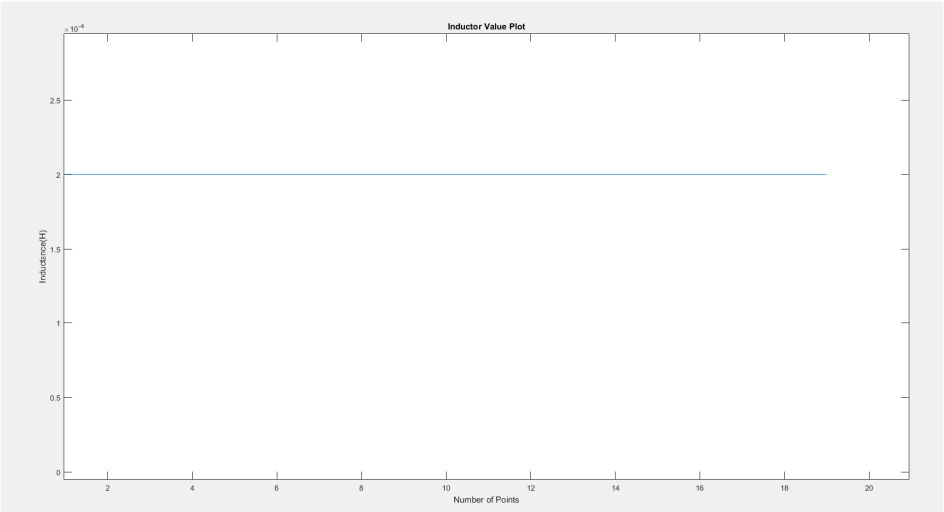


Figure 5.54. The inductance value plot

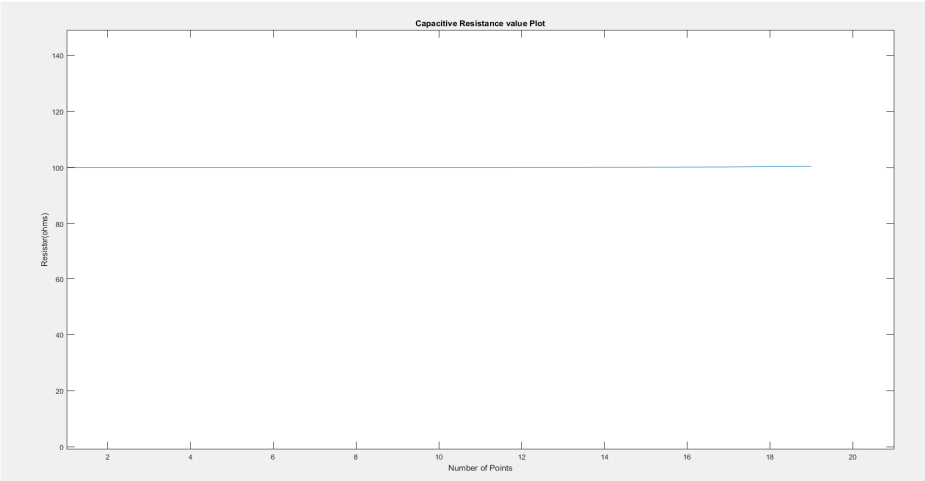


Figure 5.55. The capacitor resistance value plot

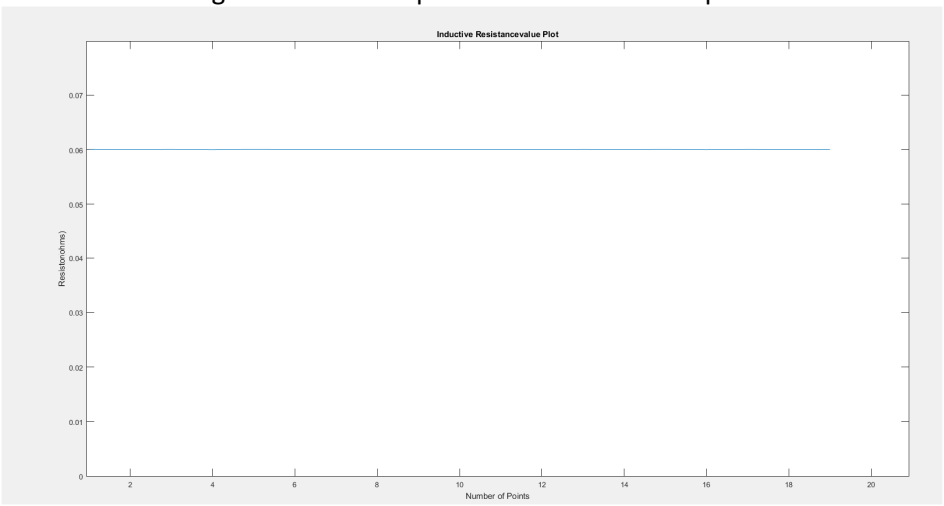


Figure 5.56. The inductor resistance value plot

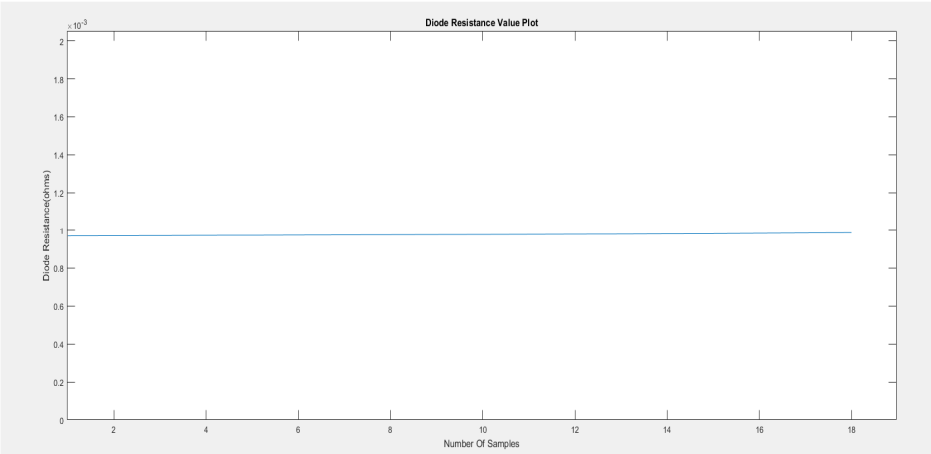


Figure 5.57. The diode resistance value plot

5.7.2.3 DC/DC BUCK converter. Offline tests. Open circuit test (Ton) during transient conditions

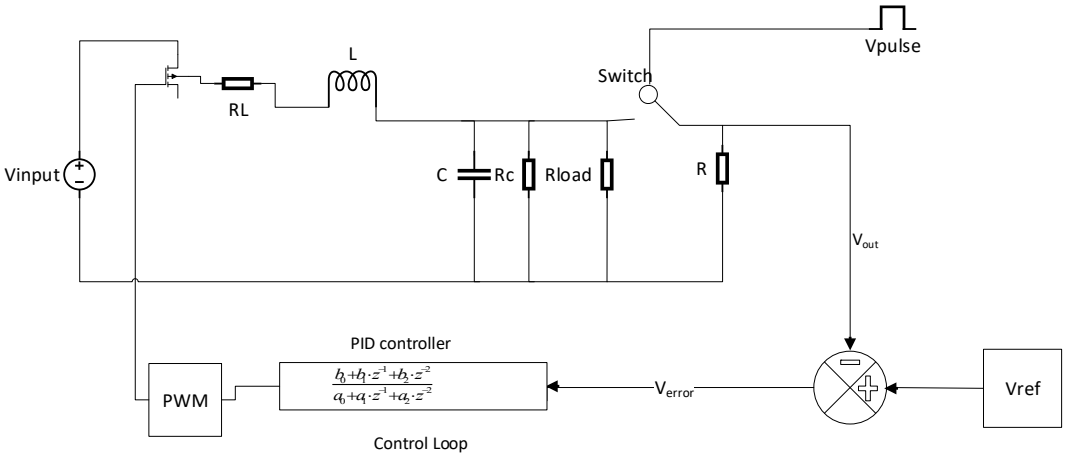


Figure 5.58. Closed Loop buck converter

DC/DC converters usually include a closed loop based on a PID controller to regulate and stabilize the output voltage V_{out} according to the reference value V_{ref}.

The control loop is estimated during the transient state. The transients can be added in the following ways to the circuit

- Adding a random noise in the input voltage
- Switching on and off a resistance in the load side

To perform the identification, a load resistance *R*, which is switched on and off is used to generate transients from the load side, as shown in next figure.

For doing offline test either open loop(No load) is considered.

Considering the open loop test the transients has to be created by adding a noise in the input voltage, but the excitation from the input voltage does not produce rich transients to calculate the parameters of the control loop, Hence another approach needs to be used.

The next approach is using a variable load at the load side, but the R_{load} cannot be opened as it gives rich transients to calculate the closed loop parameters and with only variable load, there is a high risk the converter will be blown due to high inductance current during switching on and off, hence there is always a constant load required in the circuit. One more possible approach instead of open loop the transfer function of the closed loop can be calculated for different values of R_{load}.

Hence the Parameter identification during the transient part is done with different R_{load}. The procedure for calculating the closed loop coefficients is explained in detail in the online method.

5.7.3 DC/DC BUCK converter. White-box. Parameter identification based on online tests

The parameters of buck DC/DC converters can be identified by applying an online white-box-based parameter identification approach during steady-state operation. Such tests must be split into T_{on} and T_{off}.

5.7.3.1 DC/DC BUCK converter. White-box. Parameter identification based on online tests. Steady-state during T_{on}

During T_{on} under steady-state operation, one can calculate the values of *L, R_L, C* and R_C as follows.

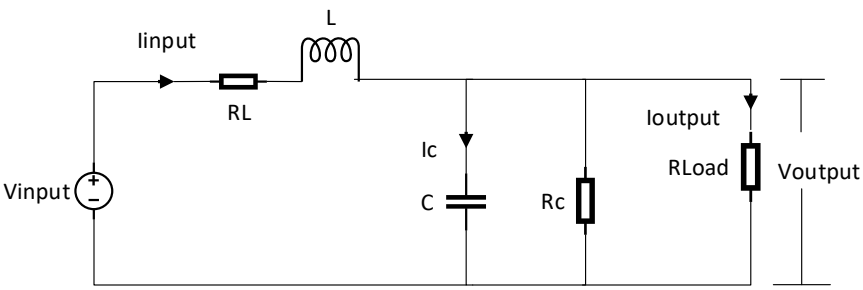


Figure 5.59. Buck DC/DC converter during T_{on}

The equations describing the T_{on} interval are as follows,

$$I_{input} = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C} + I_{output} \rightarrow C\frac{dV_{output}}{dt} = I_{input} - \frac{V_{output}}{R_C} - I_{output}$$

$$C\int dV_{output} = \int (I_{input} - I_{output} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input} - I_{output} - \frac{V_{output}}{R_c})dt$$

5.7.3.2 DC/DC BUCK converter. White-box. Parameter identification based on online tests. Steady-state during T_{off}

During T_{off} under steady-state operation, one can calculate the values of R_d (diode series resistance) as follows.

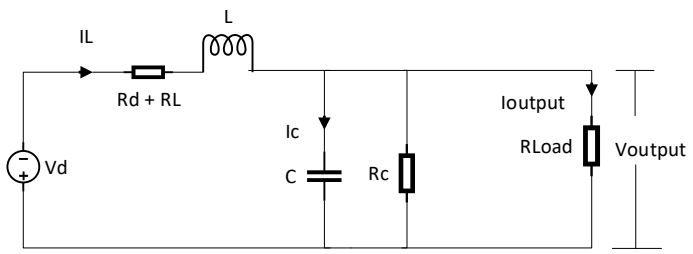


Figure 5.60. Buck DC/DC converter during T_{off}. V_d is the diode voltage.

$$-V_d - V_{output} = I_L \cdot (R_d + R_L) + L \cdot \frac{dI_L}{dt} \rightarrow L \cdot \frac{dI_L}{dt} = -V_d - V_{output} - I_L \cdot (R_d + R_L)$$

$$L \cdot \int dI_L = \int -V_d dt - \int V_{output} dt - (R_d + R_L) \int I_L \rightarrow \int dI_L = \int \frac{V_d dt}{L} - \int \frac{V_{output} dt}{L} - \frac{R_d + R_L}{L} \int I_L$$

Calculation of I_L during T_{off}

The same approach as used for calculating the I_L during T_{off} for offline method can be used.

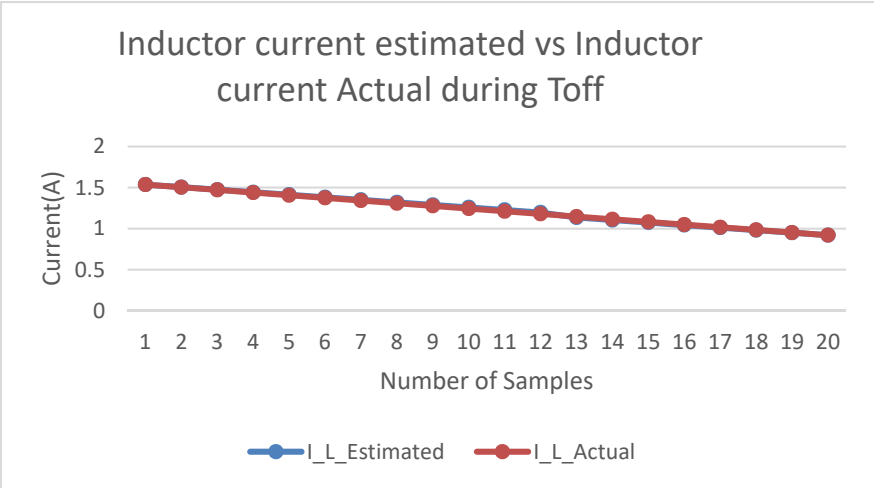


Figure 5.61. Inductor current Actual and Inductor current estimated for Toff

5.7.3.3 DC/DC BUCK converter. White-box. Parameter identification based on online tests. Steady-state during T_{on}

From the following equations with the data of V_{input},V_{output},I_{input},I_{output} the L,C,R_L,R_c are calculated

Table 1: The Parameters for online method

Parameters	Simulated value	Calculated value
L	200 uH	200 uH
C	30 uF	30 uF
R_L	0.06 Ohms	0.06 Ohms
R_c	100 Ohms	100 Ohms
R_d	1e-3 Ohms	0.98e-3 Ohms

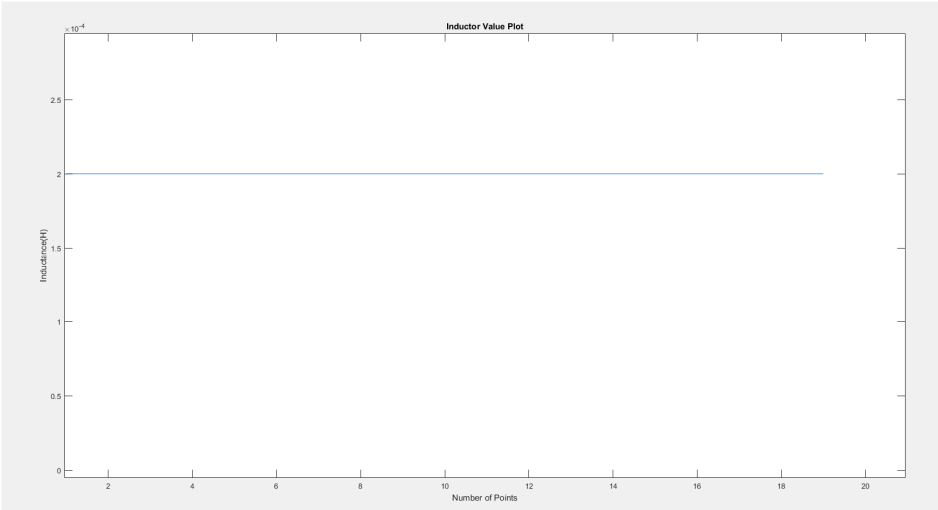


Figure 5.62. The inductance value plot

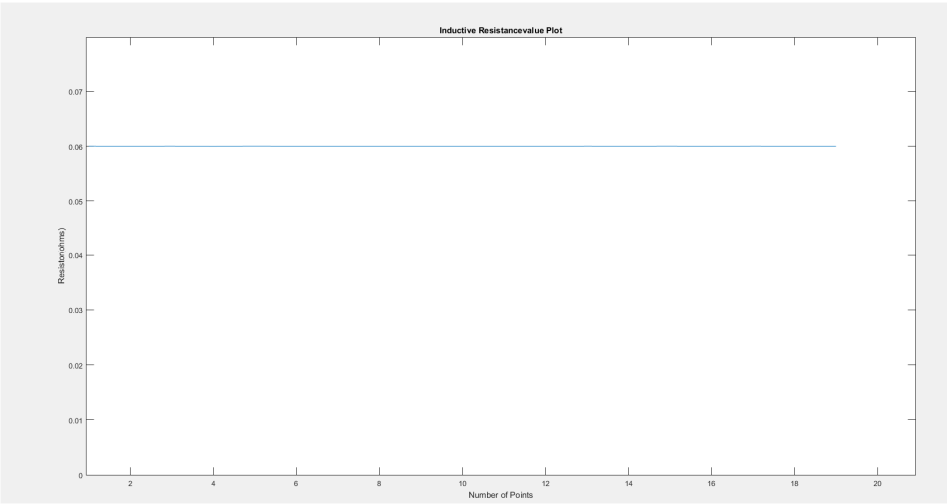


Figure 5.63. The inductive resistor value plot

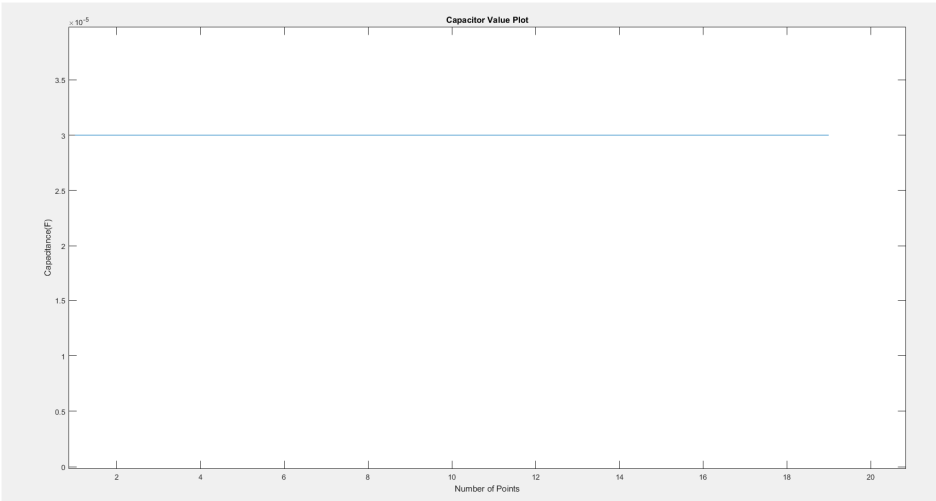


Figure 5.64. The capacitance value plot

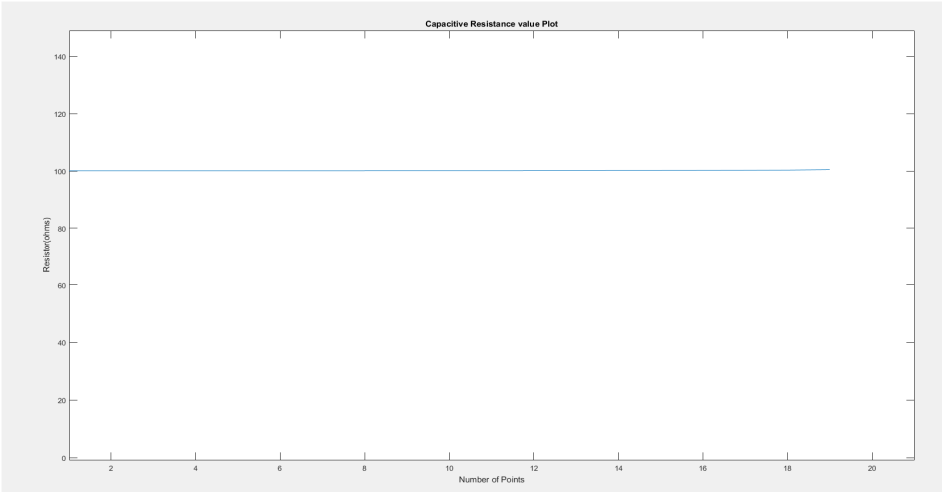


Figure 5.65. The Capacitive Resistance Value Plot

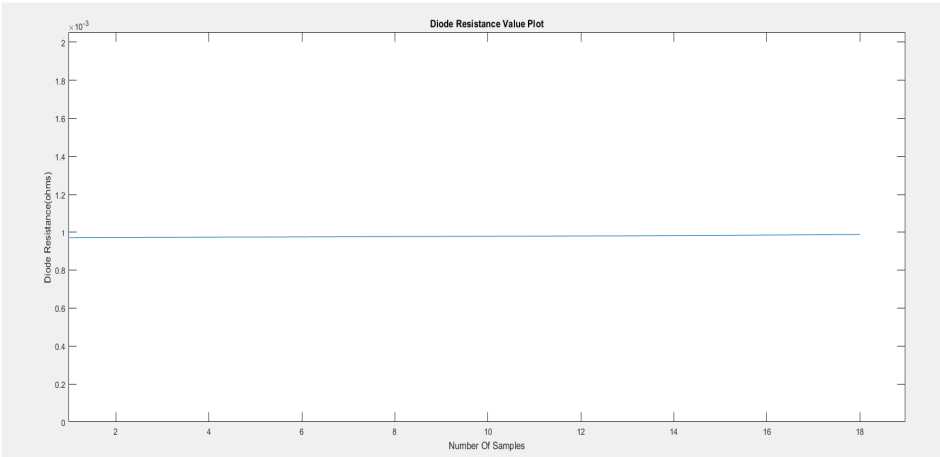


Figure 5.66. The Diode Resistance Value Plot

5.7.3.4 DC/DC BUCK converter. White-box. Parameter identification based on online tests. Transient part Steady-state during T_{off}

DC/DC converters usually include a closed loop based on a PID controller to regulate and stabilize the output voltage V_{out} according to the reference value V_{ref}.

The control loop is estimated during the transient state. The transients can be added in the following ways to the circuit

- Adding a random noise in the input voltage
- Switching on and off a resistance in the load side

To perform the identification, a load resistance *R*, which is switched on and off is used to generate transients from the load side, as shown in next figure.

In the closed loop converter there is a control loop in which the V_{out} is fedback to the control loop. In a closed loop control, **the duty cycle is varied dynamically,** so V_{out} changes correspondingly.

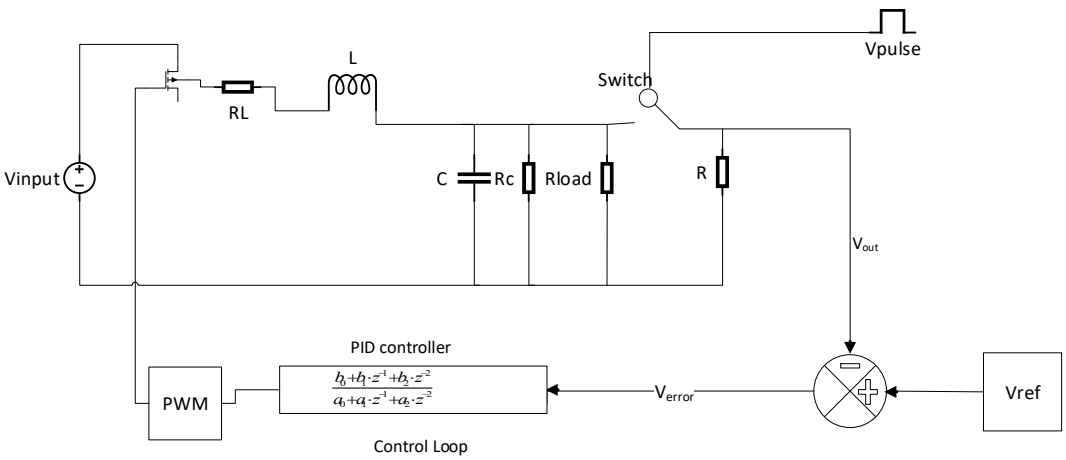


Figure 5.67. Closed loop DC DC Buck converter

A transfer function given by the ratio V_{error}/D is evaluated, $D = T_{on}/(T_{on}+T_{off})$ being the duty cycle and $V_{error} = V_{ref} - V_{out}$.

The discrete transfer function of the PID controller can be expressed as $K_P + K_I \cdot \frac{1}{1-z^{-1}} + K_D \cdot (1-z^{-1})$, K_P being the proportional component, K_I the integral component and K_D the derivative component and z^{-1} a time-delay. Therefore, the control loop is represented as an equivalent transfer function with two zeros (numerator) and two poles (denominator) [36] through the pole placement approach[37],

$$H(z) = \frac{D(z)}{V_{error}(z)} = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{(1-z^{-1})\cdot(1+a_1 \cdot z^{-1})}$$

Where b_0, b_1, b_2 are the coefficients of the poles, and a_1 is the coefficient of zeros, which are identified by means of the ***tfest*** function of Matlab [38].

The V_{out} plot comparing the initial transient and the fitting with the ***tfest*** Matlab function,

[36] J. B. V.Bobal, J.Fessl, and J.Machacek, Digital Self-tuning Controllers: Algorithms, Implementation and Applications: Springer-Verlag London limited 2005.

[37] M. M. Fawzi Saber Algreer, Microprocessor Based Signal Processing Techniques for System Identification and Adaptive Control of DC-DC Converters, PhD Thesis, Newcastle University, 2012

[38] Mohamed Ahmeid, Matthew Armstrong,Real-Time Parameter Estimation of DC–DC Converters Using a Self-Tuned Kalman Filter, IEEE Trans. on Power Electronics, vol. 32, no. 7, July 2017

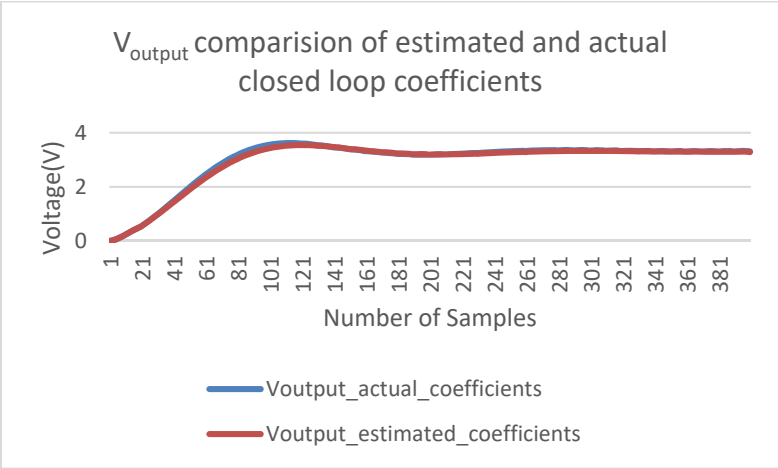


Figure 5.68. V_{output} Comparison of estimated and actual closed loop coefficients

Comparison of actual and estimated closed loop coefficients

Parameters	Actual	Estimated by *tfest*
b_0	4.672	4.470
b_1	-7.539	-7.415
b_2	3.184	3.120
a_1	-0.374	-0.382

5.8 Parameter identification in DC/DC BOOST converters.

5.8.1 DC/DC BOOST converters. Parameter identification based on manufacturers' data

A **boost converter** (step-up converter) steps up the voltage (or steps down the current) from the input (supply) to the output (load).

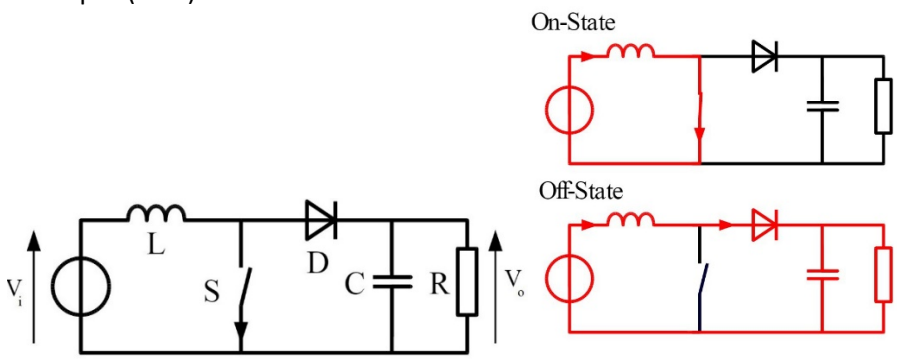


Figure 5.69. DC/CD BOOST converter

For the boost converter the manufacturer specifies V_{input}, V_{output}, switching frequency (F) and I_{output}.. The manufacturer will keep the parameters value confidential and will not disclose the values. From the datasheet[39], they have given the range of L and C values to be used for different V_{input},V_{output} values.

V_{input}	2.3 to 5.5 V
V_{output}	5 V
I_{output}	4 A
F	3.8 Mhz
L	0.3 to 1.3 uH
C	3.5 to 30 uF

For Calculating the L and C some assumptions can be made and with that the minimum value of L and C for the circuit can be calculated[40].

The output current ripple is around the 30% of the maximum I_{output} and the ripple voltage is 2 % of output voltage

From Simulation the current ripple is around the 20% of the maximum I_{output} and the ripple voltage is 1.3 % of output voltage

During T_{on}

$$D = 1 - \frac{V_{input}}{V_{output}} \qquad (1)$$

$$T_{on} = DT = D / f$$

$$V_L = V_{input}$$

$$L \cdot \frac{dI_L}{dt} = V_{input}$$

$$L = \frac{V_{input} \cdot T_{on}}{\Delta I} \qquad (2)$$

$$C \cdot \frac{dV_c}{dt} = I_{output}$$

$$C = \frac{I_{output} \cdot T_{on}}{\Delta V_{out}} \qquad (3)$$

Parameters	Simulation data	Manufacturers data
L(H)	0.4085 uH	0.59 uH
C(F)	8.74 uF	6 uF

[39] http://www.ti.com/lit/ds/symlink/tps61253a.pdf

[40] http://www.ti.com/lit/an/slva372c/slva372c.pdf

Therefore it is concluded that from manufacturer's data only a rough estimation of the parameters is possible.

5.8.2 DC/DC BOOST converters. Parameter identification based on offline tests.

It is not advisable to conduct open circuit tests or short circuit test like in the case of the buck converter DC/DC BOOST converters. Parameter identification based on offline tests. Steady-state during Ton

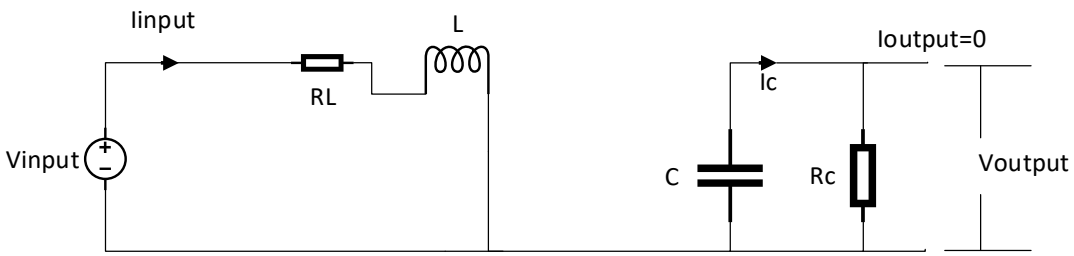


Figure 5.70. Boost Converter Ton with open load

The parametric estimation is not possible during T_{on} as with open circuit and short circuit test

5.8.2.1 DC/DC BOOST converters. Parameter identification based on offline tests. Steady-state during Toff

The open load test can be conducted and the equations for calculating the C, R_c,L and will be the changed as below. During T_{off} the R_L and R_d cannot be calculated individually and the combination of R_L+R_d can be estimated

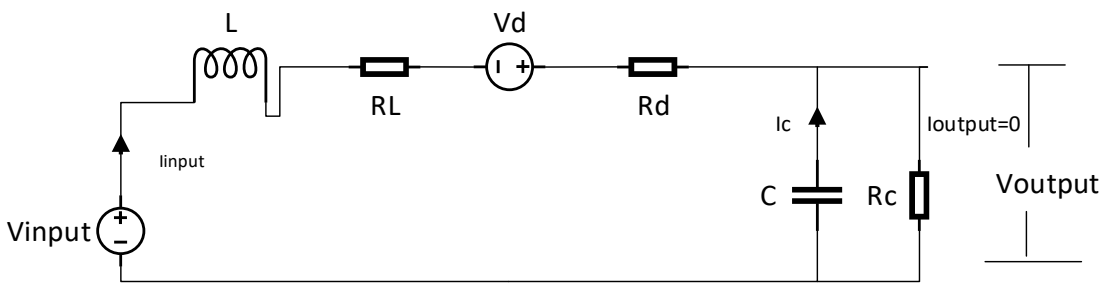


Figure 5.71. Boost Converter during Toff with Open load

The equations describing the T_{off} interval are as follows,

$$I_{input} = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C} \rightarrow C\frac{dV_{output}}{dt} = I_{input} - \frac{V_{output}}{R_C}$$

$$C\int dV_{output} = \int (I_{input} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input} - \frac{V_{output}}{R_c})dt$$

Open loop test on Boost Converter during T_{off}

Parameters	Simulated values	Calculated values
C	200 uF	200 uF
Rc	100 ohms	100 ohms

L	250 uH	250 uH
R_L+R_d	1.06 ohms	1.06 ohms

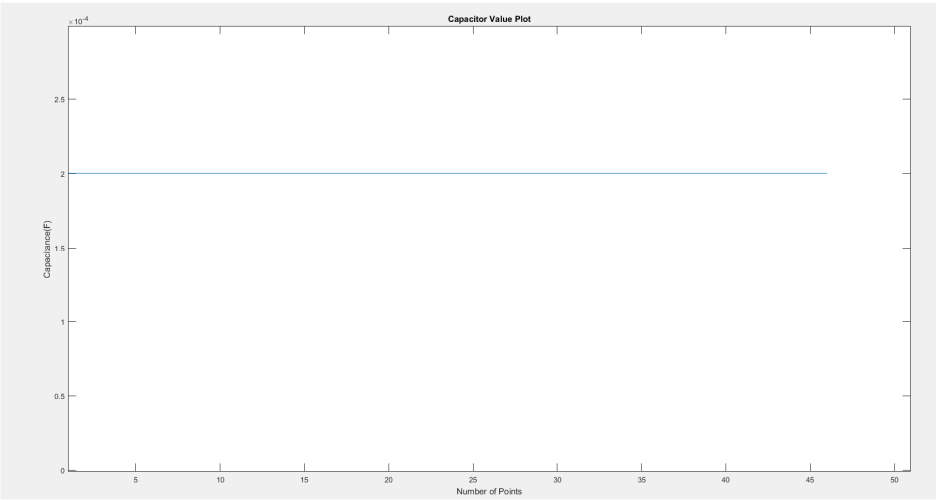


Figure 5.72. Capacitance Value Plot

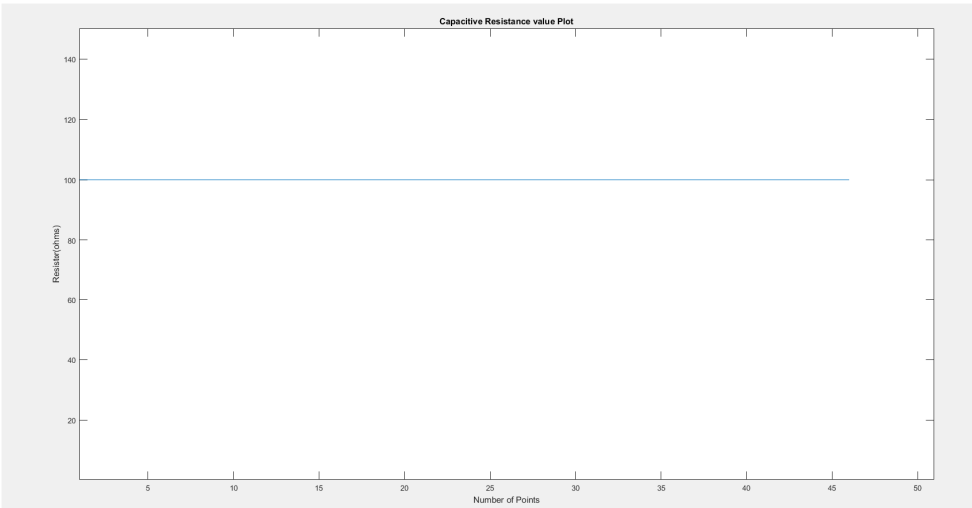


Figure 5.73. Capacitor resistance value plot

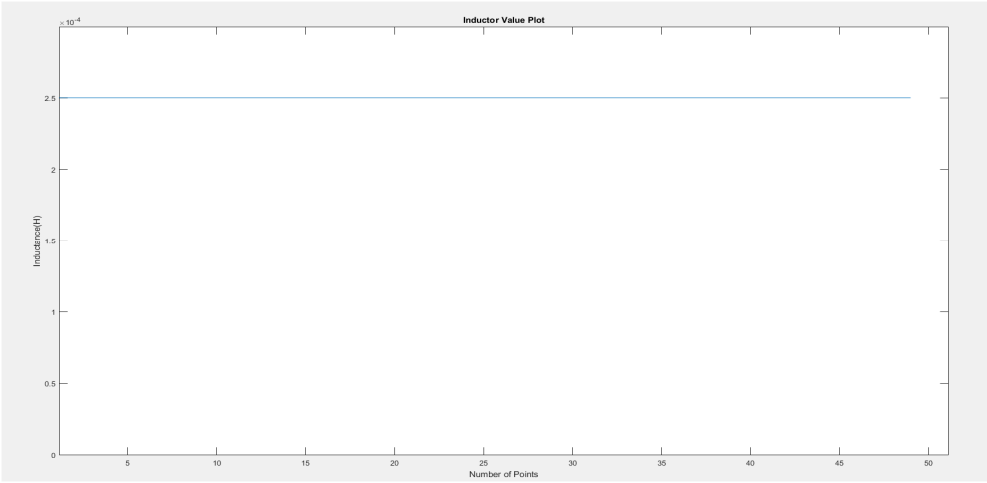


Figure 5.74. Inductance value plot

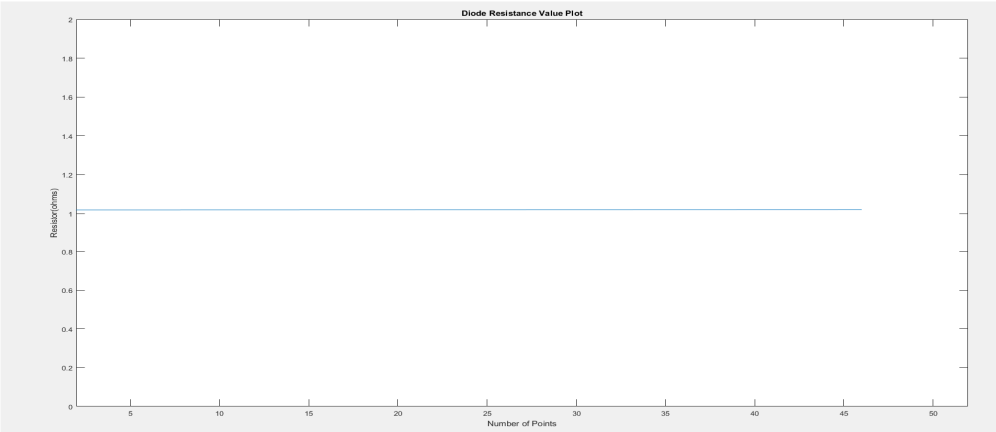


Figure 5.75. Combination of (R_d+R_L) resistance value plot

5.8.2.2 DC/DC BOOST converters. Parameter identification based on offline tests. Transient state during both Ton and Toff

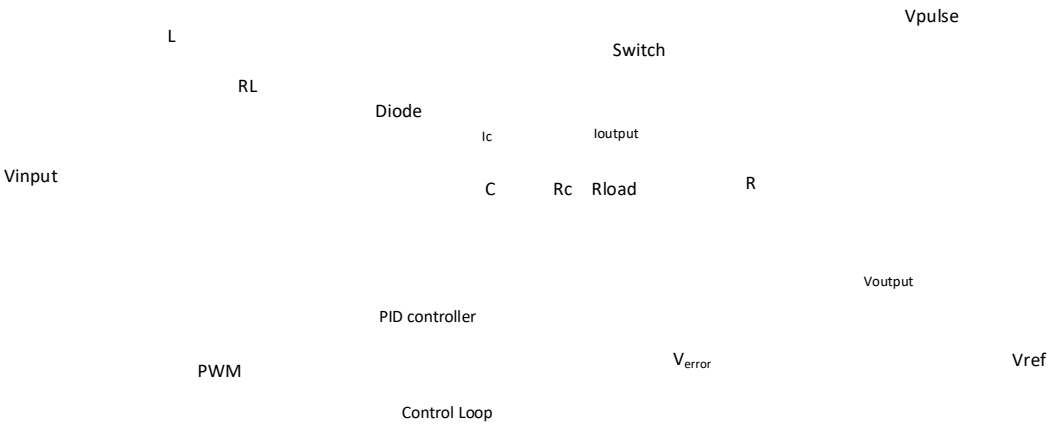


Figure 5.76. The closed loop boost converter

DC/DC converters usually include a closed loop based on a PID controller to regulate and stabilize the output voltage V_{out} according to the reference value V_{ref}.

The control loop is estimated during the transient state. The transients can be added in the following ways to the circuit

- Adding a random noise in the input voltage
- Switching on and off a resistance in the load side

To perform the identification, a load resistance *R*, which is switched on and off is used to generate transients from the load side, as shown in next figure.

For doing offline test either open loop (no load) is considered.

Considering the open loop test the transients has to be created by adding a noise in the input voltage,but the excitation from the input voltage does not produce rich transients to calculate the parameters of the control loop, Hence another approach needs to be used.

The next approach is using a variable load at the load side, but R_{load} cannot be opened as it gives rich transients to calculate the closed loop parameters and with only variable load, there is a high risk the converter will be blown due to high inductance current during switching on and off, hence there is always a constant load required in the circuit. One more possible approach instead of open loop the transfer function of the closed loop can be calculated for different values of R_{load}.

5.8.3 DC/DC BOOST converter. White-box. Parameter identification based on online tests during the transient part is done with different R_{load} .

The parameter identification based on online tests during the transient part is done with different R_{load}. The procedure for calculating the closed loop coefficients is explained in detail in the online method section.

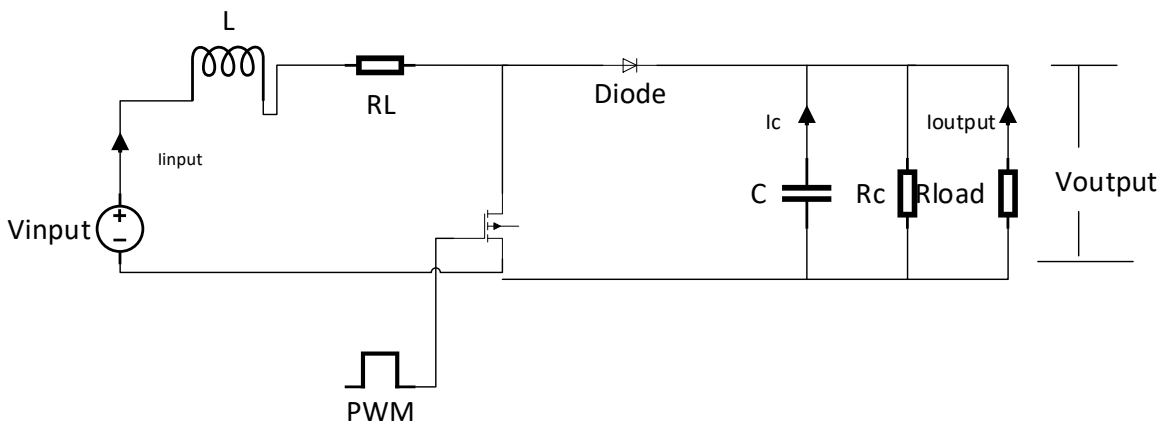


Figure 5.77. Schematic diagram of the BOOST converter

5.8.3.1 DC/DC BOOST converter. White-box. Parameter identification based on online tests. Steady-state during Ton

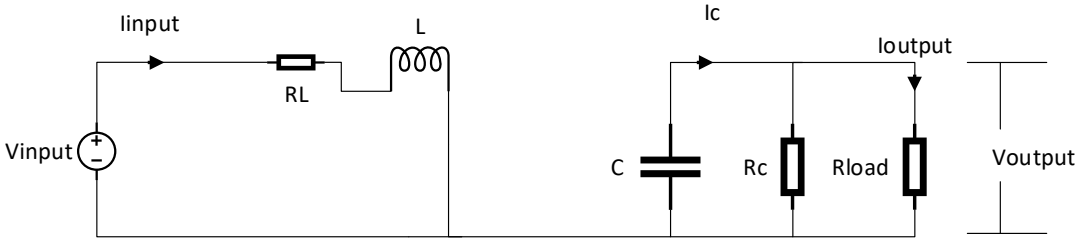


Figure 5.78. Boost converter during Ton with parasitic elements

$$V_{input} - i_{input} \cdot R_L - L \cdot \frac{di_{input}}{dt} = 0$$

$$V_{input} - i_{input} \cdot R_L = L \cdot \frac{di_{input}}{dt}$$

$$\int V_{input} \cdot dt = L \cdot \int di_{input} + R_L \cdot \int i_{input} \cdot dt$$

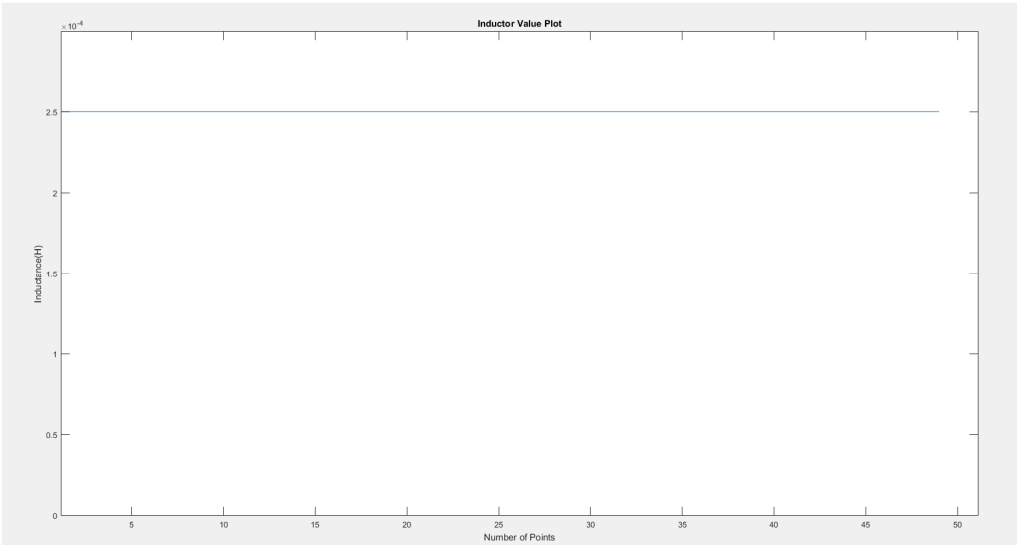


Figure 5.79. Inductance value plot

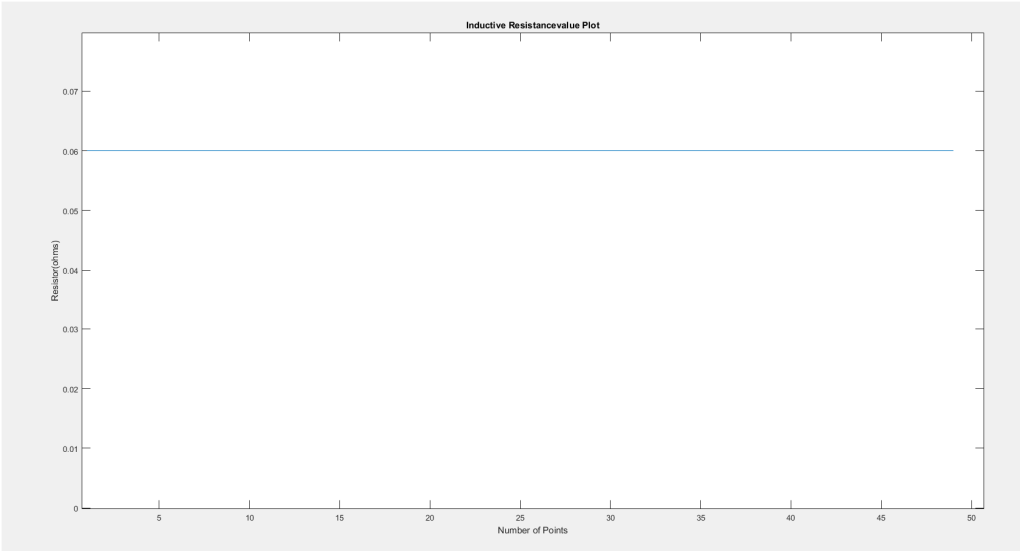


Figure 5.80. Inductor resistance Value Plot

5.8.3.2 DC/DC BOOST converter. White-box. Parameter identification based on online tests. Steady-state during Toff

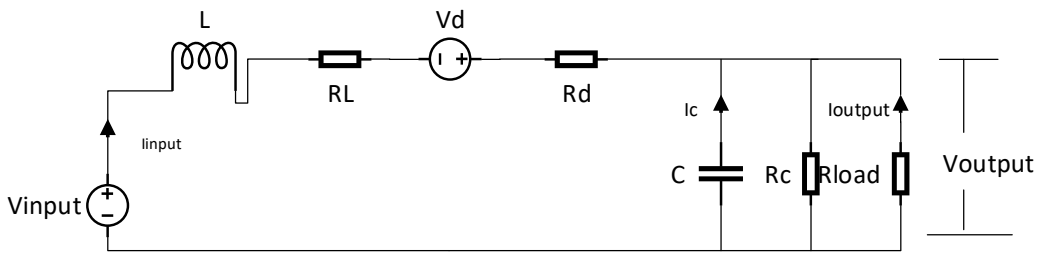


Figure 5.81. Boost Converter during T_{off}

The equations describing the T_{off} interval are as follows,

$$I_{input} = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C} + I_{output} \rightarrow C\frac{dV_{output}}{dt} = I_{input} - \frac{V_{output}}{R_C} - I_{output}$$

$$C\int dV_{output} = \int (I_{input} - I_{output} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input} - I_{output} - \frac{V_{output}}{R_c})dt$$

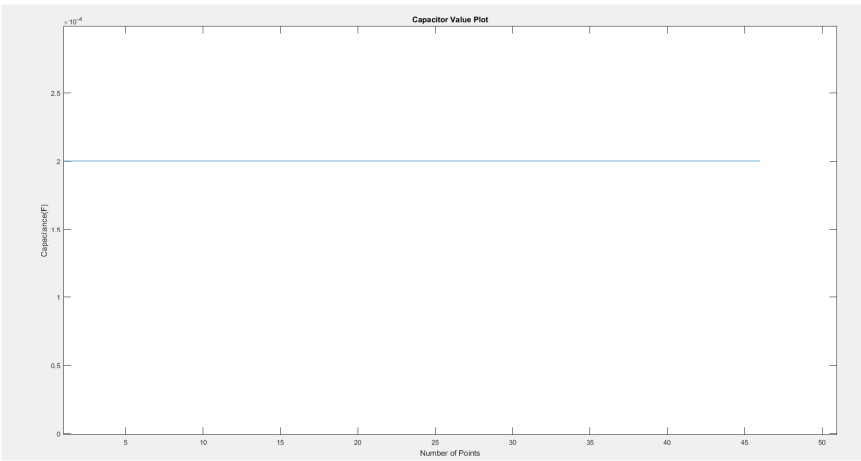


Figure 5.82. Capacitor Value Plot

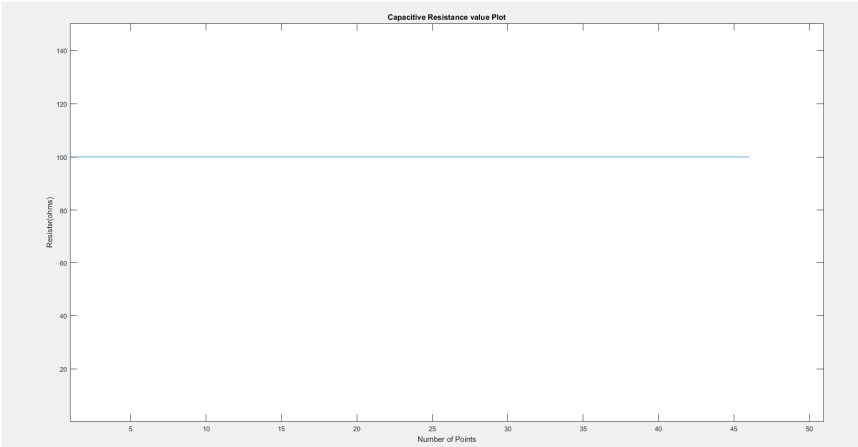


Figure 5.83. Capacitive resistance Value Plot

Parameter	Simulated value	Calculated value
L	250 uH	250 uH
R_L	0.06 ohms	0.06 ohms
C	200 uF	200 uF
R_C	100 ohms	100 ohms
R_d	1 ohm	1 ohm

Table 7. Parameters at steady state boost converter

Calculating the diode resistance:

$$V_{input} - V_d - V_{output} = I_{input} \cdot (R_d + R_L) + L \cdot \frac{dI_{input}}{dt} \rightarrow L \cdot \frac{dI_{input}}{dt} = -V_d - V_{output} - I_{input} \cdot (R_d + R_L)$$

$$L \cdot \int dI_{input} = \int V_{input} \cdot dt - \int V_d \cdot dt - \int V_{output} \cdot dt - (R_d + R_L) \int I_{input}$$

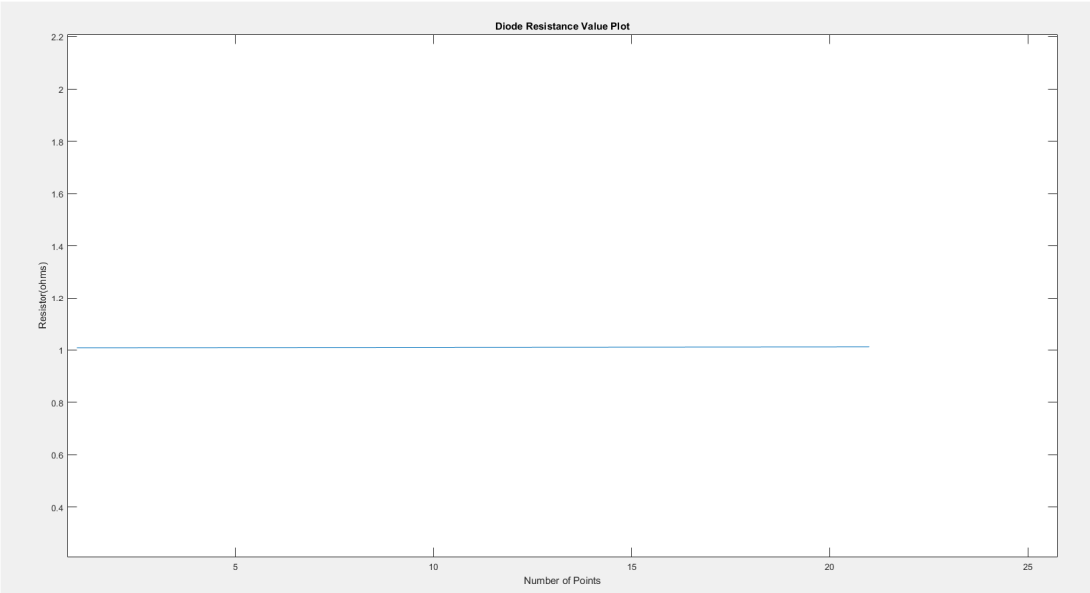


Figure 5.84. The Diode resistance plot

5.8.3.3 DC/DC BOOST converter. White-box. Parameter identification based on online tests. Transient State during Ton

DC/DC converters usually include a closed loop based on a PID controller to regulate and stabilize the output voltage V_{out} according to the reference value V_{ref}.

The control loop is estimated during the transient state. The transients can be added in the following ways to the circuit

- Adding a random noise in the input voltage
- Switching on and off a resistance in the load side

To perform the identification, a load resistance *R*, which is switched on and off is used to generate transients from the load side, as shown in next figure.

In the closed loop converter there is a control loop in which the V_{out} is fedback to the control loop. In a closed loop control, **the duty cycle is varied dynamically,** so V_{out} changes correspondingly.

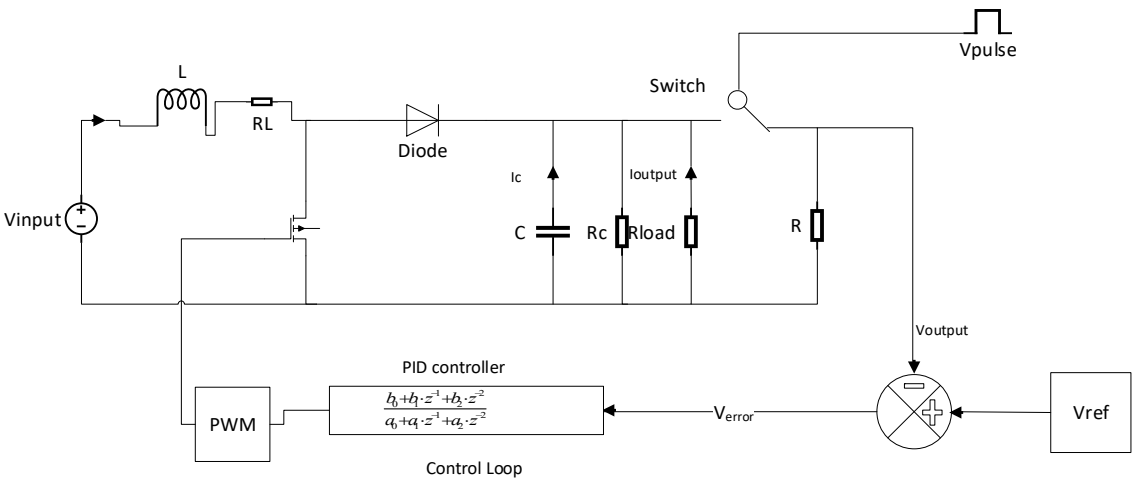


Figure 5.85. Closed loop DC DC Boost converter

A transfer function given by the ratio V_{error}/D is evaluated, $D = T_{on}/(T_{on}+T_{off})$ being the duty cycle and $V_{error} = V_{ref} - V_{out}$.

The discrete transfer function of the PID controller can be expressed as $K_P + K_I \cdot \frac{1}{1-z^{-1}} + K_D \cdot (1-z^{-1})$, K_P being the proportional component, K_I the integral component and K_D the derivative component. Therefore, the control loop is the represented as an equivalent transfer function with two zeros (numerator) and one pole (denominator) [41] through the pole placement approach[42],

$$H(z) = \frac{D(z)}{V_{error}(z)} = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{(1-z^{-1}) \cdot (1-a_1 z^{-1})}$$

Where b_0, b_1, b_2 are the coefficients of the poles, and a_1 is the coefficient of zeros, which are identified by means of the ***tfest*** function of Matlab [43].

[41] J. B. V.Bobal, J.Fessl, and J.Machacek, Digital Self-tuning Controllers: Algorithms, Implementation and Applications: Springer-Verlag London limited 2005.

[42] M. M. Fawzi Saber Algreer, Microprocessor Based Signal Processing Techniques for System Identification and Adaptive Control of DC-DC Converters, PhD Thesis, Newcastle University, 2012

[43] Mohamed Ahmeid, Matthew Armstrong,Real-Time Parameter Estimation of DC–DC Converters Using a Self-Tuned Kalman Filter, IEEE Trans. on Power Electronics, vol. 32, no. 7, July 2017

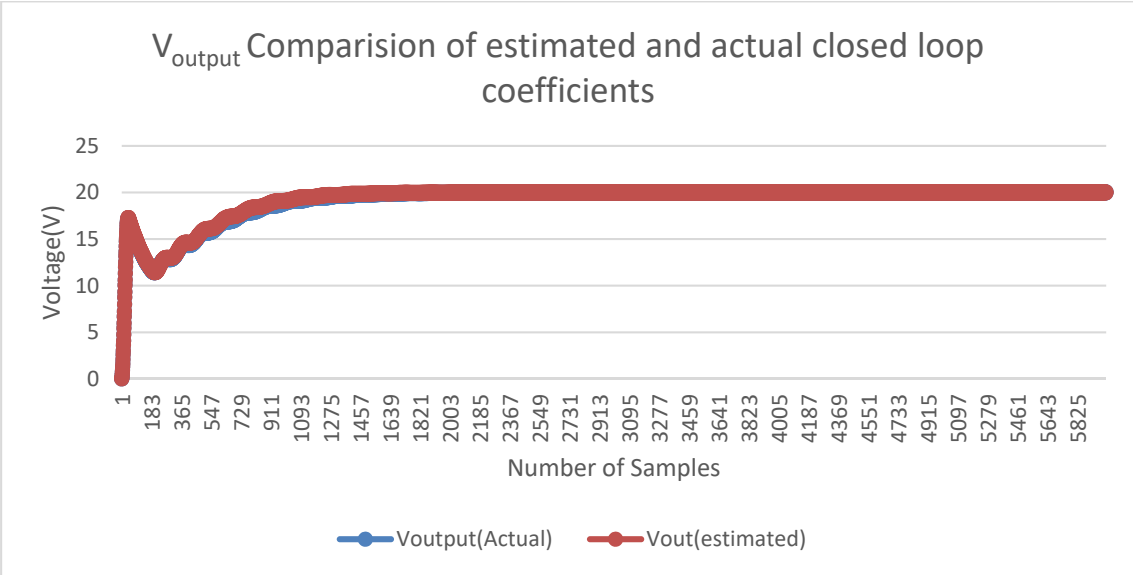


Figure 5.86. V_{output} comparison of estimated and actual closed loop coefficients

Comparison of estimated and actual closed loop coefficients

Parameters	Actual	Estimated by *tfest*
b_0	0.174	0.1738
b_1	-0.3379	-0.3416
b_2	0.164	0.1679
a_1	0	0

The V_{output} graph obtained for the boost converter is not very smooth compared to the graph obtained in the buck and buck boost converters. The reason for this is due to the value of L and C chosen. In this scenario L and C values are 150 uH and 200 uF which are high values used for the testing. For lower values of L and C the V_{output} can be smoother.

5.9 Parameter identification in DC/DC BUCK-BOOST converter

5.9.1 DC/DC BUCK-BOOST converter. Parameter identification based on manufacturers' Data

The buck–boost converter has an output voltage magnitude that can be either less than or greater than the magnitude of the input voltage.

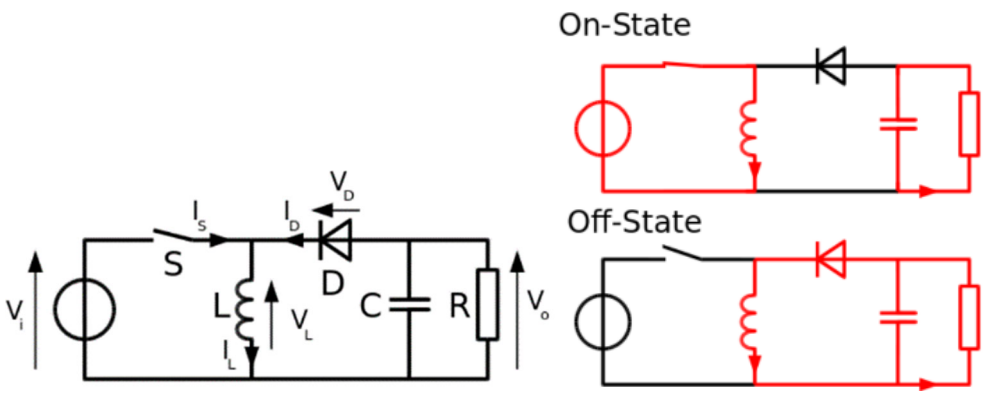


Fig. 5.87. BC/DC BUCK-BOOST converter

For the buck-boost converter the manufacturer will gives the values of V_{input}, V_{output}, switching frequency (F) and I_{output}.

The manufacturer will keep the parameters value confidential and will not disclose the values.

From the datasheet[44], they have given the range of L and C values to be used for different V_{input},V_{output} values.

V_{input}	2.3 to 5.5 V
V_{output}	2.5 to 3.6 V
I_{output}	1.5 A
F	2.5 Mhz
L	0.5 to 1.3 uH
C	10-20 uF

Assuming the output ripple current is 30% of I_{output} and the output ripple voltage is 2% of output Voltage

During T_{on}

$$D = \frac{-V_{output}}{V_{output} - V_{input}} \quad (1)$$

$$T_{on} = DT$$

$$V_L = V_{input}$$

$$L \cdot \frac{dI_L}{dt} = V_{input}$$

[44] http://www.ti.com/lit/ds/symlink/tps63024.pdf

$$L = \frac{V_{input} \cdot T_{on}}{\Delta I} \qquad (2)$$

$$C \cdot \frac{dV_c}{dt} = I_{output}$$
$$C = \frac{I_{output} \cdot T_{on}}{\Delta V_{out}} \qquad (3)$$

Parameters	Simulation data	Manufacturers data
L(H)	0.7482uH	0.747uH
C(F)	11.76uF	10uF

Therefore, it is possible to only estimate the L and C values from the manufacturers' data

5.9.2 DC/DC BUCK-BOOST converters. Parameter identification based on offline tests.

Since the Buck Boost will be a two different circuits during T_{on}. It is not advisable to conduct open circuit tests or short circuit test like the buck converter, Instead the R_{Load} can be varied, the converter can be connected with different loads and the parameters can be estimated.

5.9.2.1 DC/DC BUCK-BOOST converters. Parameter identification based on offline tests. Steady-state during Ton

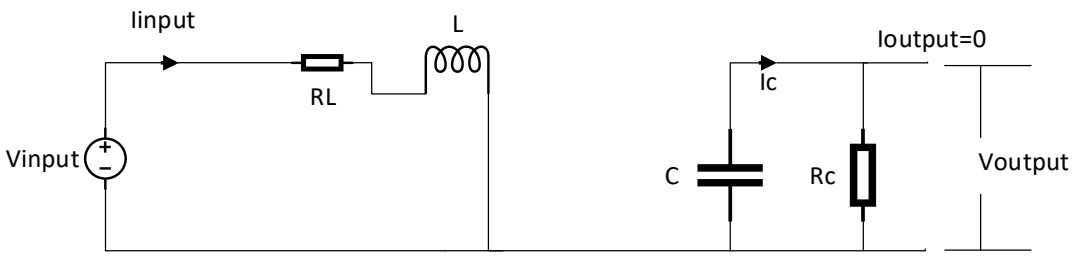


Figure 5.88. Buck-Boost Converter during T_{on} for Open load

Since during T_{on} the open load test and the short circuit test cannot be conducted and it will not be useful for calculating the parameters during T_{on}

5.9.2.1 DC/DC BUCK-BOOST converters. Parameter identification based on offline tests. Steady-state during Toff

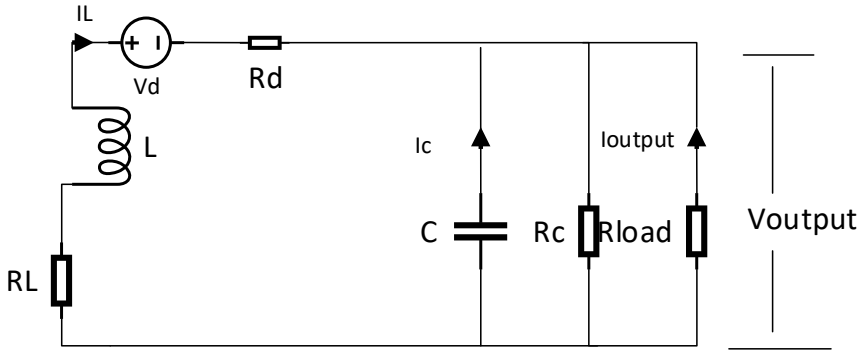


Figure 5.89. Buck-Boost Converter during T_{off} for open load

$$I_L = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C} + I_{output} \rightarrow C\frac{dV_{output}}{dt} = I_L - \frac{V_{output}}{R_C} - I_{output}$$

$$C\int dV_{output} = \int (I_L - I_{output} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_L - I_{output} - \frac{V_{output}}{R_c})dt$$

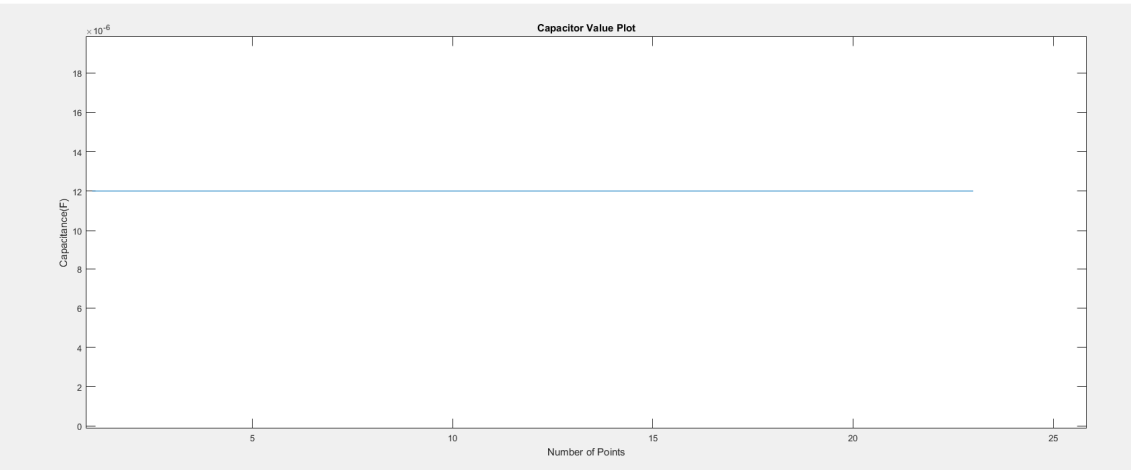


Figure 5.90. The Capacitance Value Plot

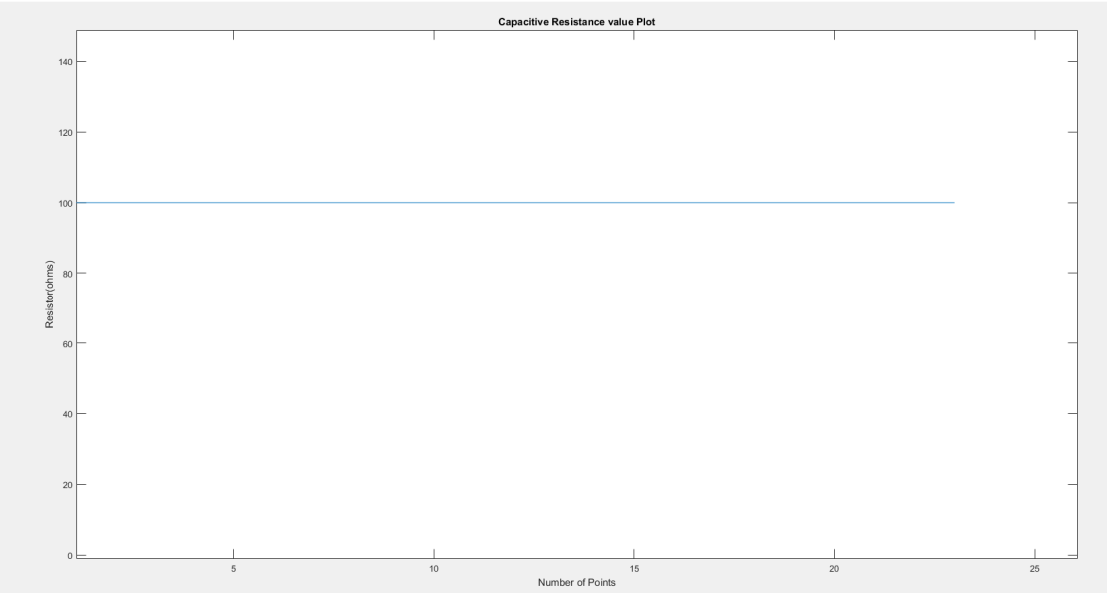


Figure 5.91. The Capacitor Resistance Value Plot

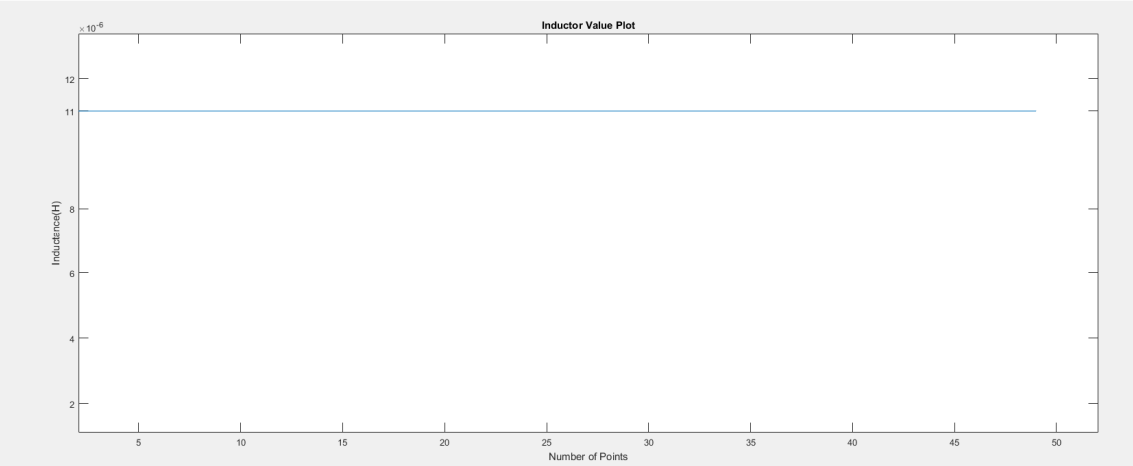


Figure 5.92. The Inductance Value Plot

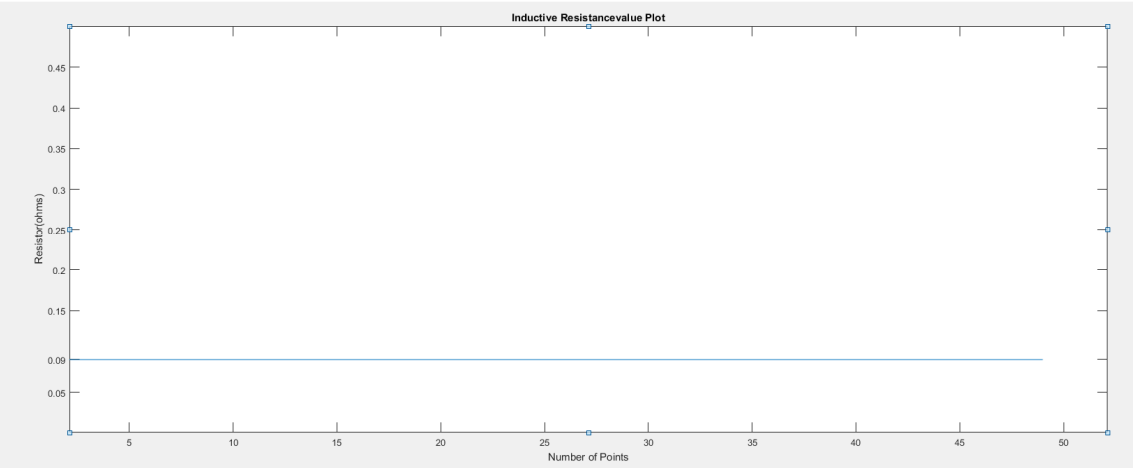


Figure 5.93. The Inductor Resistance and Diode Resistance Combined

5.9.2.2 DC/DC BUCK-BOOST converters. Parameter identification based on offline tests. Transient state

The excitation from the input voltage does not produce rich transients to calculate the parameters of the control loop, Hence another approach needs to be used.

We connect a different constant load and a variable load for this test. As with only variable load, there is a high risk the converter will be blown due to high inductance current during switching on and off.

Hence, we follow the same approach as we did in Parameter identification during the transient part but the constant load will be a different value than the constant load used in that case

5.9.3 DC/DC BUCK-BOOST converter. White-box. Parameter identification based on online tests

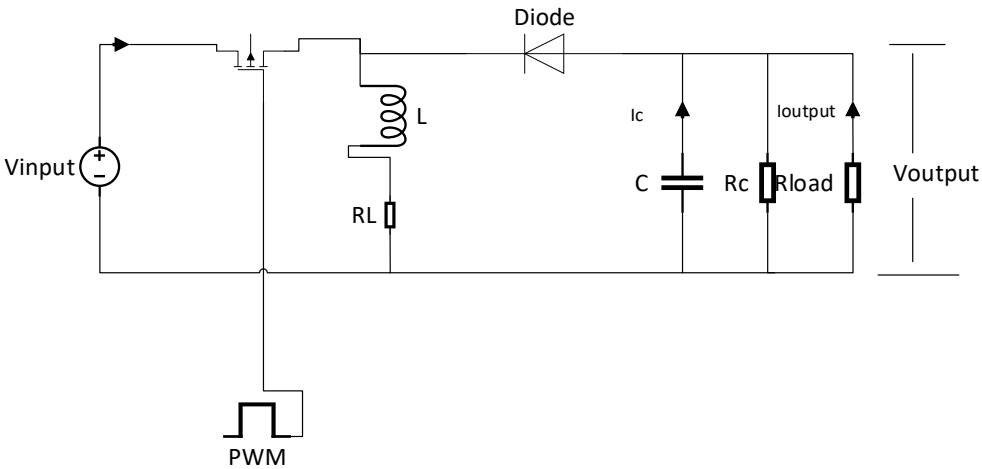


Figure 5.94. Schematic Diagram of Buck-boost converter

5.9.3.1 DC/DC BUCK-BOOST converter. White-box. Parameter identification based on online tests. Steady-state during Ton

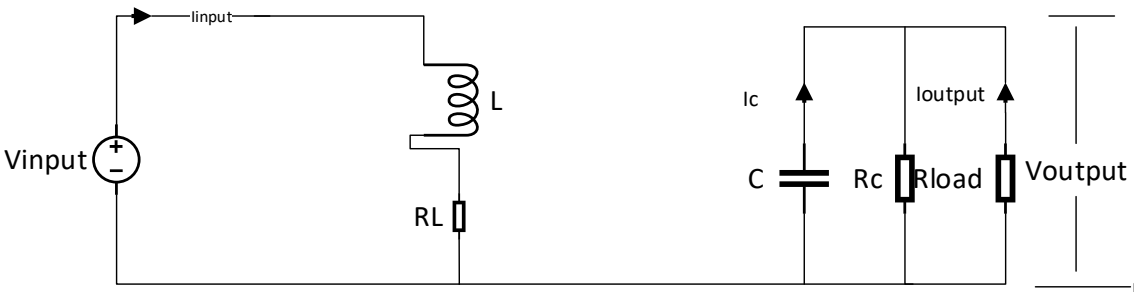


Figure 5.95. Buck boost converter during T_{on}

$$V_{input} - i_{input} \cdot R_L - L \cdot \frac{di_{input}}{dt} = 0$$

$$V_{input} - i_{input} \cdot R_L = L \cdot \frac{di_{input}}{dt}$$

$$\int V_{input} \cdot dt = L \cdot \int di_{input} + R_L \cdot \int i_{input} \cdot dt$$

5.9.3.2 DC/DC BUCK-BOOST converter. White-box. Parameter identification based on online tests. Steady-state during Toff

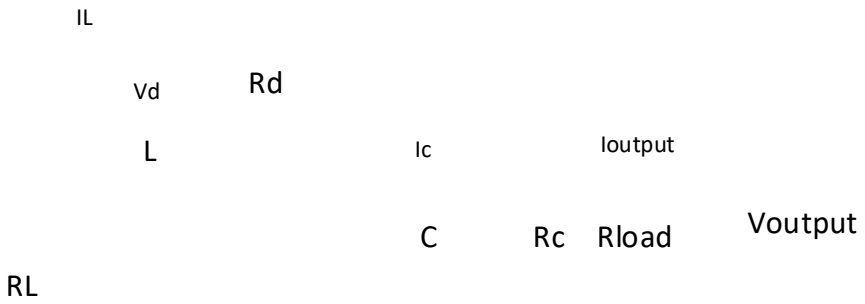


Figure 5.96. Buck boost converter during T_{off}

The value of C and R_c is calculated during the T_{off}. As during the T_{off} the inductor will discharge and this will act as a source for the circuit. The value of inductor current (I_L) is estimated as the I_L during the T_{on} is known and the data is available.

We consider two periods and as explained in the buck converter I_L calculation two points are taken as mentioned during the T_{off} ,the current will not be linear as the inductor is discharging which is observed in the simulation.

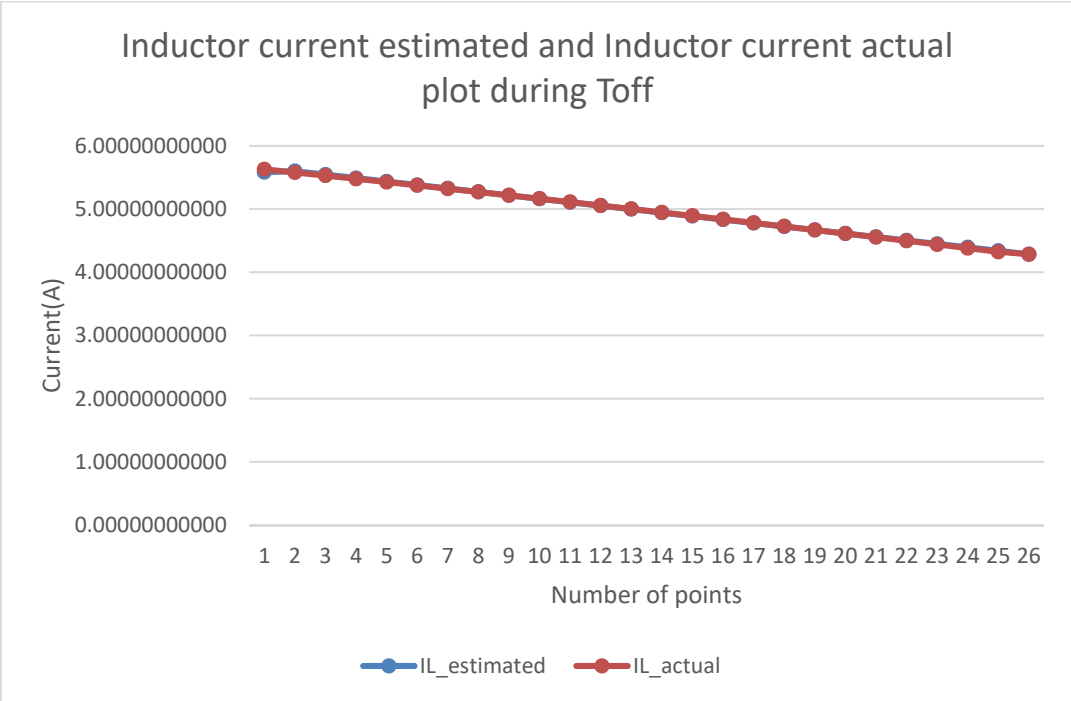


Fig 5.97. Inductor current estimated and Inductor current actual during T_{off}

The equations describing the T_{off} interval are as follows,

$$I_L = C\frac{dV_{output}}{dt} + \frac{V_{output}}{R_C} + I_{output} \rightarrow C\frac{dV_{output}}{dt} = I_L - \frac{V_{output}}{R_C} - I_{output}$$

$$C\int dV_{output} = \int (I_L - I_{output} - \frac{V_{output}}{R_c})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_L - I_{output} - \frac{V_{output}}{R_c})dt$$

5.9.3.3 DC/DC BUCK-BOOST converter. White-box. Parameter identification based on online tests. Steady-state during Ton

The parameters of Buck-Boost Converter

Parameters	Simulated value	Calculated value
L	11 uH	11 uH
C	12 uF	12 uF
R_L	0.08 ohms	0.08 ohms
R_C	100 ohms	100 ohms
Rd	0.01ohms	0.01ohms

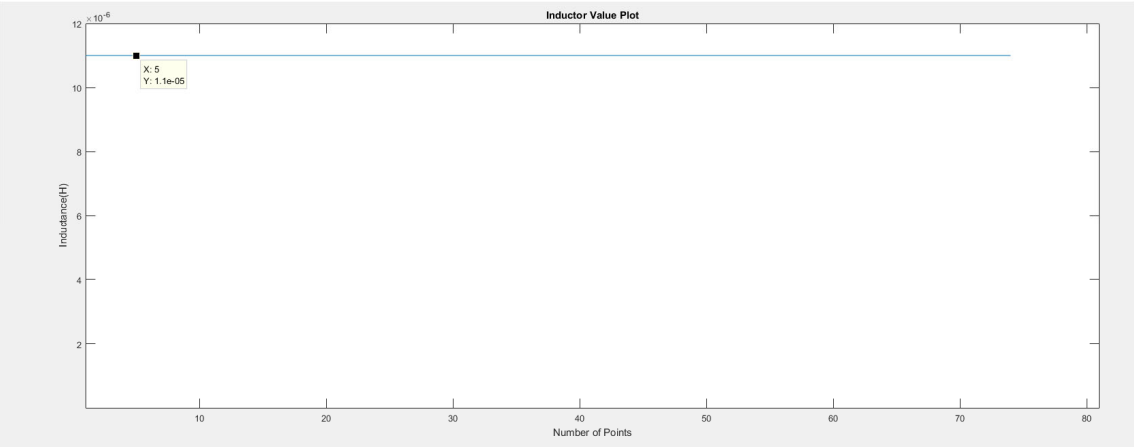


Figure 5.98. The Inductance Value Plot

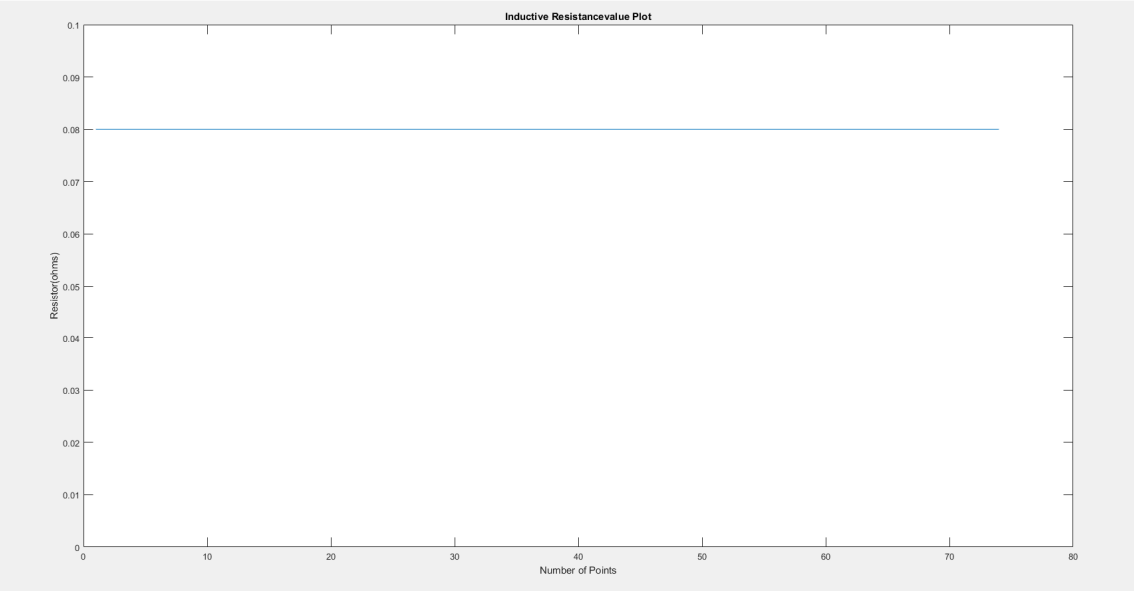


Figure 5.99. The Inductor Resistance Value Plot

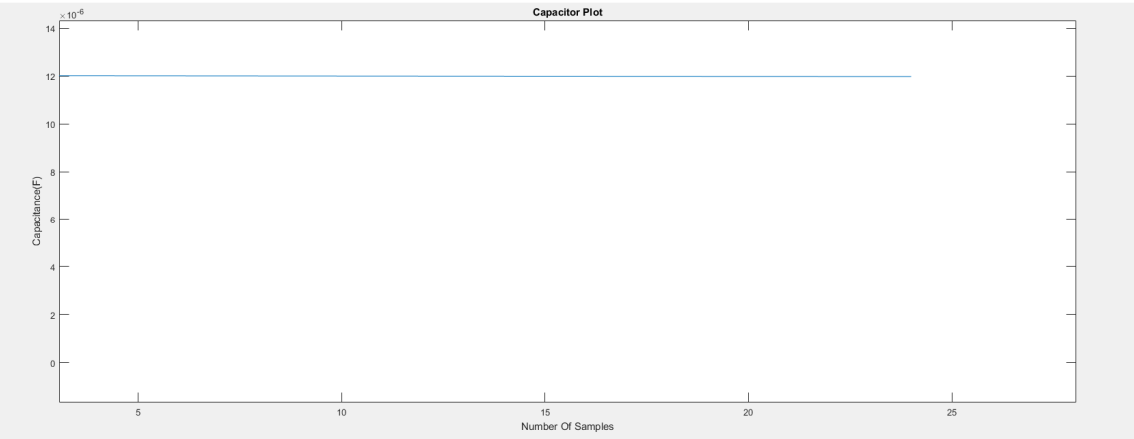


Figure 5.100. The Capacitor Value Plot

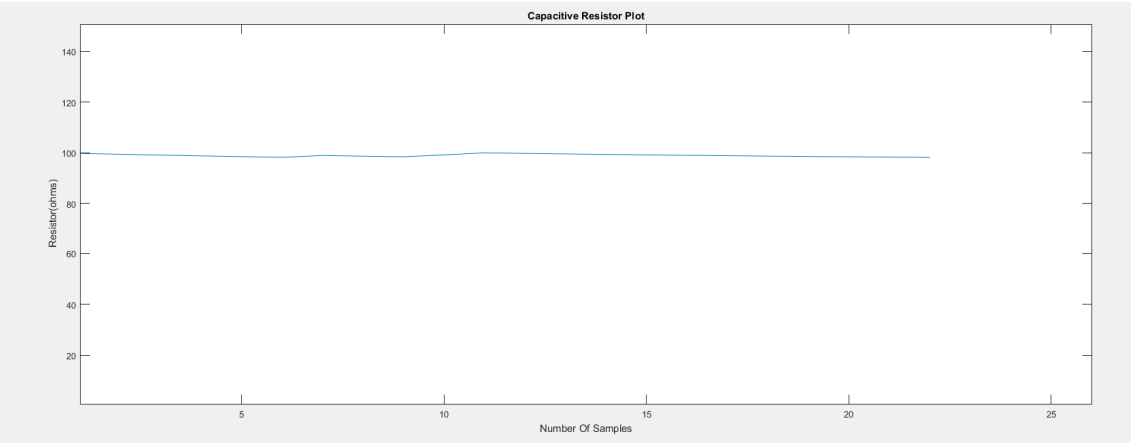


Figure 5.101. The Capacitor Resistance Value Plot

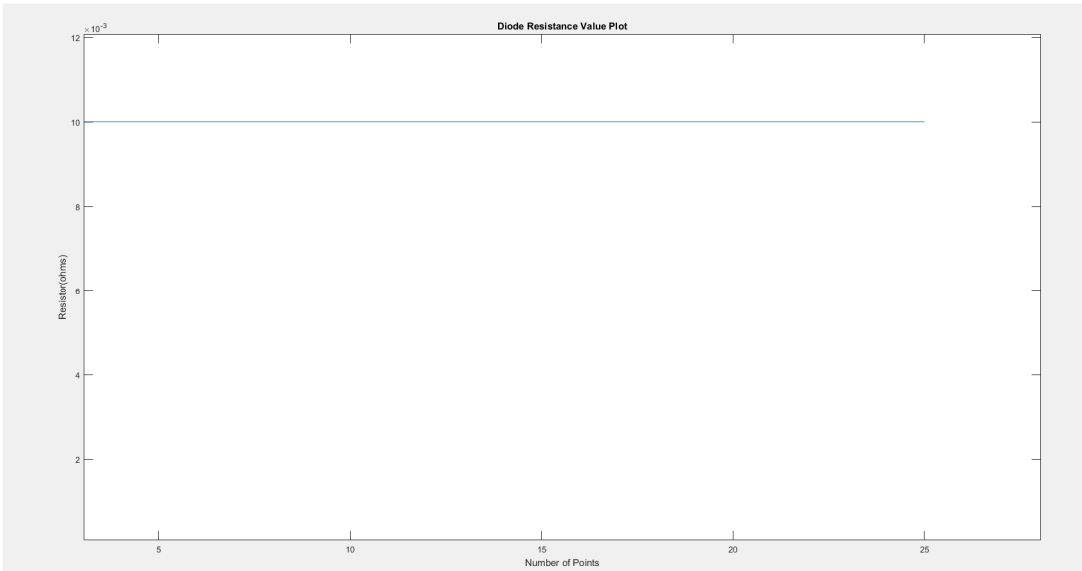


Figure 5.102. Diode Resistance Plot

5.9.3.4 DC/DC BUCK-BOOST converter. White-box. Parameter identification based on online tests. Transient part

DC/DC converters usually include a closed loop based on a PID controller to regulate and stabilize the output voltage V_{out} according to the reference value V_{ref}.

The control loop is estimated during the transient state. The transients can be added in the following ways to the circuit

- Adding a random noise in the input voltage
- Switching on and off a resistance in the load side

To perform the identification, a load resistance R, which is switched on and off is used to generate transients from the load side, as shown in next figure.

In the closed loop converter there is a control loop in which the V_{out} is fedback to the control loop. In a closed loop control, **the duty cycle is varied dynamically,** so V_{out} changes correspondingly.

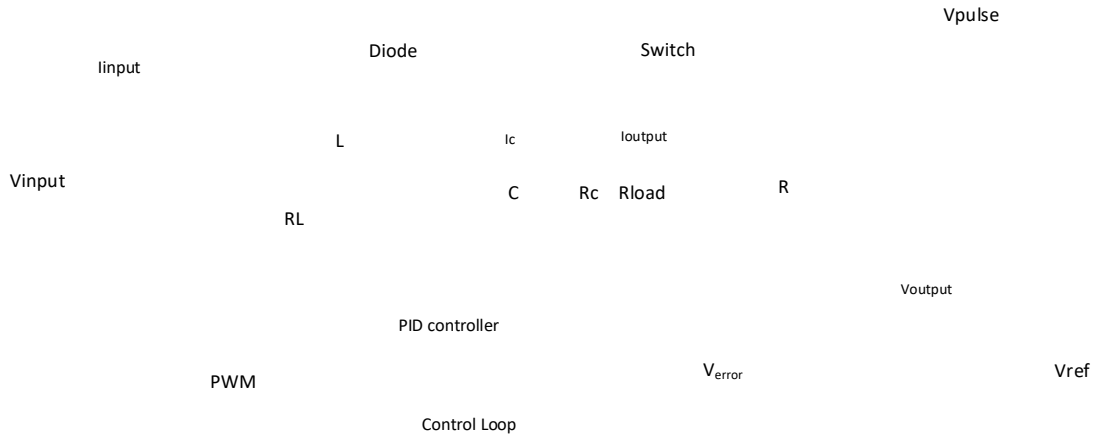


Figure 5.103. Closed loop DC DC Buck-Boost converter

A transfer function given by the ratio V_{error}/D is evaluated, $D = T_{on}/(T_{on}+T_{off})$ being the duty cycle and $V_{error} = V_{ref} - V_{out}$.

The discrete transfer function of the PID controller can be expressed as $K_P + K_I \cdot \frac{1}{1-z^{-1}} + K_D \cdot (1-z^{-1})$, K_P being the proportional component, K_I the integral component and K_D the derivative component. Therefore, the control loop is the represented as an equivalent transfer function with two zeros (numerator) and one pole (denominator) through the pole placement approach,

$$H(z) = \frac{D(z)}{V_{error}(z)} = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{(1-z^{-1}) \cdot (1 + a_1 \cdot z^{-1})}$$

Where b_0, b_1, b_2 are the coefficients of the poles, and a_1 is the coefficient of zeros, which are identified by means of the ***tfest*** function of Matlab.

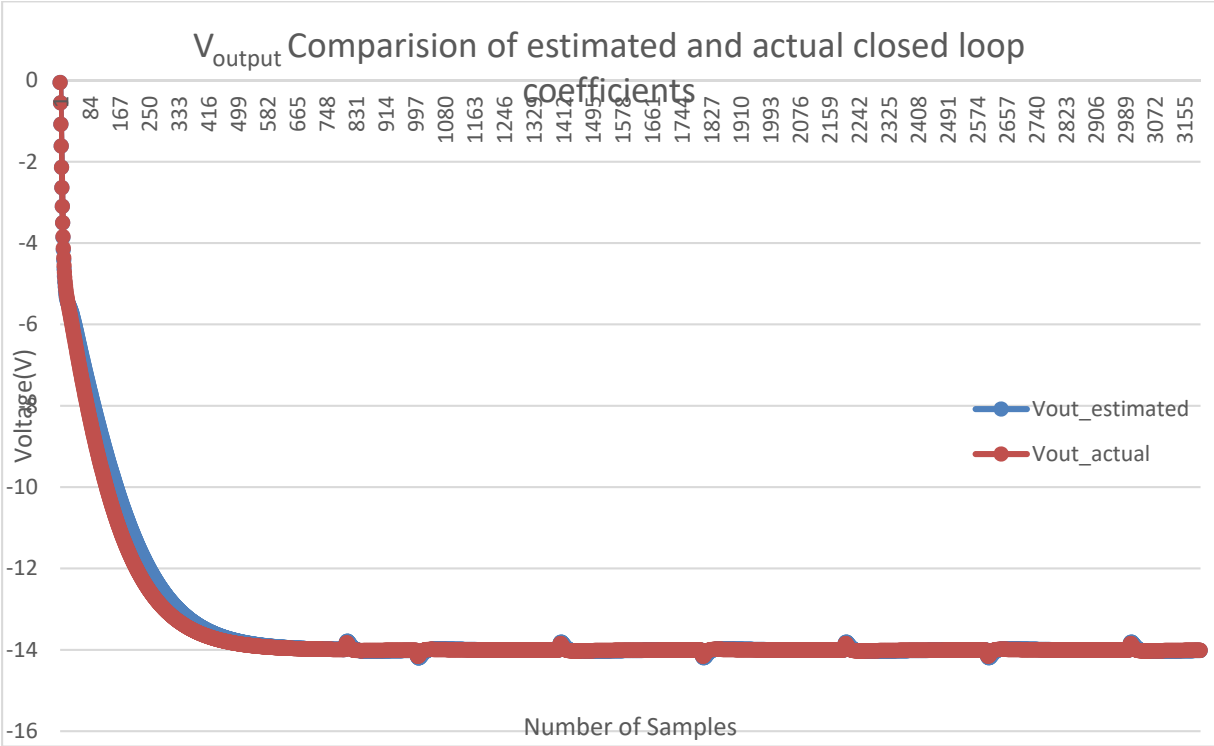


Figure 5.104. V_{output} comparison of estimated and actual closed loop coefficients

Comparison of estimated and actual closed loop coefficients

Parameters	Actual	Estimated by *tfest*
b_0	0.4771	0.4062
b_1	-0.9238	-0.7944
b_2	0.4471	0.3866
a_1	0	0

5.10 Summary of the parameter identification in DC/DC converters

Summary table for parameter estimation of DC/DC converters

Parameter Identification of DC-DC Converter					
BUCK CONVERTER					
	L(H)	**$R_L(\Omega)$**	**C(F)**	**$R_C(\Omega)$**	**$R_d(\Omega)$**
Manufacturers' Data	**Yes**	No	**Yes**	No	No
Offline Method					
T_{on}	**Yes**	**Yes**	**Yes**	**Yes**	No
T_{off}	No	No	No	No	**Yes**
Online Method					
T_{on}	**Yes**	**Yes**	**Yes**	**Yes**	No
T_{off}	No	No	No	No	**Yes**
BOOST CONVERTER					
Manufacturers' Data	**Yes**	No	**Yes**	No	No
Offline Method					
T_{on}	No	No	No	No	No
T_{off}	**Yes**	**Yes***	**Yes**	**Yes**	**Yes***
Online Method					
T_{on}	**Yes**	**Yes**	No	No	No
T_{off}	No	No	**Yes**	**Yes**	**Yes**
BUCK-BOOST CONVERTER					
Manufacturers' Data	**Yes**	No	**Yes**	No	No
Offline Method					
T_{on}	No	No	No	No	No
T_{off}	**Yes**	**Yes***	**Yes**	**Yes**	**Yes***
Online Method					
T_{on}	**Yes**	**Yes**	No	No	No
T_{off}	No	No	**Yes**	**Yes**	**Yes**

* The value calculated is combination of (R_L+R_d). The two parameters cannot be estimated separately as they are in series during the corresponding scenario

5.11 Parameter identification in passive filters. EMI filter design

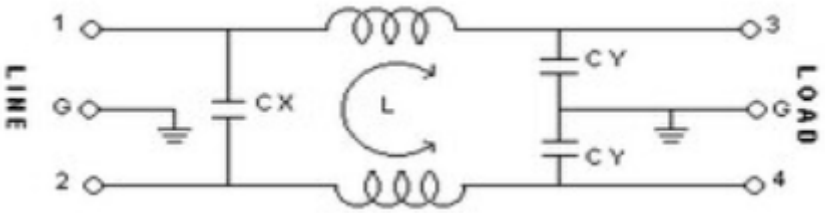


Figure 5.105. A typical EMI Filter

The EMI (Electromagnetic Interference [45] is an external disturbance that affects the electrical and electronics circuits by induction or coupling and can cause severe damage to the systems.

The EMI filters (Electromagnetic Interference filters) are mainly used to nullify the effect of electromagnetic waves before feeding to the converter or a load.

The Noise in the EMI filter of the conduction mode can be classified in two types, firstly the differential Mode noise and secondly common mode noise.

The differential mode noise is an electrical signal, which appears in one or two of the lines in a closed loop. The noise appears on two conductors of a closed loop, it appears in series with the desired signal while the current flows in opposite directions. In a typical circuit, the current will flow in opposite directions, to and from the load.

The differential mode noise is the noise produced in the supply line and this type of noise is suppressed by installing a Capacitor on the power supply line, as shown in Figure 1. The capacitor C_x is used to reduce the differential mode noise.

Common mode (CM) noise is the noise signal between the ground and neutral/phase.

The common mode is when the two coils is wound on a single core and mainly for reducing the EMI from power supplies. It mainly blocks the common mode currents[46], the magnetic flux produced by differential-mode (DM) currents in the core tend to cancel each other out since the windings are negative coupled, The CM currents, however, see a high impedance due to the combined inductance of the positive coupled windings.

Common mode choke coils are used to suppress common mode noise. This type of coil is produced by winding the signal or supply wires to one ferrite core. Since magnetic flux flows inside the ferrite core, common mode choke coils work as an inductor against common mode current. Accordingly, using a common mode choke coil provides larger impedance against common mode current and is more effective for common mode noise suppression than using several normal inductors

Common mode choke coils work as a simple wire against differential mode current (signal), while they work as an inductor against common mode current (noise).

---

[45] C. S. Moo, H. C. Yen, Y. C. Hsieh, and Y. C. Chuang, "Integrated Design of EMI filter and PFC Low-Pass Filter for Power Electronic Converters," IEE-Electric Power Applications, Vol. 150, No. 1, pp. 39-44, Jan. 2003.

[46] https://www.murata.com/~/media/webrenewal/products/emc/emifil/knowhow/26to30.ashx

The common mode noise can be decreased by using a common mode coil (Inductor with Coupling) and a capacitor installed in each line. The Capacitor C_x and the inductor L_x with coupling are the components connected in the line to suppress the common mode noise. The Capacitor C_y reduces the differential mode noise.

5.11.1 EMI Filter Design. Parameter identification based on Manufacturers data.

The topology of the EMI filter is available in the component and the manufacturer will give the schematic diagram of the filter with the values of the capacitances and inductances of the common mode and discrete mode[47].

Data from manufacturers' datasheet of an EMI filter

Parameters	Values
V_{input}	250 V
I_{output}	2 A
C_x(Common mode capacitance)	0.047uF ± 20%
C_y(Differential mode capacitance)	2200 pF ± 20%
L_x(Common mode Inductance)	2.4 mH

5.11.2 EMI Filter Design. Parameter identification based on offline data

The open load or short circuit cannot be performed for the filter to estimate the parameters of the components. One approach, which can be followed, is connecting the filter to different loads and perform the parametric estimation as done in the online method.

5.11.3 EMI Filter Design. White-box. Parameter identification based on online data

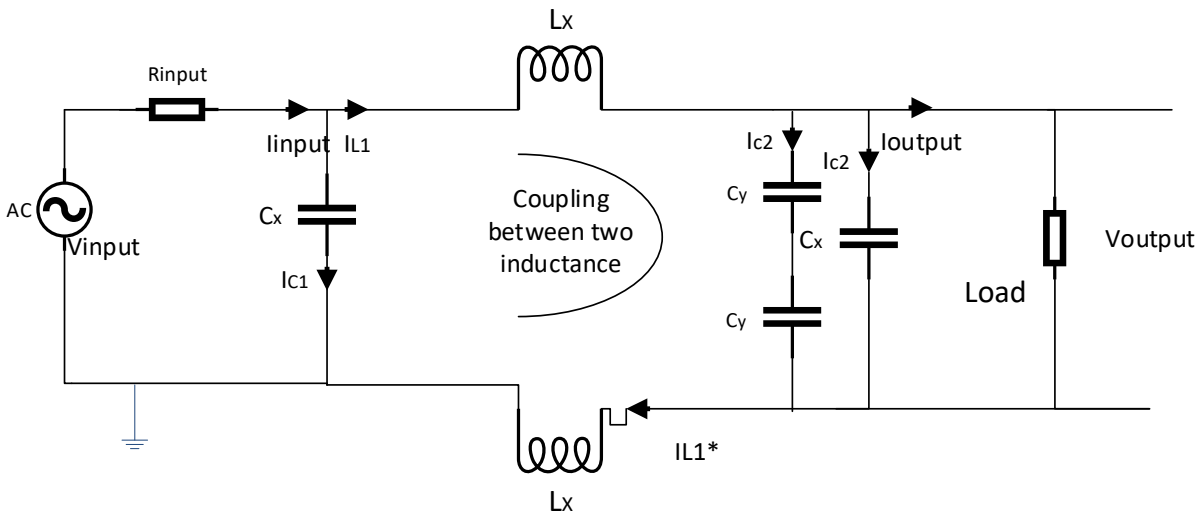


Figure 5.106. Schematic Diagram of an EMI filter

[47] http://www.mouser.com/ds/2/382/EMI_Power_Catalog-34547.pdf page 24 -28

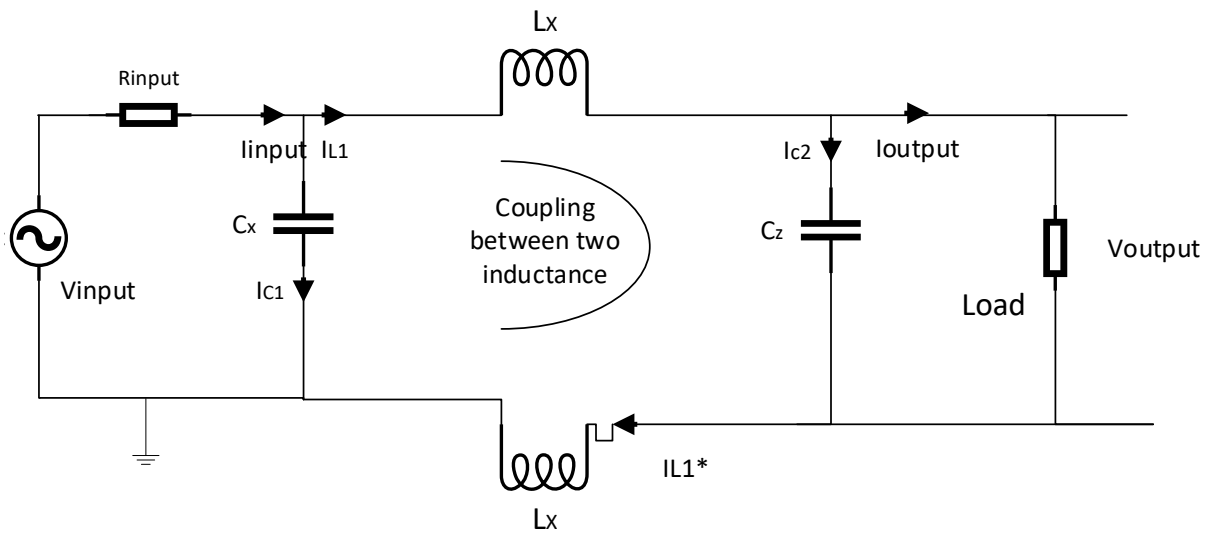


Figure 5.107. Modified schematic diagram of an EMI filter

The R_{inut} and the R_{load} in the circuit is 50 ohms as specified[48].

The equations for the Figure 2 is written below and by these equations the parameters of the EMI filter can be estimated

$$\int (I_{input} - I_{output}) \cdot dt = C_x \cdot \int d(V_{input} - I_{input} \cdot R_{input}) + C_z \cdot \int d(V_{output})$$

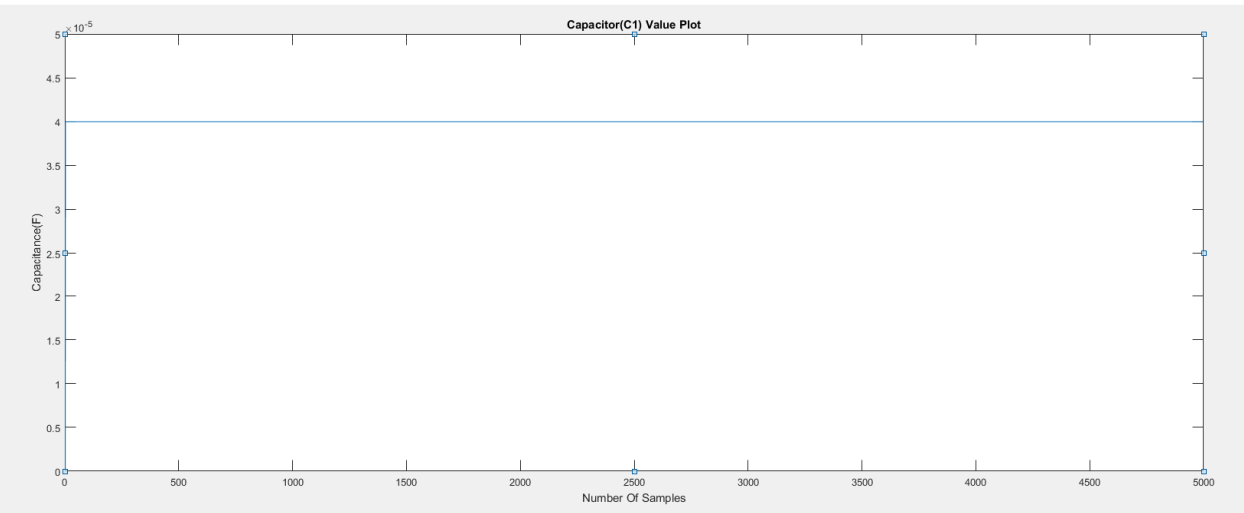


Figure 5.108. Common mode capacitance value plot

[48] Gundars Asmanis, Measurement and modeling of EMI filters high frequency parasitic parameters, PhD Thesis, Riga Technical University, 2014

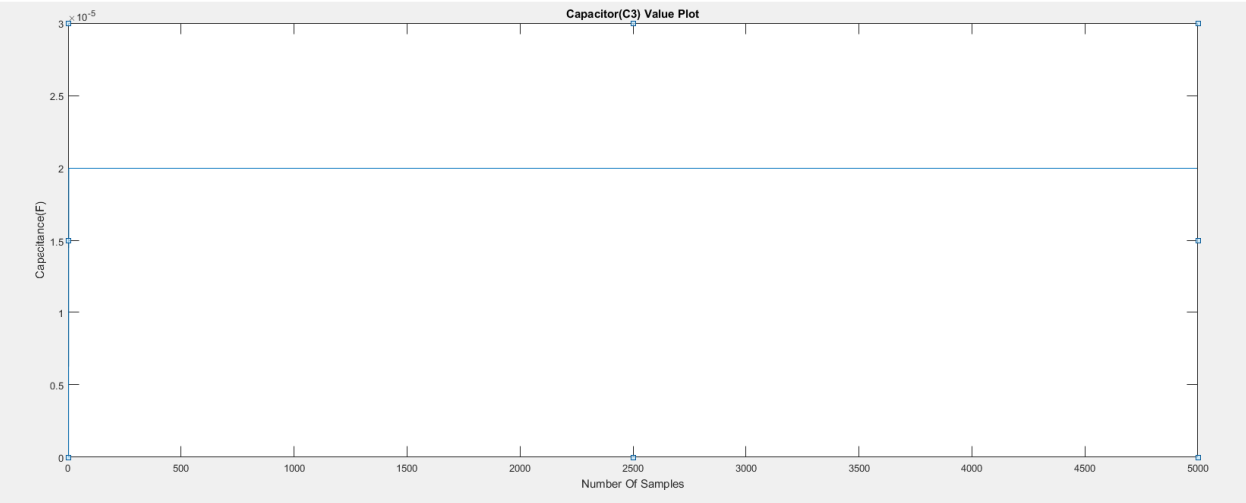


Figure 5.109. Differential mode capacitance value plot

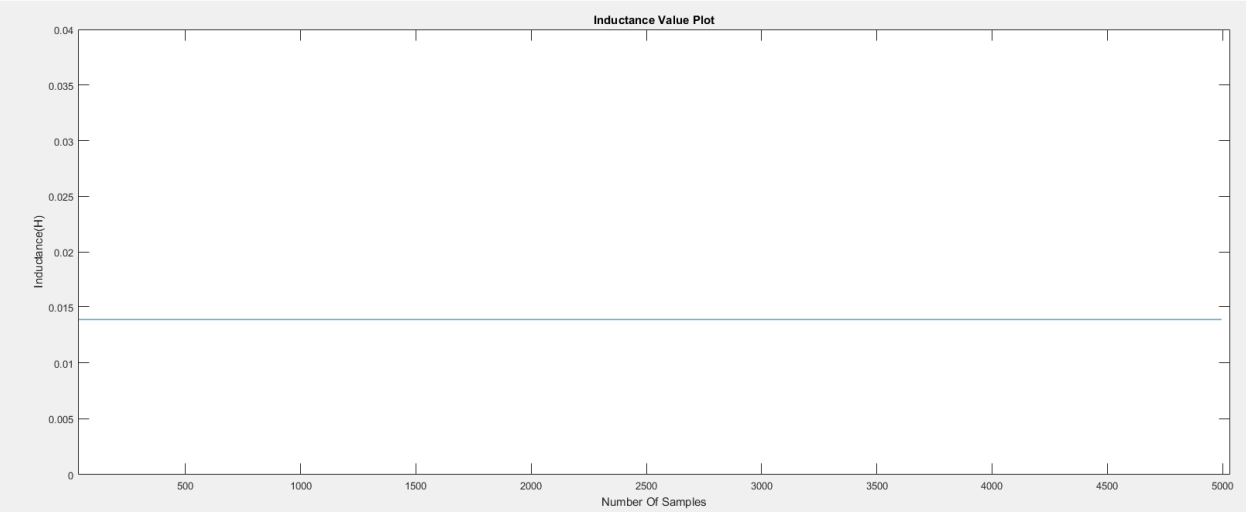


Figure 5.110. Common mode inductance value plot

The parameters from online method

Parameter	Actual Value	Calculated Value
C1	40 uF	40 uF
C2	20 uF	20 uF
L1	15 mH	14.5 mH

5.12 Parameter identification in passive filters. Inverter filter design

A typical filter design for inverters (single-phase model) is as follows,

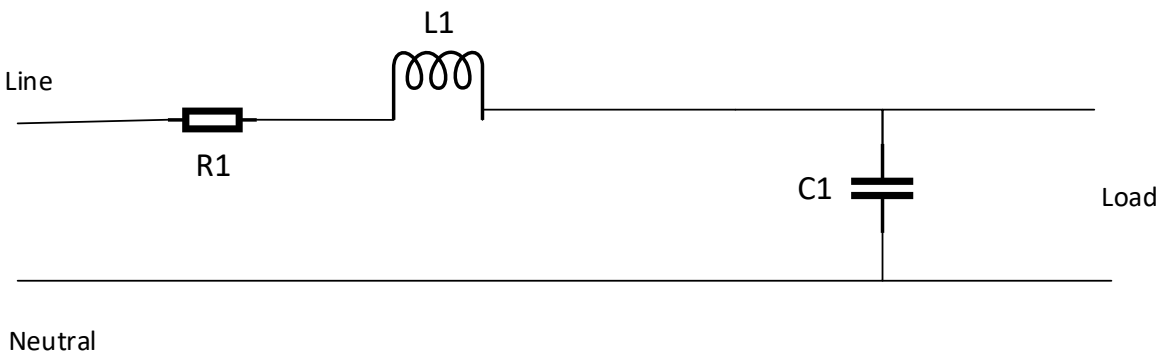


Figure 5.111. Single-phase filter for inverter

The filter for the inverter consists of the R, L, and C. It is connected to the load or other electronic components so that it suppresses the noise. The modelling and parametric estimation of this filter is explained below.

5.12.1 Inverter filter design. Parameter identification based on manufacturers' data

The manufactures data for a single phase filter with R, L, C values cannot be found, since the manufacturer keeps the data confidential.

5.12.2 Inverter filter design. Parameter identification based on offline data

For the offline method an open load can be conducted and the equations for calculating the different parameters are explained below.

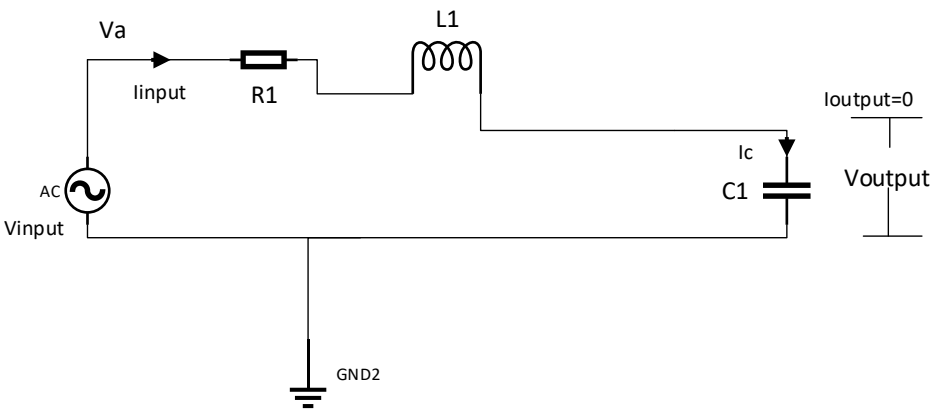


Figure 5.112. Open load test on the inverter filter

The equations to determine *C*

$$I_{input} = C\frac{dV_{output}}{dt} \rightarrow C\frac{dV_{output}}{dt} = I_{input}$$

$$C\int dV_{output} = \int (I_{input})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input})dt$$

Offline method. The inverter parameters

Parameters	Simulated	Calculated
L	15e-3 H	15e-3 H
R	0.06 ohms	0.06 ohms
C	40 uF	40 uF

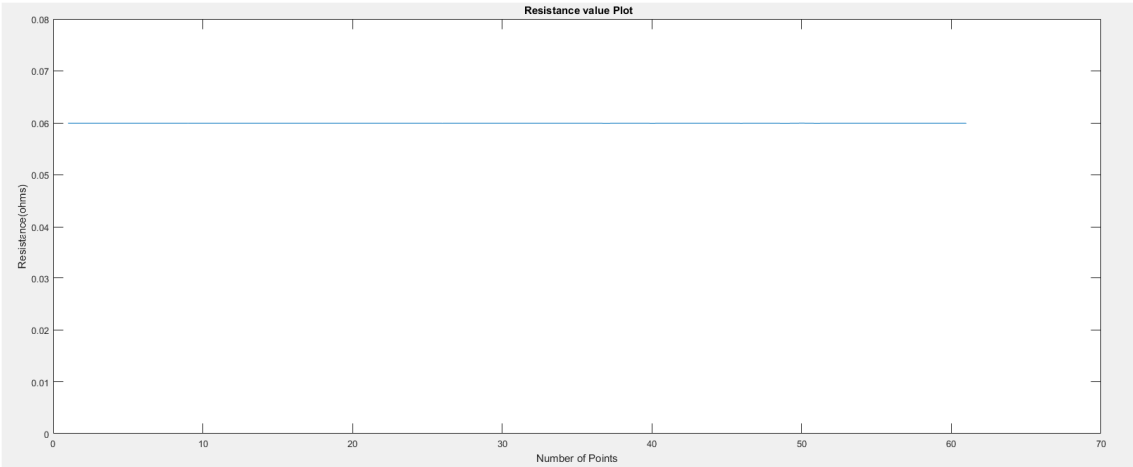


Figure 5.113. The resistance value plot

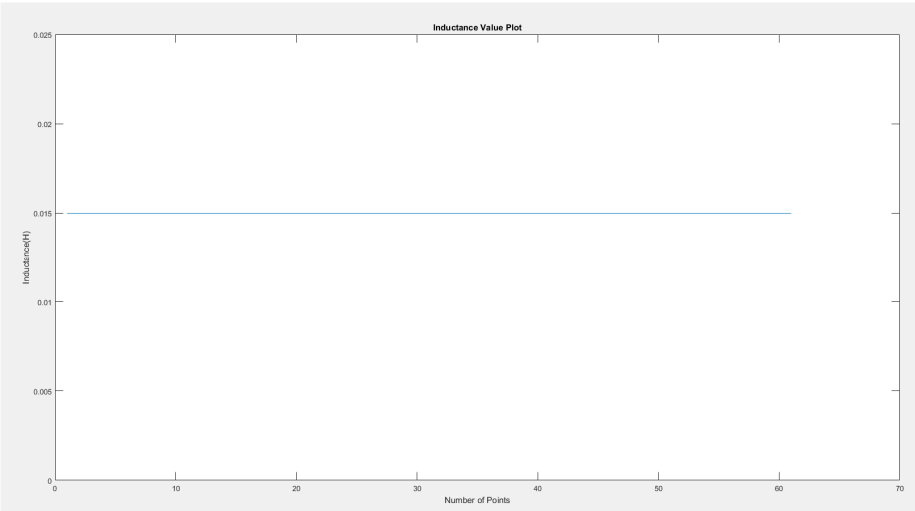


Figure 5.114. The Inductance Value Plot

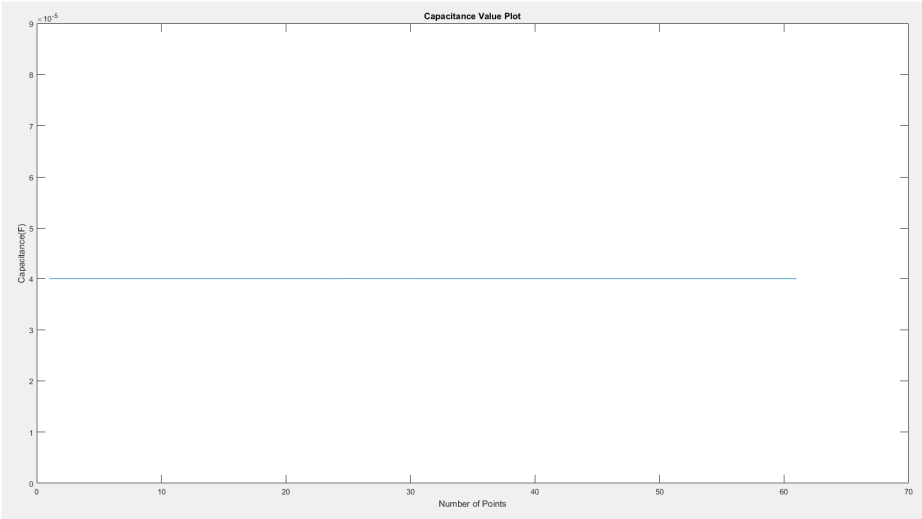


Figure 5.115. The capacitance value plot

5.12.3 Inverter Filter Design. White-box. Parameter identification based on online data

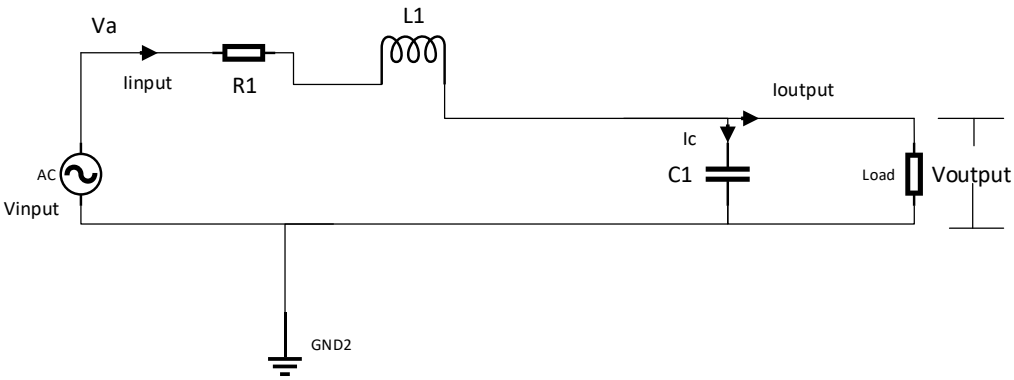


Figure 5.116. Single-phase filter for inverters

The equations to determine *C*

$$I_{input} = C\frac{dV_{output}}{dt} + I_{output} \rightarrow C\frac{dV_{output}}{dt} = I_{input} - I_{output}$$

$$C\int dV_{output} = \int (I_{input} - I_{output})dt \rightarrow \int dV_{output} = \frac{1}{C}\int (I_{input} - I_{output})dt$$

Online method. The inverter parameters

Parameters	Simulated	Calculated
L	15e-3 H	15e-3 H
R	0.06 ohms	0.06 ohms
C	40 uF	40 uF

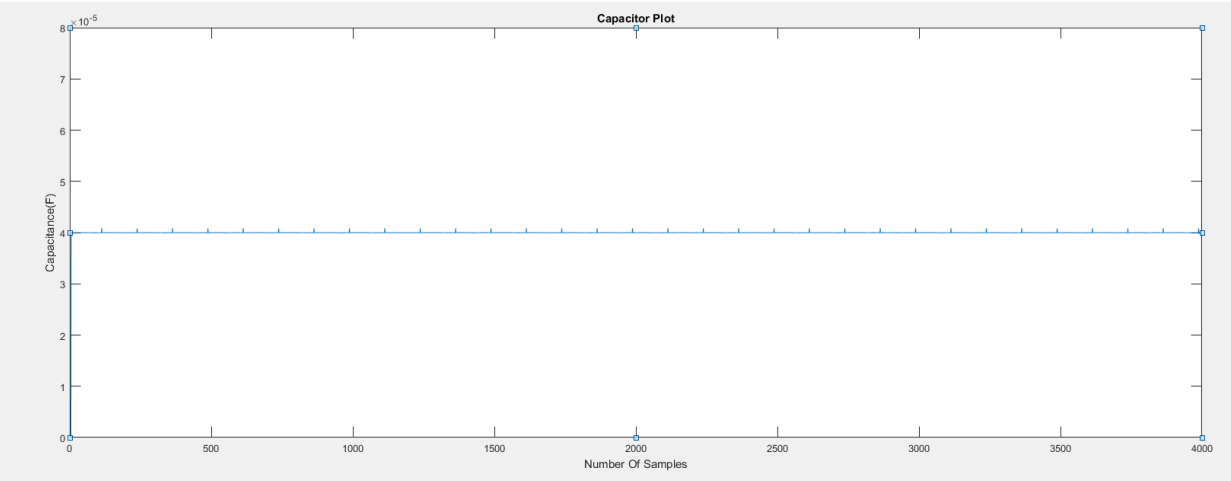


Figure 5.117. The capacitance value plot

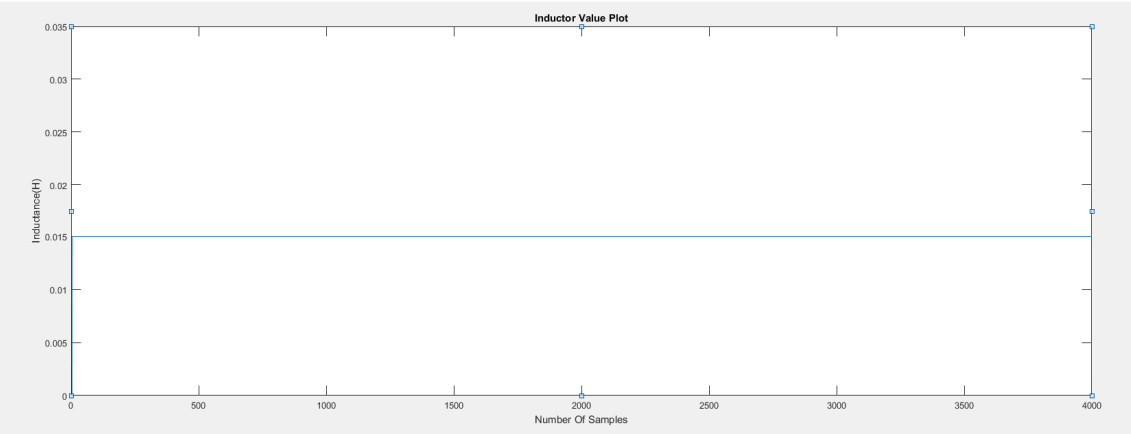


Figure 5.118. The inductance value plot

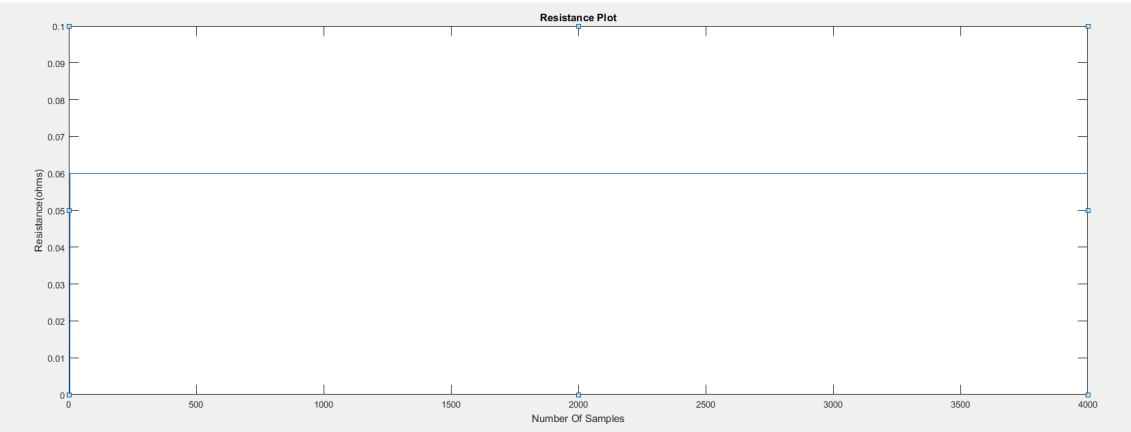


Figure 5.119. The resistance value plot

5.13 Six-pulse diode rectifier modelling. Parametric identification based on manufactures data

The manufactures discloses the forward voltage drop and the diode resistance directly. Hence, the power diode parameters can be used for the six pulse rectifier modelling[49].

Parameters from manufacturers' datasheet

Parameters	Value
Vinput	100 -1200 V
Ioutput	12A
Rd	10.7m ohms
Vd_{max}	1.26 V

5.13.1 Six-pulse diode rectifier modelling. Parametric identification based on offline data

For the offline method it is difficult to conduct the no load or short circuit test, so it is possible to deal with different loads as in the online method approach.

5.13.2 Six-pulse diode rectifier modelling .White box parametric identification based on online data

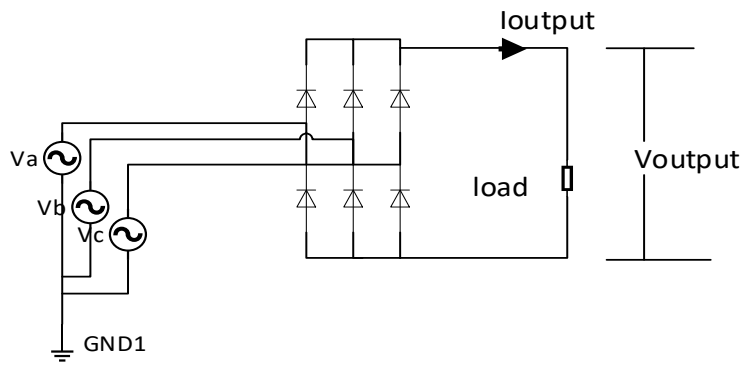


Figure 5.120. Six-pulse rectifier

Figure 1 is the basic six pulse diode based rectifier with a load[50]. Considering the on cycle in which the diode (D1) and diode (D6) will be conducting. The main parameter identification for the rectifier is the diode resistance (Rd) and the diode voltage (Vd). For calculating these two parameters, different approaches are proposed. The Six Pulse rectifier

Approach 1:

[49] https://www.vishay.com/docs/93487/vs-12fseries.pdf

[50] L. Han, J. Wang, D. Howe, "State-space average modelling of 6-and 12-pulse diode rectifiers," In proc. of Power Electronics and Applications Conference, EPE 2007, pp.1-10

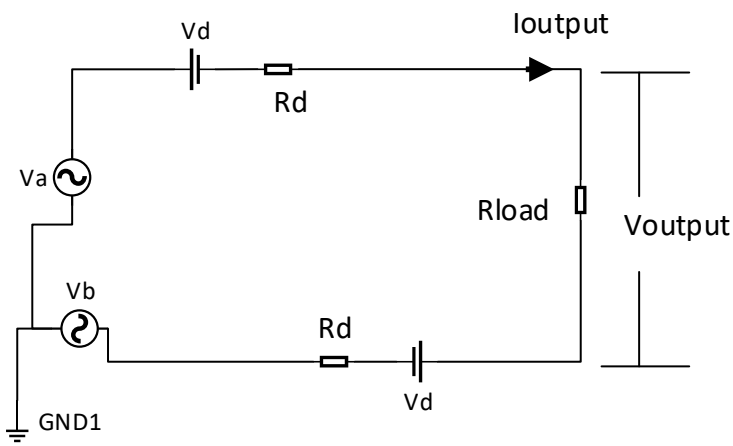


Figure 5.121. Six pulse rectifier when diodes D1 and D6 are conducting (during 0 to 60^0)

$$V_a - V_b = I_{output} \cdot R_d + V_d + V_{output} + I_{output} \cdot R_d + V_d$$

$$V_a - V_b - V_{output} - 2 \cdot V_d = 2 \cdot I_{output} \cdot R_d$$

$$R_d = \frac{V_a - V_b - V_{output} - 2 \cdot V_d}{2 \cdot I_{output}}$$

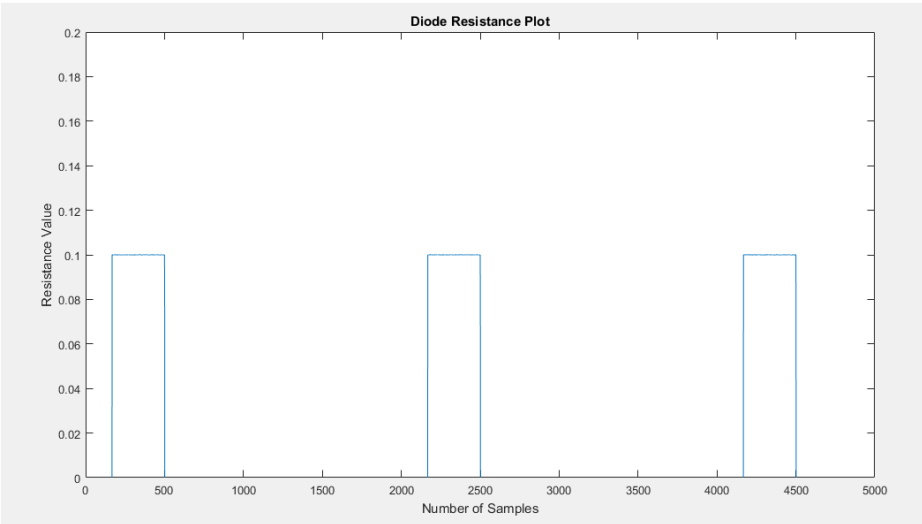


Figure 5.122. Diode resistance plot

The parameter from online method

Parameters	Actual value	Calculated Value
Rd	0.1 ohms	0.1 ohms

Approach 2:

Assuming only the linear characteristics of diode:

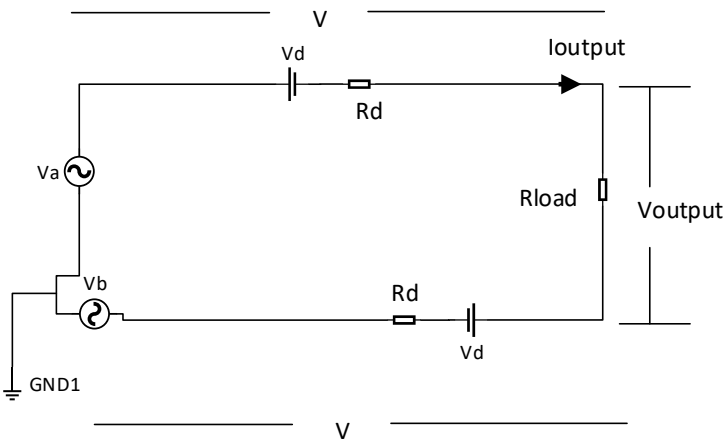


Figure 5.123. Six pulse rectifier when diodes D1 and D6 are conducting

$$V_a - V - V_{output} - V - V_b = 0$$

$$\frac{V_a - V_{output} - V_b}{2} = V$$

Where V is voltage across the diode.

$$V = V_d + I \cdot R_d$$

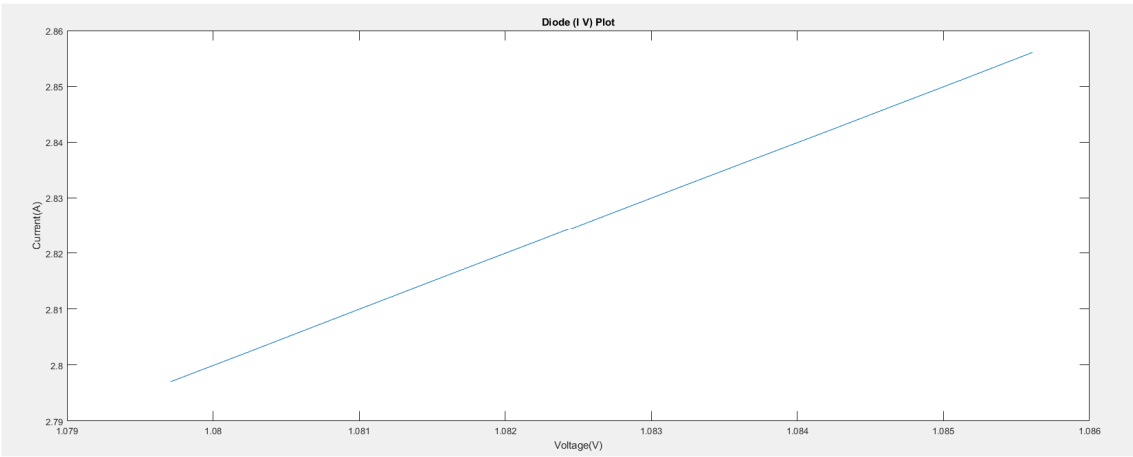


Figure 5.124. Plot of the voltage across Diode

The parameters from the online method

Parameters	Simulated	Calculated
V_d	0.8 V	0.8 V
R_d	0.1 ohms	0.1 ohms

State-of-the-art of parameter fitting methods

6 State-of-the-art of parameter fitting methods

The area of **system identification applies statistical methods to generate mathematical models of dynamical systems from measured data**.

- **Parameter fitting methods are applied to fit model parameters to experimental data.**
- **They apply optimization methods to find the best values of the parameters.**
- **Parameter fitting methods are mainly applied to data acquired from online tests.**
- **Parameter fitting methods are specifically intended for black-box and grey-box models.**

Offline parameter identification methods cannot fully reproduce real operating conditions such as temperature changes or specific environmental conditions. Therefore **when high precision is required** offline identification methods sometimes cannot meet the requirements and online **parameter identification methods are preferred**[51]at the expense of increasing complexity.

Most of the online parameter identification methods require the **application of parameter fitting algorithms** to adjust the output of the model with the experimental response of the electric/electronic system under analysis to accurately obtain the values of the parameters of the simulation model that allow a better prediction of the behavior of such system. By this way the parameters obtained allow reproducing with more fidelity the response of the electric/electronic system in a more realistic way, taking into account effects that otherwise cannot be considered from manufacturers' data sheets or by using an offline parameter identification approach.

The values of the parameters obtained are usually influenced by the objective function dealt with. **Different objective functions can be defined**, for example, to minimize the sum of squares (SoS)of the difference between the model at parameter *q* and the experimental [52]or the root mean square error (RMSE)[53]data acquired over several experiments,

$$\min(SoS(p_1, p_2, \ldots p_m)) = \min\left(\frac{1}{n}\sum_{i=1}^{n}(y_{\text{model},i} - y_{\text{exp},i})^2\right) \quad \textbf{SoS}$$

$$\min(RMSE(p_1, p_2, \ldots p_m)) = \min\left(\sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{\text{model},i} - y_{\text{exp},i})^2}\right) \quad \textbf{RMSE}$$

p_i(i = 1,…,m) being the parameters to be identified, $y_{model,i}$ the predicted output variable at time $i \cdot \Delta t$ and $y_{exp,i}$ the experimental value.

[51]X. Zhan, G. Zeng, J. Liu, Q. Wang, S. Ou, A Review on Parameters Identification Methods for Asynchronous Motor, International Journal of Advanced Computer Science and Applications, vol. 6(1), 2015
[52] Drayton Munster, Parameter Identification: A Comparison of Methods, July 19, 2009
[53] F. Ferracuti *et al.*, Data-driven models for short-term thermal behaviour prediction in real buildings, Applied Energy, 2017.

Therefore, **the SoS or the RMSE are the objective functions to be minimized** both for training and validation sets.

A **sensitivity analysis**is often appliedto better understand how muchthe solution changes when applying small changes in the parameters. The sensitivity analysis provides an indication of the parameters to be computed very accurately[54].

Model validation

Some authors suggest to **normalize the values of the parameters in the [0,1] range[55]** to improve the efficiency of the search session. This transformation is easily done by applying a diagonal transformation matrix,
Where the transformation matrix (OM = order of magnitude) is as follows,

$$\begin{pmatrix} p_1 \\ p_2 \\ \dots \\ p_m \end{pmatrix} = \begin{pmatrix} 10^{-x1} & & & \\ & 10^{-x2} & & \\ & & \dots & \\ & & & 10^{-xm} \end{pmatrix} \cdot \begin{pmatrix} P_1 \\ P_2 \\ \dots \\ P_m \end{pmatrix}$$

OM = orders of magnitude $(p_{normalized}) = (OM_{diagonal})\cdot(P)$

The deviations between predicted and measured data during both training and validation sessions can be quantified using the following indices,

Mean absolute error: $MAE = \frac{1}{n}\sum_{i=1}^{n}\left|y_{\text{model},i} - y_{\text{exp},i}\right|$

Mean absolute percentage error: $MAPE = \frac{1}{n}\sum_{i=1}^{n}\left|\frac{y_{\text{model},i} - y_{\text{exp},i}}{y_{\text{exp},i}}\right|$

Root mean square error: $RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_{\text{model},i} - y_{\text{exp},i})^2}$

6.1 Linear parameter fitting methods

Linear systems satisfy the superposition principle[56]. Linear problems involve a linear error between data and model output in the parameters and the error sum of squares is applied as a loss function as[57],

[54] Drayton Munster, Parameter Identification: A Comparison of Methods, July 19, 2009

[55]M. Hu, F. Xiao, L. Wang, Investigation of demand response potentials of residential air conditioners in smart grids using grey-box room thermal model, Applied Energy, 2017.

[56] Stephen A Billings, Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains, John Wiley & Sons, Ltd., 2013.

[57]Oliver Nelles, Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models, Springer, 2001.

$$y_i = y(i\Delta t) = \sum_{j=1}^{m} p_j \cdot g_j(\overline{u}(i\Delta t)) = p_1 \cdot g_1(\overline{u}(i\Delta t)) + p_2 \cdot g_2(\overline{u}(i\Delta t)) + \ldots + p_m \cdot g_m(\overline{u}(i\Delta t))$$

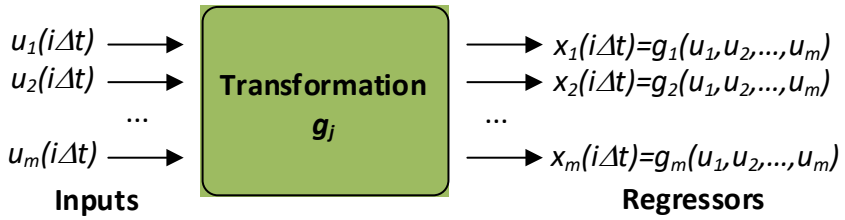


Fig. 6.1. Transformation from input parameters to repressor's

Where $g_j(u)$ are the regressors, $u(t)$ is the inputvector variables with time dependency where $t = (1,2,\ldots n)\cdot\Delta t$, $y(t)$ is the output variable, and p_j are the linear parameters, which can be estimated by linear optimization.

The regressors $g_j(u)$ can depend in any nonlinear way on the measured inputs $u(t)$. Therefore, with some a-priori knowledge, many nonlinear optimization problems can be transformed into linear ones.

Nonlinear optimization problems can be transformed into linear ones. For example, with voltages and currents as measured inputs u_1 = V, u_2 = I and the knowledge that the output depends only on the electrical power (Power = $y = V\cdot I$), a linear optimization problem arises by taking $V\cdot I$ and current as regressor, $x = g(u) = u_1\cdot u_2$. Intelligent preprocessing can often reduce the complexity by simplifying nonlinear to linear optimization problems.

6.1.1 Parameter fitting based on least squares optimization (MISO, multiple-input single-output)

It is the most widely applied solution for linear optimization problems[58].The goal is to find the model output y_{model} that best approximates the measured output y_{exp} in the least squares sense, i.e., with the minimal sum of squared error loss function value, that is,

$$\min\left(\frac{1}{n}\sum_{i=1}^{n}(y_{\text{model},i} - y_{\text{exp},i})^2\right) = \min\left(\frac{1}{n}\sum_{i=1}^{n}(\sum_{j=1}^{m} p_j \cdot g_j(\overline{u}) - y_{\text{exp},i})^2\right) = \min\left(\frac{1}{n}\sum_{i=1}^{n}(\sum_{j=1}^{m} x_j(i\cdot\Delta t) - y_{\text{exp},i})^2\right)$$

Where the sub index i refers to time and j to the sample

We assume the linear relationship $y_i = y(i\Delta t) = \sum_{j=1}^{m} p_j \cdot x_j$

[58]Oliver Nelles, Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models, Springer, 2001.

Where $W = \begin{pmatrix} x_1(\Delta t) & x_2(\Delta t) & .. & x_m(\Delta t) \\ x_1(2\Delta t) & x_2(2\Delta t) & ... & x_m(2\Delta t) \\ ... & ... & ... & ... \\ x_1(n\Delta t) & x(n\Delta t) & ... & x_m(n\Delta t) \end{pmatrix}$, $Y_{\exp} = \begin{pmatrix} y_{\exp}(\Delta t) \\ y_{\exp}(2\Delta t) \\ .... \\ y_{\exp}(n\Delta t) \end{pmatrix}$ and $P = \begin{pmatrix} p_1 \\ p_2 \\ ... \\ p_m \end{pmatrix}$

The linear regression in matrix form is as,

$$\boldsymbol{Y_{model(n,1)} = W_{(n,m)} \cdot P_{(m,1)}}$$

When $m = n$ there exist a unique solution and $W = (x_{ij})$ i = 1,…,n and j = 1,…,m. The index i refers to time whereas the index j refers to the number of inputs.

Solving for P leads to $\boldsymbol{P_{optim} = (W^T \cdot W)^{-1} \cdot W^T \cdot Y_{exp}}$ where $x_{ij} = g_j(u_i)$

We assume that square matrix $W^T \cdot W$ is invertible.

6.1.2 Parameter fitting based on ARX (AutoRegressive with eXogenous)

The **SISO (single-input single-output) ARX model**[59] shown in Figure 2, is the simplest model and the most efficient of the polynomial estimation methods because it is the result of solving linear regression equations in analytic form. The ARX model therefore is preferable, especially when the model order is high.

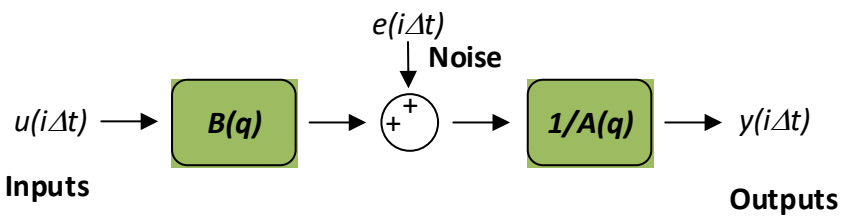


Fig. 6.2. SISO-ARX diagram

The **SISO ARX** model structure is:

$$y(i\Delta t) + a_1 y(i\Delta t - 1\Delta t) + a_2 y(i\Delta t - 2\Delta t) + ... + a_{na} y(i\Delta t - n_a \Delta t) =$$
$$= b_1 u(i\Delta t - n_k \Delta t) + ... + b_{nb} u(i\Delta t - n_k \Delta t - n_b \Delta t + 1\Delta t) + e(i\Delta t)$$

From the previous expression it results,

$$y(i\Delta t) = -a_1 y(i\Delta t - 1\Delta t) - a_2 y(i\Delta t - 2\Delta t) + ... - a_{na} y(i\Delta t - n_a \Delta t) +$$
$$+ b_1 u(i\Delta t - n_k \Delta t) + b_2 u(i\Delta t - n_k \Delta t - 1\Delta t) + ... + b_{nb} u(i\Delta t - n_k \Delta t - n_b \Delta t + 1\Delta t) + e(i\Delta t)$$

$y(i\Delta t)$ being the output at time $i \cdot \Delta t$, n_a the number of poles, n_b the number of zeroes plus 1, n_k the number of input samples that occur before the input affects the output, also called the *dead time* in the system.

$y(i\Delta t - 1\Delta t)$ … $y\, i\Delta t - n_a \Delta t)$ are previous outputs on which the current output depends, $u(i\Delta t - n_k \Delta t)$…$u(i\Delta t - n_k . \Delta t - n_b . \Delta t + 1\Delta t)$ are previous and delayed inputs on which the current output depends and $e(i\Delta t)$ is white-noise disturbance value.

[59] [ARX-Matlab] ARX Model(online)http://fr.mathworks.com/help/ident/ref/arx.html

Parameters n_a *and* n_b are the orders of the ARX model (inputs and output) , whereas n_k is the delay. Which can be compacted as,

$$y(i\Delta t) = -\sum_{j=1}^{n_a} a_j \cdot y(i\Delta t - j\Delta t) + \sum_{j=1}^{n_b} b_j \cdot u(i\Delta t - n_k\Delta t - j\Delta t + 1\Delta t) + e(i\Delta t)$$

Or in a more compact form,

$$\boxed{y(i) = -\sum_{j=1}^{n_a} a_j \cdot y(i-j) + \sum_{j=1}^{n_b} b_j \cdot u(i - n_k - j + 1) + e(i)}$$

Taking into account the **SISO ARX model** we can define,

$u(i) = u(i)$ (single-input value defined in $t = i\Delta t$) *dim =1*

$P = [a_1, a_2, \ldots, a_{na}, b_1, b_2, \ldots, b_{nb}]^T$ the vector of coefficients with dim =(n_a+n_b,1)

$W_i = [-y(i-1), -y(i-2), \ldots -y(i-n_a), u(i-n_k), u(i-n_k-1), \ldots, u(i-n_k-n_b+1)]^T$ the vector of outputs and inputs with dim= (n_a+n_b,1)

The SISO ARX model can be written in the form of a linear regression model as,

$$y_{model,I\,(1,1)} = P^T{}_{(1,na+nb)} \cdot W_{i(na+nb,1)}$$

The solution is given by $P_{optim} = (W^T \cdot W)^{-1} \cdot W^T \cdot Y_{exp}$

A more compact way to write the difference equation is

$$A(q)y(i\Delta t) = B(q)u(i\Delta t - n_k) + e(i\Delta t)$$

q being the delay operator. Specifically, $q^{-na}u(i\Delta t) = u(i\Delta t - n_a\Delta t)$.

$$A(q) = 1 + a_1 q^{-1} + \ldots + a_{na} q^{-na}$$

$$B(q) = b_1 + b_2 q^{-1} + \ldots b_{nb} q^{-nb+1}$$

For **time-series data that contains no inputs**, one output and orders = n_a, the model has **AR structure** of order n_a. In this case, the **AR model structure** is,

$$A(q)Y(i\Delta t) = e(i\Delta t)$$

or alternatively, $\boxed{y_{\text{model}}(i) = -\sum_{k=1}^{n_a} a_k \cdot y(i-k) + e(i)}$

A **multi-input multi-output (MIMO)** system can be decomposed into several multi-input single-output (MISO) systems [60].

The general form of a **MISO ARX model** is as,

$$y(i\Delta t) = -\sum_{j=1}^{n_a} a_j \cdot y(i\Delta t - j\Delta t) + \sum_{j=1}^{n_b} b_j \cdot u(i\Delta t - n_k\Delta t - j\Delta t + 1\Delta t) + e(i\Delta t)$$

$\boxed{y_{\text{model}}(i) = -\sum_{j=1}^{n_a} a_j \cdot y(i-j) + \sum_{j=1}^{n_b}\sum_{l=1}^{r} b_{jl} u_l(i - n_k - j + 1) + e(i)}$ where $e(i) = y_{model}(i) - y_{exp}(i)$

[60] J. Ding, C. Dong and Y. Yang, "Bias compensation based hierarchical parameter estimation for dual-rate sampled systems with colored noises," 2015 34th Chinese Control Conference (CCC), Hangzhou, 2015, pp. 1953-1957. doi: 10.1109/ChiCC.2015.7259930

Where,

$u = [u_1,u_2,\ldots,u_r]^T$ is the inputs vector with r inputs

$W_i = [-y(i-1),-y(i-2),\ldots-y(i-n_a),u_1(i-n_k),u_1(i-n_k-1),\ldots,u_1(i-n_k-n_b+1),\ldots, u_r(i-n_k),u_r(i-n_k-1),\ldots,u_r(i-n_k-n_b+1)]^T$ with dim $=(n_a+r\cdot n_b,1)$

$P = [a_1,a_2,\ldots,a_{na},b_{11},b_{21},\ldots,b_{1,nb},\ldots, b_{r1},b_{r1},\ldots,b_{r,nb}]^T$ is the vector of parameters with dim= $(n_a+r\cdot n_b,1)$

The MISO ARX model can be expressed as a linear regression model as,

$$y_{model,i\,(1,1)} = P^T_{(1,na+r\cdot nb)}\, W_{i(na+r\cdot nb,1)}$$

The solution is given by $P_{optim} = (W^T\cdot W)^{-1}\cdot W^T\cdot Y_{exp}$

6.1.3 Parameter fitting based on ARMAX (AutoRegressive Moving Average with eXogenous)

Unlike ARX, in the autoregressive-moving average with exogenous terms (ARMAX) model, the system structure includes the stochastic dynamics. ARMAX models are useful when having dominating disturbances that enter early in the process, such as at the input. For example, a wind gust affecting an aircraft is a dominating disturbance early in the process. The ARMAX model has more flexibility than the ARX model in handling models that contain disturbances.

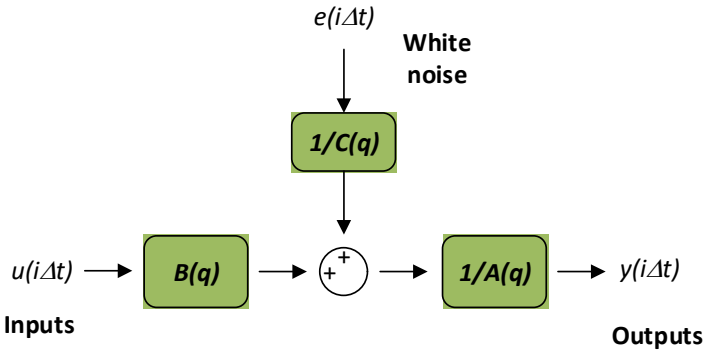


Fig. 6.3. SISO-ARMAX diagram

The **SISO ARMAX** model structure is[61]:

$$\begin{aligned} y(i\Delta t) + a_1 y(i\Delta t - 1\Delta t) + a_2 y(i\Delta t - 2\Delta t) + \ldots\ldots + a_{na} y(i\Delta t - n_a\Delta t) = \\ = b_1 u(i\Delta t - n_k\Delta t) + \ldots + b_{nb} u(i\Delta t - n_k\Delta t - n_b\Delta t + 1\Delta t) \\ + e(i\Delta t) + c_1 e(i\Delta t - 1\Delta t) + \ldots + c_{nc} e(i\Delta t - n_c\Delta t) \end{aligned}$$

From the previous expression it results,

$$\begin{aligned} y(i\Delta t) = -a_1 y(i\Delta t - 1\Delta t) - a_2 y(i\Delta t - 2\Delta t) - \ldots\ldots - a_{na} y(i\Delta t - n_a\Delta t) = \\ + b_1 u(i\Delta t - n_k\Delta t) + \ldots + b_{nb} u(i\Delta t - n_k\Delta t - n_b\Delta t + 1\Delta t) \\ + e(i\Delta t) + c_1 e(i\Delta t - 1\Delta t) + \ldots + c_{nc} e(i\Delta t - n_c\Delta t) \end{aligned}$$

[61] Eric H.K. Fung, Y.K. Wong, H.F. Ho, Marc P. Mignolet, Modelling and prediction of machining errors using ARMAX and NARMAX structures, Applied Mathematical Modelling 27 (2003) 611–627

Parameters n_a, n_b *and* n_c are the orders of the ARX model (inputs, output and white noise), whereas n_k is the delay.

Which can be compacted as,

$$y_{\text{model}}(i\Delta t) = -\sum_{j=1}^{n_a} a_j \cdot y(i\Delta t - j\Delta t) + \sum_{j=1}^{n_b} b_j \cdot u(i\Delta t - n_k\Delta t - j\Delta t + 1\Delta t) + \sum_{j=1}^{n_c} c_j \cdot e(i\Delta t - j\Delta t) + e(i\Delta t)$$

Or in a more compact form,

$$\boxed{y_{\text{model}}(i) = -\sum_{j=1}^{n_a} a_j \cdot y(i-j) + \sum_{j=1}^{n_b} b_j \cdot u(i-n_k-j+1) + \sum_{j=1}^{n_c} c_j \cdot e(i-j) + e(i)}$$

Taking into account the **SISO ARMAX model** we can define,

$u(i) = u(i)$ (single-input value defined in $t = i\Delta t$) *dim =1*

$P = [a_1,a_2,\ldots,a_{na},b_1,b_2,\ldots,b_{nb},c_1,c_2,\ldots,c_{nc}]^T$ the vector of coefficients with dim $=(n_a+n_b+n_c,1)$

$W_i = [-y(i-1),-y(i-2),\ldots-y(i-n_a),u(i-n_k),u(i-n_k-1),\ldots,u(i-n_k-n_b+1),e(i-1),e(i-2),\ldots,e(i-n_c)]^T$ the vector of outputs and inputs with dim= $(n_a+n_b+n_c,1)$

The **SISO ARMAX** model can be written in the form of a linear regression model as,

$$y_{model,I\,(1,1)} = P^T_{(1,na+nb+nc)} \cdot W_{i(na+nb+nc,1)}$$

The solution is given by $P_{optim} = (W^T \cdot W)^{-1} \cdot W^T \cdot Y_{exp}$

A more compact way to write the difference equation is:

$$A(q)y(t)=B(q)u(t-n_k)+C(q)e(t)$$

The parameters n_a, n_b, *and* n_c are the orders of the ARMAX model, and n_k is the delay, whereas q is the delay operator. Specifically,

$$A(q) = 1 + a_1q^{-1} + \ldots + a_{na}q^{-n_a}$$

$$B(q)=b_1+b_2q^{-1}+\ldots+b_{nb}q^{-n_b+1}$$

$$C(q)=1+c_1q^{-1}+\ldots+c_{nc}q^{-n_c}$$

If data is a time series that has no input channels and one output channel, the armax calculates an **ARMA model** for the time series

$$A(q)y(t)=C(q)e(t)$$

A **multi-input multi-output (MIMO)** system can be decomposed into several multi-input single-output (MISO) systems [62].

The general form of a **MISO ARMAX model** is as,

$$y_{\text{model}}(i) = -\sum_{j=1}^{n_a} a_j \cdot y(i-j) + \sum_{j=1}^{n_b} b_j \cdot u(i-n_k-j+1) + \sum_{j=1}^{n_c} c_j \cdot e(i-j) + e(i)$$

[62] J. Ding, C. Dong and Y. Yang, "Bias compensation based hierarchical parameter estimation for dual-rate sampled systems with colored noises," 2015 34th Chinese Control Conference (CCC), Hangzhou, 2015, pp. 1953-1957. doi: 10.1109/ChiCC.2015.7259930

$$y_{\text{model}}(i) = -\sum_{j=1}^{n_a} a_j \cdot y(i-j) + \sum_{j=1}^{n_b}\sum_{l=1}^{r} b_{jl} u_l(i-n_k-j+1) + \sum_{j=1}^{n_c} c_j \cdot e(i-j) + e(i)$$

where $e(i) = y_{model}(i) - y_{exp}(i)$

Where,

$u = [u_1,u_2,\ldots,u_r]^T$ is the inputs vector with r inputs

$W_i = [-y(i-1),-y(i-2),\ldots-y(i-n_a),u_1(i-n_k),u_1(i-n_k-1),\ldots,u_1(i-n_k-n_b+1),\ldots, u_r(i-n_k),u_r(i-n_k-1),\ldots,u_r(i-n_k-n_b+1), e(i-1),e(i-2),\ldots,e(i-n_c)]^T$ with dim $=(n_a+r\cdot n_b+n_c,1)$

$P = [a_1,a_2,\ldots,a_{na},b_{11},b_{21},\ldots,b_{1nb},\ldots, b_{r1},b_{r2},\ldots,b_{rnb}, c_1,c_2,\ldots,c_{na}]^T$ is the vector of parameters with dim= $(n_a+r\cdot n_b+n_c,1)$

The **MISO ARMAX** model can be expressed as a linear regression model as,

$$y_{model,i\,(1,1)} = P^T_{(1,na+r\cdot nb+nc)}\, W_{i(na+r\cdot nb+nc,1)}$$

The solution is given by $P_{optim} = (W^T\cdot W)^{-1}\cdot W^T\cdot Y_{exp}$

6.2 Nonlinear parameter fitting methods

Nonlinear systems are usually defined as any system which is not linear, that is any system that does not satisfy the superposition principle[63]. Most systems encountered in the real world are nonlinear in nature, but linear models cannot capture the rich dynamic behavior associated with nonlinear systems. It is imperative to have identification techniques which are specific for nonlinear systems.

The **NARX** (nonlinear ARX) model can be used to represent a nonlinear process. The nonlinear NARX model of a finite dimensional system is defined by[64]

$$y(i) = f_N[y(i-1),\ldots,y(i-n_a),u(i-n_k),\ldots,u(i-n_k-n_b+1)] + e(i)$$

where $y(i)$ is the autoregressive (AR) variable or system output; $u(i)$ is the exogenous (X) variable or system input; $e(i)$ is the moving average (MA) variable or **white noise**, i.e. $E_t[e(t)]=0$ and $E_t[e(t)e(t-i)]=\delta_i\sigma_a^2$; f_N is a nonlinear function; n_a and n_b, are the AR and X orders, respectively; δ_i is the Kronecker delta.

[63] Stephen A Billings, Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains, John Wiley & Sons, Ltd., 2013

[64] Alireza Rahrooh, Scott Shepard, Identification of nonlinear systems using NARMAX model, Nonlinear Analysis 71 (2009) pp.1198–e1202

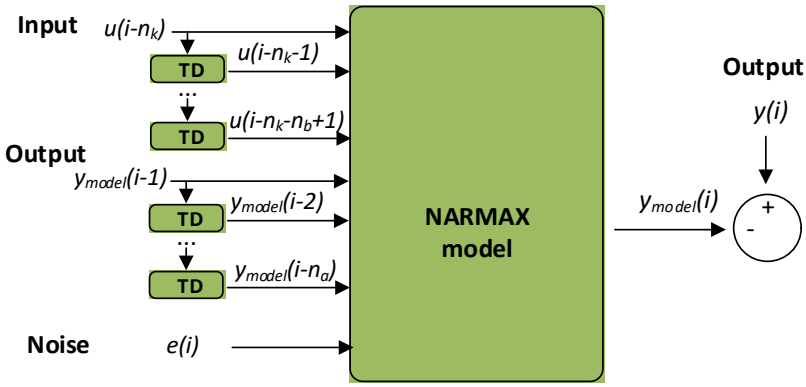


Fig. 6.4. **NARX** representation being $e(i) = y(i) - y_{model}(i)$

The NARX model is essentially an expansion of past inputs, outputs, and a noise term. The functional description of an actual system can be very complex and the explicit form of this functional is usually unknown. Therefore, the practical modeling of real signal must be based upon a chosen set of the unknown functions.

For example, a NARX model as,

$$y(i) = -a_1 \underbrace{y(i-1)}_{x1} - a_2 \underbrace{y(i-2)y(i-3)}_{x2} + b_{11} \underbrace{u_1(i-3)u_1(i-2)}_{x3} + b_{12} \underbrace{u_1(i-1)u_1(i-2)}_{x4} + b_{21} \underbrace{u_2(i-3)u_2(i-2)u_2(i-1)}_{x5} + e(i)$$

Leads to

$W_i = [-y(i-1), -y(i-2)y(i-3), u_1(i-3)u_1(i-2), u_1(i-1)u_1(i-2), u_2(i-3)u_2(i-2)u_2(i-1)]^T$ with dim =(5,1)

$P = [a_1, a_2, b_{11}, b_{12}, b_{21}]^T$ is the vector of parameters with dim= (5,1)

The MISO NARX model can be expressed as a linear regression model as,

$$\boldsymbol{y_{model,i\,(1,1)} = P^T_{(1,na+r\cdot nb)}\, W_{i(na+r\cdot nb,1)}}$$

The solution is given by $\boldsymbol{P_{optim} = (W^T \cdot W)^{-1} \cdot W^T \cdot Y_{exp}}$

Similarly, the **NARMAX** (nonlinear ARMAX) model is used to represent a nonlinear process. The nonlinear NARMAX model of a finite dimensional system is defined by[65]

$$y(i) = f_N[y(i-1),\ldots,y(i-n_a),u(i-n_k),\ldots,u(i-n_k-n_b+1),e(i-1),\ldots,e(i-n_c)] + e(i)$$

where $y(i)$ is the autoregressive (AR) variable or system output; $u(i)$ is the exogenous (X) variable or system input; $e(i)$ is the moving average (MA) variable or **white noise**, i.e. $E_t[e(t)]=0$ and $E_t[e(t)e(t-i)]=\delta_i\sigma_a^2$; f_N is a nonlinear function; n_a, n_c and n_b, are the AR, MA and X orders, respectively; δ_i is the Kronecker delta.

[65] Alireza Rahrooh, Scott Shepard, Identification of nonlinear systems using NARMAX model, Nonlinear Analysis 71 (2009) pp.1198–e1202

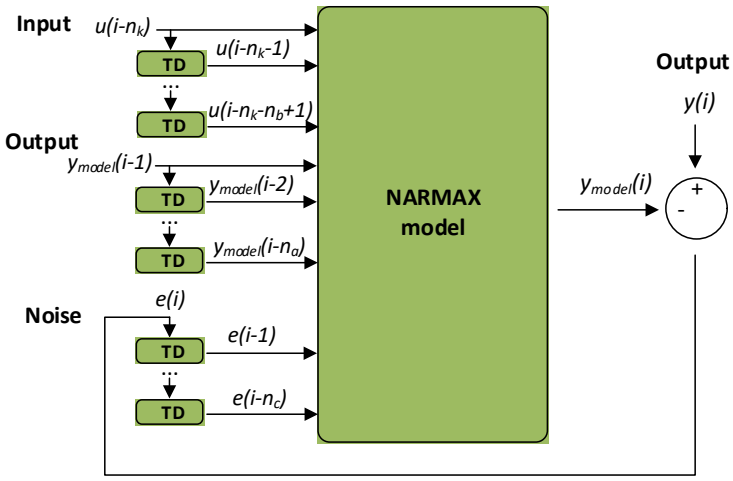


Fig. 6.5. **NARMAX** representation[66] being $e(i) = y(i) - y_{model}(i)$

The NARMAX model is essentially an expansion of past inputs, outputs, and noise terms. In a NARMAX description of the system, it is modeled in terms of a nonlinear functional expansion of lagged or past input, output and prediction errors. The functional description of an actual system can be very complex and the explicit form of this functional is usually unknown. Therefore, the practical modeling of real signal must be based upon a chosen set of the unknown functions.

NARMAX models, where the nonlinear AR and ARX models are a special case, are the most popular representations for nonlinear dynamic system identification in the discrete-time domain. The continuous-time domain counterparts of the ARMAX and NARMAX models are ordinary differential equations (ODEs), which can be either linear or nonlinear.

6.2.1 Polynomial NARMAX model

The most commonly used NARMAX model is the power-form polynomial representation [40],

$$y_{\text{model}}(i) = p_0 + \sum_{i_1=1}^{n} f_{i_1}((x_{i_1}(i)) + \sum_{i_1=1}^{n}\sum_{i_2=i_1}^{n} f_{i_1 i_2}(x_{i_1}(i), x_{i_2}(i)) + \ldots + \sum_{i_1=1}^{n}\ldots\sum_{i_l=i_{l-1}}^{n} f_{i_1 i_2 \ldots i_l}(x_{i_1}(i), x_{i_2}(i), \ldots, x_{i_l}(i)) + e(i)$$

where l is the degree of polynomial nonlinearity, $\pi_{i1i2\ldots im}$ are model parameters, $n=n_a+ n_b+n_c$, and

$$f_{i_1 i_2 \ldots i_m}(x_{i_1}(k), x_{i_2}(k), \ldots, x_{i_m}(k)) = p_{i_1 i_2 \ldots i_m} \prod_{k=1}^{m} x_{i_l}(k) \quad 1 \le m \le l$$

$$x_m(i) = \begin{cases} y(i-m) & 1 \le m \le n_a \\ u(i-(m-n_y)) & n_a+1 \le m \le n_a+n_b \\ e(i-(m-n_a-n_b)) & n_a+n_b+1 \le m \le n_a+n_b+n_c \end{cases}$$

More specifically, the polynomial NARMAX model can be represented as,

[66] G. Acuña, C. Ramirez, M. Curilem, Comparing NARX and NARMAX models using ANN and SVM for cash demand forecasting for ATMWCCI 2012 IEEE World Congress on Computational Intelligence June, 10-15, 2012 - Brisbane, Australia

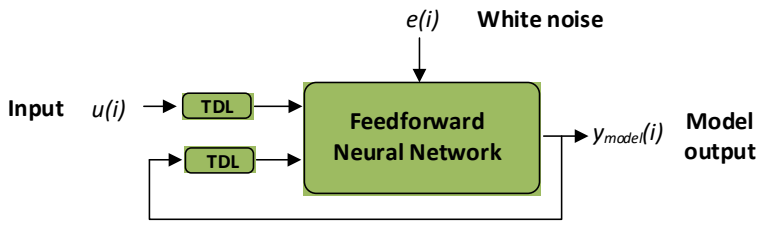
$$y_{\text{model}}(i) = p_0 + \sum_{i_1=1}^{n} p_{i_1} x_{i_1}(i) + \sum_{i_1=1}^{n}\sum_{i_2=i_1}^{n} p_{i_1 i_2} x_{i_1}(i) x_{i_2}(i) + \ldots + \sum_{i_1=1}^{n}\ldots\sum_{i_l=i_{l-1}}^{n} p_{i_1 i_2 \ldots i_l} x_{i_1}(i) x_{i_2}(i) \ldots x_{i_l}(i) + e(i)$$

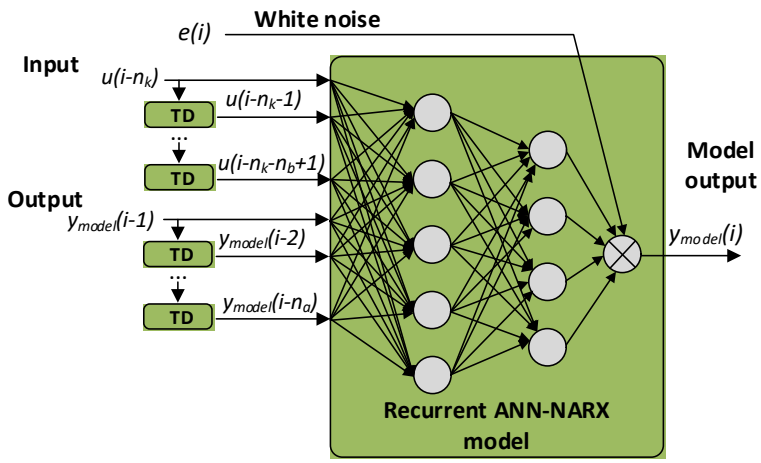
The degree of a multivariate polynomial is defined as the highest order among the terms. For example, the degree of the polynomial $h(x_1, x_2, x_3) = a_1 x_1^3 + a_2 x_1 x_2 + a_3 x_1^2 x_3^2$ is $l = 4$.

6.2.2 Parameter fitting based on NN (neural networks)

Processes with nonlinear dynamics often offer challenging difficulties to obtain accurate results using classic autoregressive methods. Alternative methods based on neural networks (NN) are being used for this purpose.

NN-based approaches can deal with the nonlinear processes based on experimental data. Fig. 6.6 shows a three-layer feed-forward neural network, since, in general, three layers are sufficient, although to improve convergence during the learning process additional hidden layers can be required at the expense of increasing the computational burden [67]. The basic structure, having one hidden layer with sigmoid function, is known to be powerful enough to produce an arbitrary mapping among variables[68].





[67] Eric H.K. Fung, Y.K. Wong, H.F. Ho, Marc P. Mignolet, Modelling and prediction of machining errors using ARMAX and NARMAX structures, Applied Mathematical Modelling 27 (2003) 611–627

[68] J.A. Freeman, D.M. Skapura, Neural Networks: Algorithms, Applications and Programming Techniques, Addison-Wesley, Reading, MA, 1991.

Fig. 6.6. The structure of a neural network NARX model with a hidden layer[69]. TD: Tapped Delay

The following steps explain the calculation of the NN output based on the input vector.

- Step 1: Assign $w^T(i)$ to the input vector $x^T(i)$, where $w^T(i)$ is the data vector given by (4).
  $w^T(i) = [y(i-1), y(i-2),\ldots y(i-n_a),u(i-n_k),u(i-n_k-1),\ldots,u(i-n_k-n_b+1),e(i-1),e(i-2),\ldots,e(i-n_c)]$
  The initial values of ε(-1),ε(-2) … and ε(-n) are set to zero.

- Step 2: Calculate the input of the hidden (*h*) layer units as,

$$net_j^h(i) = \sum_{k=1}^{p} w_{jk} x_k(i) + b_j^h$$

  $p = n_a+n_k+n_c$ being the number of input nodes of the network, *j* is the *j-th* hidden unit, w_{jk}^h is the connection weight between k-th input unit and j-th hidden unit and b_j^h is the bias term of the j-th hidden unit.

- Step 3: Calculate the output from a node in the hidden layer $z_j = f_j^h[net_j^h(i)] = f[\sum_{k=1}^{p} w_{jk} x_k(i) + b_j^h]$ where f_j^h can be a sigmoid function defined by $f_j^h[net_j^h(i)] = \frac{1}{1+\exp[-net_j^h(i)]}$

- Step 4: Calculate the input to the output nodes as, $net_l^o(i) = \sum_{m=1}^{h} W_{lm}(i) z_m(i)$, where *l* is the l-th output unit, $W_{lm}^o(i)$ is the connection weight between m-th hidden unit and l-th output unit.

- Step 5: Calculate the outputs from the output nodes as, $zz(i) = f_l^o[net_l^q(i)]$, where f_l^o is the linear activation function defined by $f_l^o[net_l^o(i)] = net_l^o(i)$

Similarly, the NN-NARMAX model can be represented as,

[69] S. Shan and Z. Hou, "Neural Network NARMAX Model Based Unmanned Aircraft Control Surface Reconfiguration," 2016 9th International Symposium on Computational Intelligence and Design (ISCID), Hangzhou, 2016, pp. 154-157. doi: 10.1109/ISCID.2016.1043

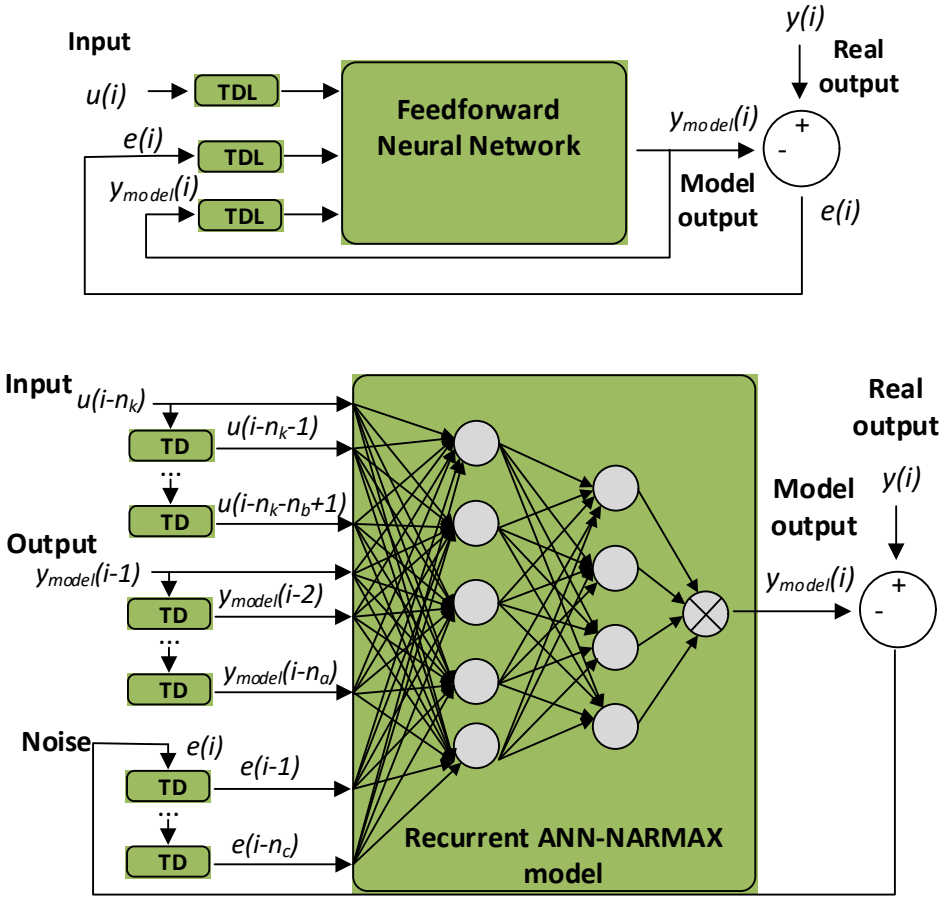


Fig. 6.7. The structure of a neural network NARMAX model with a hidden layer. TD: Tapped Delay

6.2.3 Parameter fitting based on GA (genetic algorithms)

Genetic algorithms (GAs) are well fitted to deal with discrete and non-linear optimization problems. GAs fall within the category of evolutionary methods, which solve search and optimization problems[70]. GAs apply mathematical operations to imitate genetic reproduction mechanisms. They include individual's selection (are the offspring solutions for the next generation), and mutation (it produces random changes to parents and children solutions) to attain an optimal solution. The model parameters are found during the optimization process, are coded into binary strings forming the chromosomes. The crossover operation is applied, which at each generation produces a random binary vector from which the genes of both parents are combined to form the offspring. Next, the mutation process is applied, by which the GA creates small random variations in each bit of the chromosomes,

[70] Jinyao Yan, J.R. Deller Jr, NARMAX model identification using a set-theoretic evolutionary approach, Signal Processing 123 (2016) 30–41.

providing genetic diversity and enabling GA to search for wider spaces. The whole process is repeated until an established tolerance criterion is achieved[71].

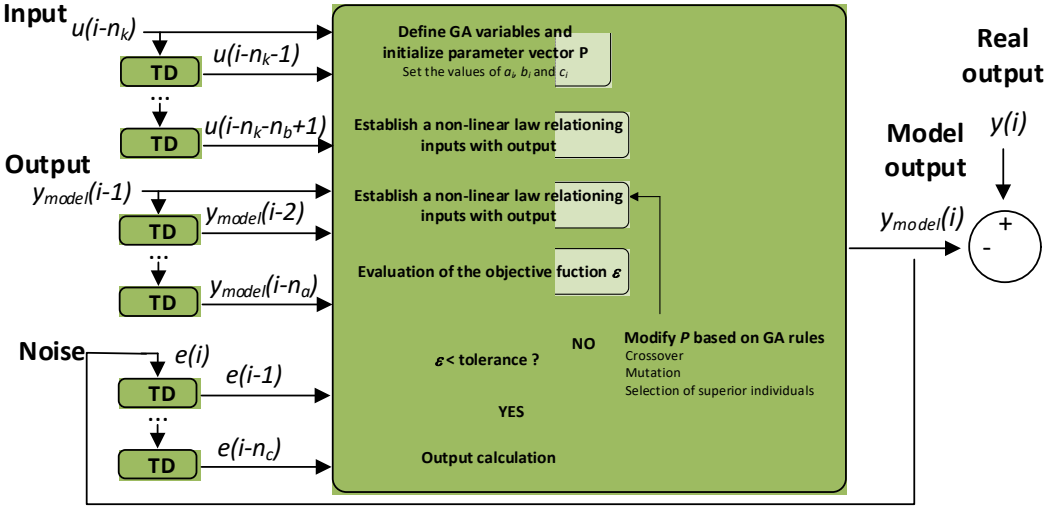


Fig. 6.8. The structure of a GA-NARMAX model. TD: Tapped Delay

6.2.4 Parameter fitting based on PSO (particle swarm optimization)[72]

PSO methods tend to reproduce the behavior of a swarm, representing a simplified social, since the. Like, the social behavior of different organisms including fish schooling and bird flocking can be analyzed as an optimization problem. PSO methods try to find the best position or state with time of each particle within a multidimensional space. During swim or flight, each particle change its position depending on its own experience and the neighbors' experiences. To this end PSO take into account the current position and velocity as well as the best previous positions attained by each particle and its neighbors[73]. This strategy is aimed at moving the swarm toward the best solution.

Once selected the structure of the best ARMAX model, the model parameters estimated by using a PSO-based parameter estimation process as follows,

- 1.- Randomly generate the initial trial parameters vectors P_i over the d-dimensional space.
- 2.- Calculate the loss function for each parameter vector P_i. In this step, a number of tentative ARMAX models with the same model structure but with different combinations of parameter vector have to be established. The loss function t is adopted to represent the

[71] H. Saavedra, J.-R. Riba, L. Romeral, Multi-objective Optimal Design of a Five-Phase Fault-Tolerant Axial Flux PM Motor, Advances in Electrical and Computer Engineering (AECE), vol. 15, no. 1, pp. 69-76, Jan. 2015.

[72] Bo Wang, Neng-ling Tai, Hai-qing Zhai, Jian Ye, Jia-dong Zhuc, Liang-bo Qi, A new ARMAX model based on evolutionary algorithm and particle swarm optimization for short-term load forecasting, Electric Power Systems Research 78 (2008) 1679–1685.

[73] Chao-Ming Huang, Chi-Jen Huang, and Ming-Li Wang, A Particle Swarm Optimization to Identifying the ARMAX Model for Short-Term Load Forecasting, IEEE Transactions On Power Systems, 20(2), May 2005

objective function of parameter, being expressed as, $L_o = e'_f x e_f/(H-h)$, e_f (dimension Hx1) being the simulated error vector of the historical data points, and e'_f is the transpose of e_f. h is the number of parameters in the fitted model and H is the number of historical data points.

- 3.- Each particle or parameter vector P_i memorizes its own loss function and designates the minimum one, $P_{best,i}^d$ (vector of dimension d). The particle with the best loss function among P_{best} is denoted as G_{best}. It is noted that in the first iteration, each particle is set directly to $P_{best,i}^d$, and the particle with the best loss function among P_{best} is set to G_{best}.
- 4.- Change the velocity of each particle according to $L_o = e'_f x e_f/(H-h)$. An offspring vector is then created by applying $P_i^d(t+1)= P_i^d(t)+ v_i^d(t+1)$ in the range $[a,b]$. If $P_i^d(t+1) > P_{max}^d$, then $P_i^d(t+1) = b$, and if $P_i^d(t+1) < P_{min}^d$, then $P_i^d(t+1) = a$.
- 5.- Repeat steps 2.- to 4.- to obtain the best solution in the population.

The solution with the best loss function is chosen as the tentative candidate model after passing further accuracy checks. Otherwise, the process defined above must be iterated until reaching an appropriate solution.

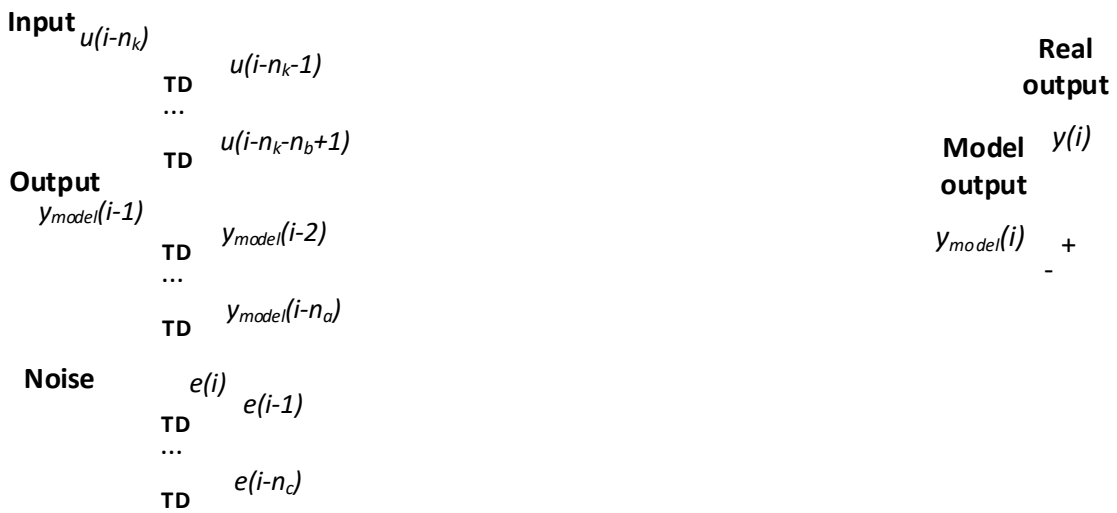


Fig. 6.9. The structure of a PSO-NARMAX model.