22nd International Conference on Renewable Energies and Power Quality (ICREPQ'24)
Bilbao (Spain), 26th to 28th June 2024
Renewable Energy and Power Quality Journal (RE&PQJ)
ISSN 2172-038 X, Volume No.22, September 2024

An Adaptive Secondary Control Based on Recursive Neural Networks for Isolated Microgrids

L. Rodríguez[1], A. Pantoja[1], J. Revelo[1], and J. Barco-Jiménez[2,3,4]

[1] Department of Electronics
University of Nariño
[2] Program of Electronic Engineering
University CESMAG
[3] Postgraduate Programs in Electrical and Electronic Engineering - PPIEE
University of Valle

[4] Electrical Engineering Department
University of Malaga

Abstract. This work proposes a model reference adaptive control based on recursive neural networks. This secondary-level controller corrects the deviations on the voltage and frequency setpoints of a simple primary control in an isolated microgrid (MG) with multiple distributed generators (DGs). The controller has two nonlinear autoregressive with external input neural networks to emulate the microgrid and guide the whole system according to an appropriate reference model. The networks are trained with synthetic data from a simulation MG and the order of the networks are obtained with a deterministic method. The method is tested with a three-units MG in a Matlab simulation under different working conditions. Results show the proper performance of the proposed controllers in comparison to a PI strategy.

Key words. Model reference adaptive control, microgrids control, recursive neural networks, secondary control.

1. Introduction

The inclusion of distributed generators (DGs) presents a challenge in isolated microgrids. The DGs are generally controlled and connected to the microgrid through on-grid and off-grid inverters. Therefore, the control loop of these devices must guarantee obtaining stability, satisfying the microgrid's loads, and maintaining power quality in key variables, such as frequency and voltage [1].

To ensure proper operation of microgrids, a hierarchical control scheme is proposed, achieving operation objectives by three main levels. The first level is given by local current and voltage control loops, used to ensure power sharing capability and achieve basic control of voltage and frequency of each generator [2]. However, when disturbances or anomalies appear, the local controller is unable to guarantee the setpoints of the main variables. The secondary control level is used to deal with frequency and voltage deviations by adjusting the preset values of the current setpoints at the first level. Finally, the tertiary level is used for the cost optimization and management of the energy flow in the microgrid. Normally, the primary and tertiary levels require decentralized and centralized controllers, respectively, while the secondary level can be implemented using both structures.

On the other hand, it is well known that in real microgrids there exist transient phenomena due to abrupt changes in the loads, failures in the power units, and the incorporation of DGs with stochastic resources. Therefore, high variations in voltage and frequency are produced, including increases in harmonic distortions that affect system stability and power quality. These issues are addressed mainly by a secondary controller.

Nowadays, several studies have focused on solving the aforementioned problems. For instance, [3] shows PI controllers, whose coefficients are determined by particle swarm optimization (PSO) to improve the dynamic and steady-state voltage and frequency dynamic and the steady-state behaviour of the system. Similarly, distributed controllers have been proposed, mostly reliant on consensus techniques. In [4], a distribution optimization problem is solved to achieve a convergent solution for all distributed units. In other studies, multiagent-based load restoration algorithms have been proposed [5], whose agents make decisions based on local information from direct neighbours and global information based on the average consensus method.

For highly nonlinear and complex AC/DC microgrids (with switched power converters), control schemes based on machine learning techniques, such as adaptive neural networks (NN) and evolutionary algorithms are gaining widespread interest. Intelligent controllers are very promising because they can adapt to uncertainties and can also be used when the model of the system is not available.

Recently, recursive neural networks with learning capabilities have been widely applied to the control of complex power systems [6].

Additionally, microgrids with low inertia and high penetration of renewable sources require complex control structures to maintain frequency and voltage within an acceptable range. Alternatively, in [7], a novel secondary controller based on reinforced deep learning is presented. This controller offers advantages over traditional ones as it responds better to the nonlinear dynamics of the microgrid, outperforming droop controllers in some cases. Similarly, in [8], a reinforcement learning-based controller is introduced, which does not require model information but is trained under feedback control based on Lyapunov principles. Simulation results demonstrate that the neural secondary controller responds very well to changes in line impedance, resistive or inductive loads, and uncertainty in other parameters. In this regard, neural controllers show promising results in addressing various issues affecting the balance of frequency and voltage in microgrids.

Based on the above considerations, this study proposes a secondary control into a microgrid composed of several distributed generators with a basic droop primary controller. The system must be able to guarantee the control of voltage and frequency at the point of common coupling (PCC) under conditions of load variation, failures in the power generation sources and changes on the resources of DGs. The main contributions are:

- The proposal of an adaptive secondary control based on recursive neural networks capable of regulating the voltage and frequency. The design and calibration do not depend on an explicit model of the system.
- A comparison of the implemented controller with a conventional centralized secondary control.
- The validation of the controller in simulation scenarios that generate high variations in voltage and frequency.

2. Hierarchical Control Structure for a Microgrid

A microgrid can be modelled as several sources connected through lines and local loads in the nodes. A typical DG unit (e.g., a photovoltaic generator) consists of a DC main source, a DC/AC converter, and an LCL filter, as it is shown in Fig. 1. In an isolated microgrid with multiple DGs, the voltage and frequency are supported by the generators, and the load consumption is assumed to be distributed among the units according to their nominal power.

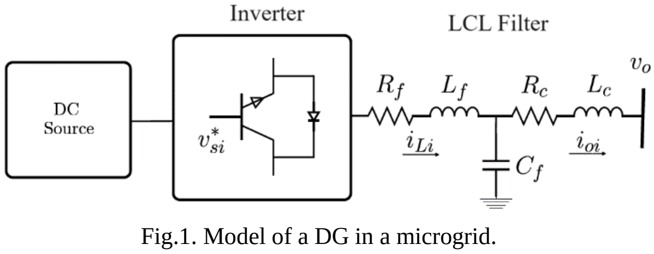

Fig.1. Model of a DG in a microgrid.

A. Primary Control

The basic control strategy in a primary level is the droop controller. Based on the measured active and reactive power information, the droop relationship provides the operating frequency for the inverter as well as the voltage reference for the voltage controller. In this case, each DG has an appropriately tuned droop control.

B. Secondary control

The secondary control layer coordinates the operation of multiple DERs within the microgrid to optimize system performance and achieve specific objectives, such as reducing the voltage and frequency deviation. It includes a centralized control system that communicate with the individual DERs and adjust their output based on system requirements. This layer typically operates on a second-to-minute time scale.

To compensate for the voltage and frequency deviation caused by the droop control and to overcome the impedance impact of the transmission line on the power assignment, the secondary voltage and frequency control intervene to adjust the reference of the primary control through the inclusion of two control signals (u_ω, u_v), which are added to the control signals in the droop strategy as

$$\omega^* = \omega_n - R_p P + u_\omega$$
$$v^* = v_n - R_q Q + u_v,$$

where $\omega^*$and v^* are de desired values for frequency and voltage, $R_p$and R_q are the tuned droop constants, and P, Q, ω_n, v_n are the active/reactive power and the frequency and voltage measured in the generation node.

3. A MRAC Secondary Control

The architecture of the proposed Model Reference Adaptive Control (MRAC) uses a neural network for the main controller and a second neural network to emulate the plant (i.e., the microgrid model). Fig.2 shows the architecture, where $y_{ref}(k+1)$ is the output of the reference model that the pair controller-plant must follow, $y_p(k+1)$ is the microgrid output (voltage or frequency signals), and $e_c(k)$ is the tracking error to be reduced. In this structure, the neural network controller provides an appropriate control signal $u(k)$ to the system to keep the output as close as possible to the desired reference output, specified by the reference model with input $r(k)$. The tracking error is used to adjust the parameters of the neural network in the controller. The reference model establishes the expected performance of the closed-loop system and must be carefully chosen so that the real closed-loop system may be able to achieve a behaviour like the required one.

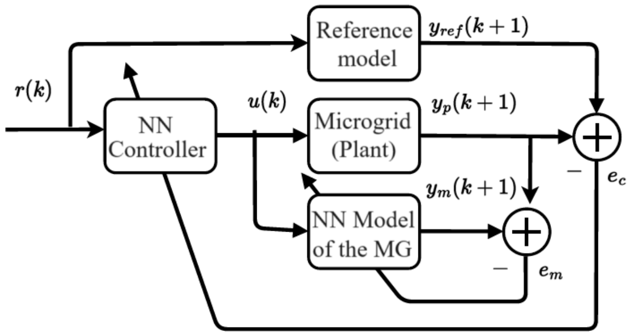

Fig.2. MRAC control structure.

A. Nonlinear autoregressive with external input Neural Networks (NNARX)

Given the complex dynamics of the microgrids and the expected controller, NNARX are interesting topologies to emulate nonlinear and hard-to-model behaviours. These networks are dynamic and recurrent, consisting of several layers with feedback. The main structure whose structure is shown in the upper (or bottom) schematics of Fig.3. They are based on the autoregressive model used to describe systems with inertia, where the predicted values depend on previous outputs and external input values on the dependent variable. The network's outputs, their respective inputs, and the desired references are input data in the training process [9].

The controller (using both neural networks) can undergo either online or offline training procedures. In this scenario, offline training was employed. Fig. 3 illustrates the neural structure utilized in the offline training process, comprising four layers (two for each network) that represent a closed-loop control system.

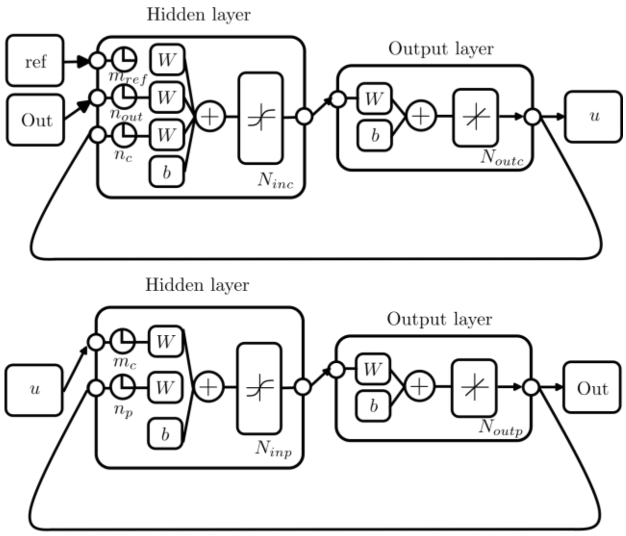

Fig.3. MRAC training controller structure. The controller NN is presented in the upper schematic, while the model NN is in the bottom.

The upper first two layers represent the neural controller, while the last two layers approximate the plant to be controlled. Here, m_{ref} and m_c denote the delays associated with the desired references and the control signal, respectively. Moreover, $n_{out,}$ n_c, and n_p represent the number of delays related to the output signal of the neural network, the output of the controller layers, and the output of the layers representing the system.

Notice that training can be performed independently if we have appropriate data for each network. Consequently, we first train the NN to emulate the dynamics of the system (NN model of the MG), and then, utilize this model to train exclusively the layers corresponding to the controller.

The NN for control needs to learn an emulation of the reference model chosen to lead the behaviour of the MG. Then, the controller is trained to produce the proper input to the MG (u) to a reference when the plant produces the output (i.e., another input of the controller). This training uses a "synthetic" closed-loop system, requiring data solely from the NN plant and a reference model.

B. NNARX training and parameter tuning

The first NNARX to be trained is the one corresponding to the microgrid emulation. The training is data-based, so the plant is simulated with random inputs and the outputs are measured to obtain the plant database. With an appropriate model, the control is trained with the pairs input-output obtained from the desired reference model (in this case, a two-order linear model, with unitary gain to avoid steady-state error, a damping factor of 0.8, and a natural frequency of 11.1 rad/s are used). The control and the model of the plant interacts in this second phase, but only the weights of the controller are tuned to achieve the response of the reference model.

To train the neural network layers for the plant, the initial step involves selecting input and output variables capturing the microgrid dynamics. This study prioritizes secondary control signals, notably the park-transformed voltage signal and microgrid frequency, sampled at the common coupling point (PCC) every 0.5 ms.

Obtaining the microgrid´s output data requires a simulation model to introduce the input signals. In this case, and based on simulation tests, the optimal intervals for voltages are in [6, 20] V and [1.1, 2] rad/s for frequency, with varying periods from 2 to 50 ms. Due to the MG's sensitivity, input signals undergo processing via a moving average filter.

After signal acquisition, processing becomes essential. As in [22], signals are normalized to achieve a zero mean and a standard deviation of 1. The process for a voltage input is depicted in Fig. 4, displaying a pseudorandom input signal, their corresponding response, and its prepressed result.

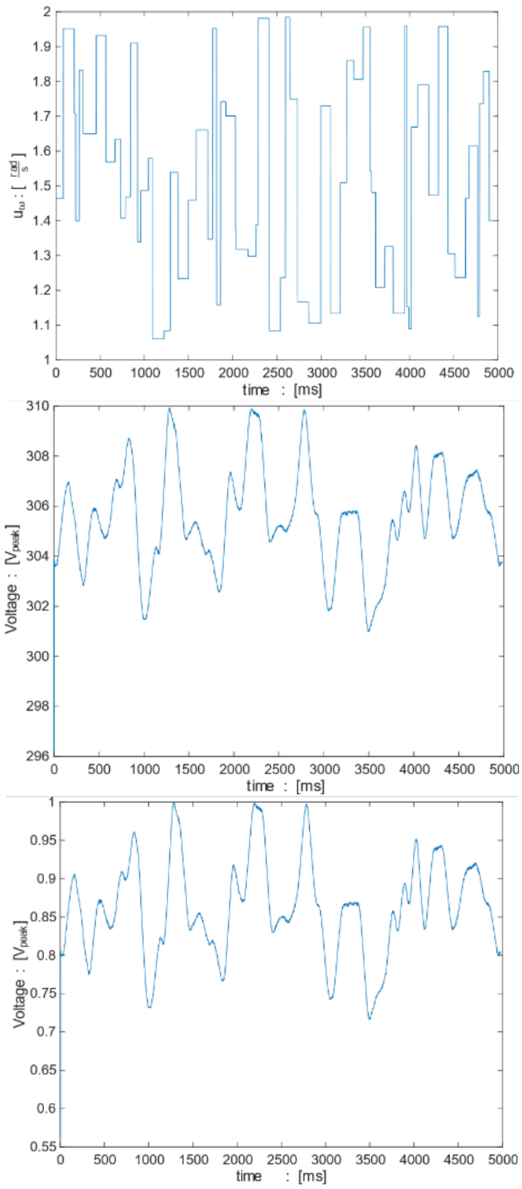

Fig. 4. Response curves from top to bottom: (a) u_v, (b) the generated voltage response, and (c) the pre-processed voltage signal to train the model NN.

C. *NNARX model order calculations*

The model order (delays of inputs and outputs of the NNARX) is performed by the Lipschitz number method, which is a deterministic method that makes use of the geometrical information of the input-output data. The method can be used for all ARX model structures and linear or nonlinear systems. The method of Lipschitz numbers [11] is based on the continuity property of the functions representing the input-output models and does not depend on the approximation method or model structure. This method involves determining a Lipschitz constant, which quantifies the maximum rate of change of the function over its entire domain. This constant helps in establishing the stability and convergence properties of mathematical equations and optimization algorithms.

After determining the model's order using the Lipschitz number method in [11], the next step involves identifying additional parameters such as the number of layers and the size of each layer. These hyperparameters are fine-tuned through trial and error until the optimal configuration within a predefined range is found. During training, only one parameter of the neural network is adjusted and calibrated at a time to maintain stability in performance. The remaining parameters are kept constant during calibration. It's worth noting that this procedure is repeated for each parameter of the network.

4. Simulation Results and Analysis

To test the effectiveness of the proposed approach, a standalone MG with three DG units with individual loads and transmission lines (Fig. 5) is simulated in MATLAB. To compare the response of the proposed controller, three secondary controllers are implemented: a PI simple controller, a MRAC with a simple database, and a complex MRAC with a database obtained with multiple events, failures, and changes in loads. Detailed parameters for the MG model and secondary control are provided in Table I and II, respectively.

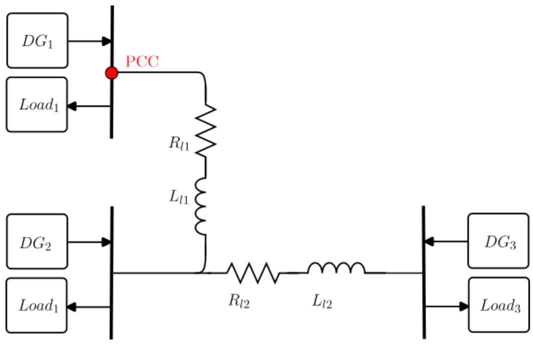

Fig. 5. General schematic of the MG test system.

Table I. - Parameters of the test network

	DG1 & DG3		DG2	
Model	Rp1,3	9.4e-5	Rp2	12.5e-5
	Rq1,3	1.3e-3	Rq2	1.75e-3
	R_f	100 mΩ	L_f	0.30 H
	R_c	30 Ω	L_c	0.30 H
Load	R_{Load}	10 Ω	L_{Load}	0.01 H
	Z_{12}		Z_{23}	
Line	R_{l1}	230 mΩ	L_{l1}	0.32 μH
	R_{l2}	350 mΩ	L_{l2}	1.84 μH

Table II. - Controller parameters

	Param	Value	Param	Value
Simple	m_{ref}	6	n_{out}	10
	n_c	6	N_{inc}	6
	m_c	10	n_p	10
	N_{inp}	2	$\boldsymbol{N_{outp}}$	1
Complete	$\boldsymbol{m_{ref}}$	6	$\boldsymbol{n_{out}}$	10
	$\boldsymbol{n_c}$	6	$\boldsymbol{N_{inc}}$	6
	$\boldsymbol{m_c}$	10	$\boldsymbol{n_p}$	10
	$\boldsymbol{N_{inp}}$	2	$\boldsymbol{N_{outp}}$	2
	Voltage		Frequency	
Reference	V^*	311 V_{peak}	f^*	60 Hz

To analyse the performance of the controllers, we set up five operating scenarios: *i)* The system operates with only the primary controller in the time interval [0 0.5) s; *ii)* in *t=0.5 s*, the secondary controller is activated; *iii)* in *t=1 s*, the load on node three is doubled; *iv)* in time 1.5 s, the load on node 3 is halved; and, *v)* at *t=2 s*, the DG_3 is disconnected from the system and reconnected at second 3.

Simulation results (Fig. 6 and 7) unveil performance disparities among controllers across stages. Initially (0 - 0.5 s), all generators exhibit voltage and frequency deviations due to primary control. Upon secondary control activation at t=0.5 s, voltage deviations diminish, with the PI controller showing the lowest steady-state error, trailed by the simple and complex neural controllers. Similarly, the PI controller tracks frequency closely compared to neural controllers.

During load fluctuations (1-2 s), the PI controller demonstrates accurate voltage tracking in DG_1 and DG_2, while the simple neural controller excels in DG_3. However, the complex neural controller fares poorly across all generators. In terms of frequency, the PI controller yields optimal results with longer transients compared to neural controllers.

During the disconnection and reconnection of DG_3 (2-3 s), higher overshoot occurs due to system perturbations. Notably, the complex neural controller exhibits the best response during DG_3 reconnection, while the PI controller shows the poorest performance.

To compare the controllers, some indexes are calculated based on the responses in all the scenarios. The analysis of Table III reveals that the complex neural controller requires the least control effort (ISU), followed by the simple neural and PI controllers. However, graphical analysis alone is insufficient for determining controller performance. Other metrics indicate the complex neural controller performs best in frequency, achieving the lowest IAE (integral of the absolute error) and ISE (integral of the squared error). Conversely, for voltage, it fares poorly, while the PI controller exhibits the highest ISE.

Table III. - Comparison Metrics

	Index	PI	SIMPLE	COMPLETE
Frequency	IAE	0.325 M	0.685 M	0.310 M
	ISE	0.167 M	0.108 M	39.22 k
	ISU	6.497 M	10.25 M	4.172 M
Voltage	IAE	21.88 M	19.86 M	38.46 M
	ISE	0.879 G	0.964 G	1.037 G
	ISU	40.62 M	31.08 M	7.340 M

Comparatively, the full neural controller demonstrates superior energy performance, evident in its lower ISU across all generators. These findings underscore neural controllers' superior performance, attributed to their excellent frequency response and overall energetic behaviour.

4. Conclusion

This paper focuses on the secondary control layer for voltage and frequency restoration in autonomous MGs, using as an approach a centralized controller based on machine learning. Additionally, simulations of the controller in different scenarios and trained with a varying amount of data are presented. Simulations validate the effectiveness of the proposed method for secondary control under conditions with load disturbances and its plug-and-play operation. Moreover, the results show the efficiency of the proposed algorithm, and its simplicity are interesting for implementation. There is, however, an important tradeoff obtaining the database to train the NNARXs, although some parameters like the number of neurons can be obtained with deterministic methods.

Acknowledgement

This work is supported in part by Project BPIN 2021000100499, Asignación para la CTeI del SGR, Colombia.

References

[1] D. S. Burbano-Benavides, O. D. Ortiz-Sotelo, J. Revelo-Fuelagán, and J. E. Candelo-Becerra, “Design of an on-grid microinverter control technique for managing active and reactive power in a microgrid,” Applied Sciences, vol. 11, no. 11, 2021.
[2] M. H. Cintuglu, T. Youssef, and O. A. Mohammed, “Development and application of a real-time testbed for multiagent system interoperability: A case study on hierarchical microgrid control,” IEEE Transactions on Smart Grid, vol. 9, no. 3, pp. 1759–1768, 2018.
[3] M. A. Ebrahim, B. A. Aziz, F. A. Osman, and M. N. F. Nashed, “Optimal PI based secondary control for autonomous micro-grid via particle swarm optimization technique,” in 2018 Twentieth International Middle East Power Systems Conference (MEPCON), pp. 1148–1155, 2018.
[4] M. Yazdanian and A. Mehrizi-Sani, “Distributed control techniques in microgrids,” IEEE Transactions on Smart Grid, vol. 5, no. 6, pp. 2901–2909, 2014.
[5] A. Parisio, E. Rikos, and L. Glielmo, “A model predictive control approach to microgrid operation optimization,” IEEE Transactions on Control Systems Technology, Vol. 22, no. 5, pp. 1813–1827, 2014.
[6] S. K. Sahoo, A. K. Sinha, and N. K. Kishore, “Control techniques in AC, DC, and hybrid AC/DC microgrid: A review,” IEEE Journal of Emerging and Selected Topics in Power Electronics, Vol. 6, no. 2, pp. 738–759, 2018.
[7] P. I. N. Barbalho, V. A. Lacerda, R. A. S. Fernandes, y D. V. Coury, “Deep reinforcement learning-based secondary control for microgrids in islanded mode”, Electr. Power Syst. Res., Vol. 212, no 445374, p. 108315, 2022.
[8] M. Adibi y J. Van Der Woude, “Secondary Frequency Control of Microgrids: An Online Reinforcement Learning Approach”, IEEE Trans. Automat. Contr., Vol. 67, no 9, pp. 4824-4831, 2022.
[9] G. Julio, O. Leony, A. Alexander, P. Wilson, and R. Milton, “An hybrid algorithm based NARX for non-linear identification and modelling of an AC/DC hybrid microgrid simulation,” in 2022 IEEE Sixth Ecuador Technical Chapters Meeting (ETCM), pp. 1–6, 2022.
[10] M. Norgaard, O. E. Ravn, N. K. Poulsen, and L. K. Hansen, “Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner’s Handbook,” Berlin, Heidelberg: Springer-Verlag, 1st ed., 2000.
[11] J. D. Bomberger and D. E. Seborg, “Determination of model order for NARX models directly from input-output data,” Journal of Process Control, Vol. 8, no. 5, pp. 459–468, 1998.

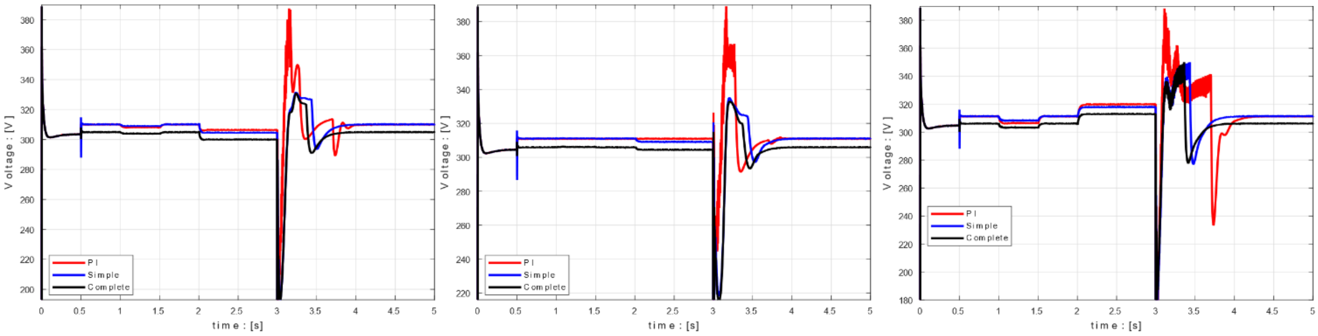

Fig.6. Simulation results for voltage in all DGs of the microgrid. The results show the response to the test scenarios for the PI control (red), the simple MRAC (blue), and the complete MRAC (black).

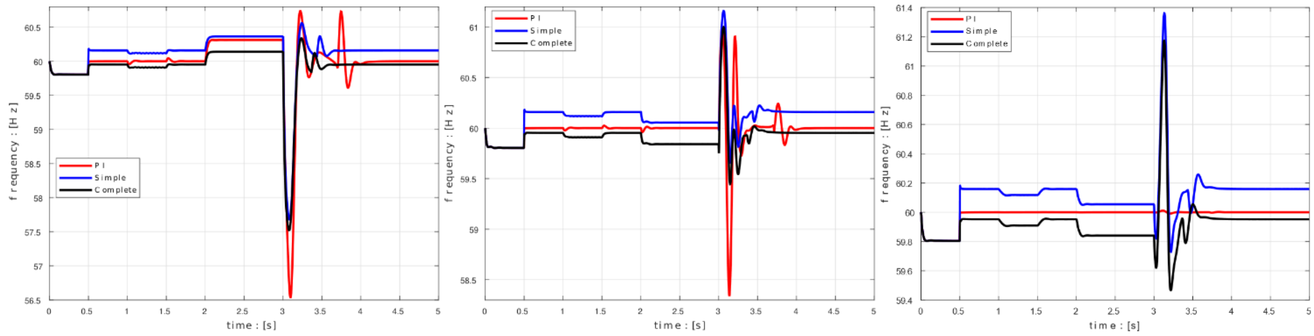

Fig.7. Simulation results for voltage in all DGs of the microgrid. The results show the response to the test scenarios for the PI control (red), the simple MRAC (blue), and the complete MRAC (black).