

Integrating Bayesian inference and neural ODEs for microgrids dynamics parameters estimation



Fathi Farah Fadoul ^{a,b,*}, Ramazan Çağlar ^a

^a Department of Electrical Engineering, Istanbul Technical University, Istanbul, Turkey

^b The Africa Center of Excellence for Logistics and Transport(CEALT), University of Djibouti, Djibouti

ARTICLE INFO

Keywords:

Bayesian inference
Neural ODEs
Uncertainty Analysis
Microgrid
Probability Posterior Distribution

ABSTRACT

The integration of solar and wind energy sources in microgrids has witnessed significant growth, giving rise to distinct challenges due to their intermittent nature when it comes to achieving efficient microgrid control. However, estimating the parameters of the dynamic microgrid components facilitates capturing the complex and time-varying characteristics of renewable energy generation. This requires an accurate estimation of the parameters from the dynamic differential equations for effective modeling and control. In this research paper, we presented a novel methodology based on the integration of Bayesian inference and Neural ODEs. The Bayesian inference quantifies the uncertainty, and the Neural ODEs model the dynamic systems. By combining the strengths of both methods, we aimed to achieve a precise and robust parameter estimation of the dynamic microgrid components. The methodology is validated on a simulated microgrid that consists of a diesel generator, Solar PV array, double-fed induction generator, and a battery energy storage system. The results showed promised inferences estimation obtained from the parameter posterior distribution even in the presence of uncertainty. This can enhance our understanding of the dynamics of renewable energy systems and can contribute to the advancement of decision-making microgrid control strategies.

1. Introduction

1.1. Background and motivation

According to the IEA [1], there are approximately 775 million people globally without electricity by 2022. This is due to a lack of energy infrastructure, which is expensive to build and maintain [2–4]. Furthermore, reliable, and economical fuel sources for power generation are scarce in some areas[5,6]. Thus, in the past decade, investments in renewable energies have continued to increase[7,8]. Among other technologies, Smart grids are now widely acknowledged as the ultimate solution for harnessing renewable energy sources(RES). Particularly, Microgrids are currently an answer for accessing electricity in isolated places. A total of 4500 microgrids are installed across the globe [9].

They often comprise loads, batteries, wind turbines, solar panels, and other DERs [10]. They can run separately from or in parallel with a larger power grid. In this developmental setting, microgrid control is vital to provide a continuous and sustainable power supply [11]. This command system balances electricity supply and demand within the microgrid and confirms the grid's stability and reliability by providing frequency regulation and voltage support. However, precise control is challenging due to the availability of renewable energy sources, dynamic weather conditions, and variable energy consumption. Therefore, estimating accurately the parameters of the state-of-charge of the battery, renewable energy accessibility, and load demand limits maximizes the application of microgrid control systems.

Abbreviations: IEA, International Energy Agency; DERs, distributed energy resources; Neural ODEs, Neural Ordinary Differential Equations; SVR, support vector regression; PCC, point common coupling; NUTS, No-U-Turn Sampler; SGHMC, Stochastic Gradient Hamiltonian Monte Carlo; SGLD, Stochastic Langevin Gradient Descent; SDEs, Stochastic Differentials Equations; DFIG, Double-Fed Induction Generator; BESS, Battery Energy Storage System; SRF, Synchronous Reference Frame; MCSE, Markov Chain Monte Carlo Standard Error; Rhat, Gelman-Rubin diagnostic statistic; Naïve SE, Naïve Standard Error; ESS, Effective Sample Size; Pdf, probability density function; MSE, Mean Squared Error; NLL, Negative Log-Likelihood; DIC, Deviance Information Criterion; AIC, Akaike Information Criterion; BIC, Bayesian Information Criterion.

* Corresponding author at: Department of Electrical Engineering, Istanbul Technical University, Istanbul, Turkey.

E-mail addresses: fadoul20@itu.edu.tr (F.F. Fadoul), caglarr@itu.edu.tr (R. Çağlar).

<https://doi.org/10.1016/j.segan.2024.101498>

Received 8 August 2023; Received in revised form 26 June 2024; Accepted 11 August 2024

Available online 15 August 2024

2352-4677/© 2024 Elsevier Ltd. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

| Nomenclatures | |
|------------------------------------|---|
| Diesel Generator Parameters | |
| δ | Rotor angle |
| w | Angular speed |
| P_m | Mechanical power |
| P_e | Electrical power |
| E_{fd} | Field excitation voltage |
| V_t | Terminal voltage |
| H | Inertia constant (in seconds) |
| D | Damping coefficient (in p.u) |
| x_d | d-axis synchronous reactance (in p.u) |
| T_a | Automatic voltage regulator time constant (in seconds) |
| K_a | Automatic voltage regulator gain (in p.u) |
| V_{ref} | Reference voltage (in p.u) |
| V_c | Converter voltage (in p.u) |
| T'_{d0} | Open circuit time constant (in seconds) |
| w_s | Synchronous speed (in radians per second) |
| T'_{q0} | Short circuit time constant (in seconds) |
| I_d | d-axis current (in p.u) |
| I_q | q-axis current (in p.u) |
| DFIG Parameters | |
| E'_{dd} | Transient or sub-transient voltage component in the direct axis (d-axis) of a synchronous machine |
| E'_{qq} | Transient or sub-transient voltage component in the quadrature axis (q-axis) of a synchronous machine |
| i_{qs} | The stator current component in the quadrature axis (q-axis) |
| i_{ds} | The stator current component in the direct axis (d-axis) |
| w_s | Synchronous speed (in radians per second) |
| x_{dd} | d-axis transient reactance (in p.u) |
| x_q | q-axis synchronous reactance (in p.u) |
| x_{qq} | q-axis transient reactance (in p.u) |
| s | Slip (in p.u) |
| L | Magnetizing inductance (in p.u) |
| R_r | Rotor resistance (in p.u) |
| L_r | Rotor inductance (in p.u) |
| L_m | Mutual inductance (in p.u) |
| v_{dr} | d-axis rotor voltage (in p.u) |
| v_{qr} | q-axis rotor voltage (in volts) |
| $\psi_{qr} :$ | Rotor flux linkage in the quadrature axis |
| ψ_{dr} | Rotor flux linkage in the direct axis |
| L' | Equivalent inductance (in p.u) |
| T'_0 | Time constant (in seconds) |
| x_{dd} | d-axis transient reactance (in p.u) |
| PV Array Parameters | |
| i_{pym} | PVM current (in amperes) |
| V_{pvm} | Voltage across the PV module |
| i_L | Inductor current (in amperes) |
| C_{pvm} | DC-Link capacitor (in Farads) |
| r_{pv} | PVM resistance (in Ohms) |
| V_{dc} | DC-Link voltage (in voltage) |
| C_{dc} | DC-Link capacitance (in Farads) |
| S_{pv} | PVM snubber network time constant (in p.u) |
| L_b | Inverter inductor (in mHenry) |
| i_{01} | DC-Link current (in amperes) |
| BESS Parameters: | |
| C_{p0} | Battery capacitance (in Farads) |
| R_{p0} | Battery resistance (in Ohms) |
| R | Load resistance (in Ohms) |
| R_{binit} | Battery initial resistance (in Ohms) |
| E_b | Battery voltage (in Volts) |
| L_C | Battery inductance (in mHenry) |
| C_0 | Output filter capacitance (in Farads) |
| S_{bat} | Battery state of charge |
| i_{02} | Output current (in amperes) |
| i_{bat} | Battery current - This is the current flowing into or out of the battery (in amperes) |
| V_0 | Output voltage of the battery system (in Volts) |
| V_1 | Intermediate voltage in the battery system (in Volts) |

1.2. Literature review

Previous studies in microgrid modeling and simulation have reported that data-driven methods, model-based approaches, and state estimation have been put forth as parameter estimation methods. Data-driven approaches build predictive models that can estimate microgrid parameters in real time through the historical data obtained from the microgrid [12–16]. Bayesian methods particularly increase the parameter estimation accuracy and reduce the related uncertainty [17]. In [18], Bayesian inference quantifies the uncertainty of a synchronous thermal model parameter estimation by providing the parameter probability posterior distribution. Machine learning techniques like SVR are used to minimize the error between the observed and simulated values of the microgrid system [19]. This research [20] suggests a long short-term memory recurrent neural network approach to determine the microgrid model parameters. On the other hand, model-based methods estimate power flows, voltage, and frequency variation parameters with the help of mathematical models [21,22]. For instance, the optimization-based method in [23] employed particle swarm optimization to predict system parameters and raise the microgrid's performance. The choice of the control parameters and the load frequency parameters have an impact on the microgrid dynamic response. [24] investigates the PCC substation measurements to detail the microgrid dynamic model. Stability control methods require knowledge of the system parameters. [25] proposed an experimental validation of a

nonlinear gray box model that reproduced the behaviors of the dynamics microgrid response. With these analogous models, the complex structure of the microgrid is relieved but there is a need to improve the non-linear functions within the microgrid. Kalman filters are effective tools for state estimation in both linear and non-linear systems. They can successfully manage noisy data and are computationally efficient. Nevertheless, they require a thorough understanding of the system model, and when there are model uncertainties, their performance may suffer [26]. Recursive least squares are a non-parametric technique that works with adaptive systems because it may estimate system parameters without having a precise model [27]. Nevertheless, it could require large computation and struggle with numerical stability.

Despite the variety of the proposed methods in the literature, Bayesian inference and Neural ODEs have been preferred in this paper for their robustness in managing uncertainties and their ability to model dynamic systems with high accuracy for determining microgrid system parameters. To this point, current research in Bayesian inference for microgrids has just focused on the integration of real-time data for updating the probability assessment of systems parameters. In microgrid operations, this integration allows adaption with volatile renewable energy sources. A growing tendency is the application of Bayesian networks and probabilistic graphical models in microgrids which can control the uncertainty in distributed energy resources and their interactions. However, accurately modeling the key parameters of the microgrid remains a critical question that must be addressed. Feasibly,

Table 1

Comparative Analysis of Methodologies in Microgrid Parameter Estimation and Simulation.

| Reference | Methodology | Model Type | Key Findings | Advantages | Limitations |
|-----------|---------------------------------|--------------------|---|---|--|
| [12–15] | Data-driven Predictive Models | Various | Real-time parameter estimation through historical data analysis. | Utilizes large historical data sets, suitable for real-time applications. | Relies heavily on data quality and quantity. |
| [16] | Bayesian Inference | Statistical | Increased accuracy in parameter estimation by reducing uncertainty. | Quantifies uncertainty, integrates prior knowledge. | Computationally intensive. |
| [17] | Bayesian Inference | Statistical | Quantifies uncertainty in synchronous thermal model parameter estimation via posterior distributions. | Provides detailed probabilistic understandings into parameter estimates. | Requires detailed prior distributions. |
| [18] | Support Vector Regression (SVR) | Machine Learning | Minimized error between observed and simulated values. | Effective in handling nonlinear relationships. | May be sensitive to kernel choice and tuning parameters. |
| [19] | LSTM Recurrent Neural Network | Deep Learning | Determination of microgrid model parameters. | Can capture complex temporal patterns and dependencies. | Requires substantial training data. |
| [20,21] | Model-based Estimation | Mathematical Model | Estimation of power flows, voltage, and frequency variations. | High precision in parameter estimation. | Limited by model accuracy and complexity. |
| [22] | Particle Swarm Optimization | Optimization | Predicted system parameters to enhance microgrid performance. | Optimizes performance based on a wide range of parameters. | Requires careful tuning of swarm parameters. |
| [23] | Model-based Estimation | Mathematical Model | Detailed the microgrid dynamic model using PCC substation measurements. | Allows for precise control and tuning of microgrid operations. | Dependent on the accuracy of substation measurements. |
| [24] | Nonlinear Gray Box Model | Hybrid Model | Reproduced dynamic microgrid responses experimentally. | Balances model fidelity with computational efficiency. | Might not capture all nonlinear dynamics accurately. |
| [25] | Kalman Filters | State Estimation | Effective state estimation in noisy environments. | Computationally efficient and robust to noise. | Performance degrades with model inaccuracies. |
| [26] | Recursive Least Squares | Adaptive System | Estimation of system parameters in adaptive systems without precise models. | Adaptive to changing conditions, no need for a precise model. | Computational demands, issues with numerical stability. |

Neural ODEs can be a strong candidate in modeling microgrids due to their ability to capture the continuous dynamics of these systems more naturally than traditional discrete methods. Neural ODEs have been applied in studies similar to the temporal evolution of microgrid components under various operational conditions. These references stress the capability of the Neural ODE method to handle the continuous-time dynamics systems which are fundamental to a precise simulation of the microgrid system [28–30]. Briefly, Bayesian methods excel in modeling uncertainties and accordingly, Neural ODEs are effective in capturing dynamics. Given these distinct strengths, the integration of these two models appears to be a promising approach to enhance the analytical capabilities in various applications, particularly in environments where both uncertainty and dynamic behavior play critical roles. This interactive integration aims to adventure the uncertainty quantification of the Bayesian methods with the dynamic system analysis provided by the Neural ODEs. In the past few years, research on Bayesian inference and Neural ODEs has continued to advance by improving the model prediction's robustness and uncertainty quantification. Studies have effectively managed to integrate the Bayesian inference methodology with Neural ODEs functionality. NUTS sampler, SGHMC sampler, and SGLD sampler were established to estimate the posterior distributions of the model parameters [31,32]. Also, the integration of variational inference with the flow normalization of the Bayesian Neural ODE has shown fruitful results in handling dynamical systems, for instance, predator-prey models and epidemiological modeling [31]. Therefore, increasing the performance of the probability identification of model specifications. Furthermore, in complex systems where traditional models struggle, Bayesian Neural ODEs have been accurately applied such as predicting dynamics systems governed by SDEs [33,34]. Additionally, Gaussian models were used to infer the ODE system from the data which shows the performance of the Bayesian Neural ODE [35]. These studies emphasize the importance of Bayesian Neural ODEs in practical applications. Table 1 presents a comparison of the state-of-the-art methods previously referenced for parameter estimation.

1.3. Contribution and organization

In this paper, we estimated the parameters of a set of differential equations that describe the dynamics of a microgrid by combining Bayesian inference with the neural ODEs method. We aimed to address the challenges associated with intermittent renewable energy resources

that lead to uncertain operating conditions. In our research, we propose neural ODEs as a likelihood function in the Bayesian inference method to estimate the posterior parameter probability distributions governing the dynamics of a microgrid system. We captured the intricate dynamics by incorporating the flexibility of Neural ODEs models in the Bayesian probabilistic framework. This method is beneficial for microgrid components that behave non-linearly and time-varyingly. We sought to contribute to the microgrid parameter estimation field by investigating and increasing the accuracy of this combined method according to the related uncertainty and variations. The current state of research shows a gap in the integration of neural ODEs and Bayesian inference. Whereas both methods have been studied separately in various domains, their mixture for parameter estimation in microgrid studies is an unexplored area. The highlights of the contribution in the paper are summarized as follows:

- New and innovative methodology with the integration of neural ODEs and Bayesian Inference
- Flexibility of neural ODEs models capturing the interaction of non-linear behaviors
- Robustness and uncertainty measurement in parameter estimation for microgrids systems
- Improved enhancement in parameter estimation for a better understanding of the parameters

Traditional methods employed in microgrid dynamics are limited by their need to fix parameters, and by their lack of flexibility to adapt to real data and estimate uncertainties in large and complex systems like microgrids which present volatile resources such as wind and solar. With this need in mind, Neural ODE appeared to be an ideal candidate for fine-tuning the parameterization with its data-driven learning. This model performs well because it can implement the dynamics of differential equations, unlike other methods. This study used Julia programming, specifically the Turing.jl package, to integrate Bayesian inference potentials to estimate each parameter. However, these applications have their limitations, as they require an enormous amount of processing time and are difficult to interpret.

The introduction depicts the purposes of the study, highlights the relevance, provides an overview of previous works, and presents the contributions of the paper. In the methodology, the modeling of the dynamic differential equations of microgrid components is presented and the proposed Bayesian inference and Neural ODEs are described.

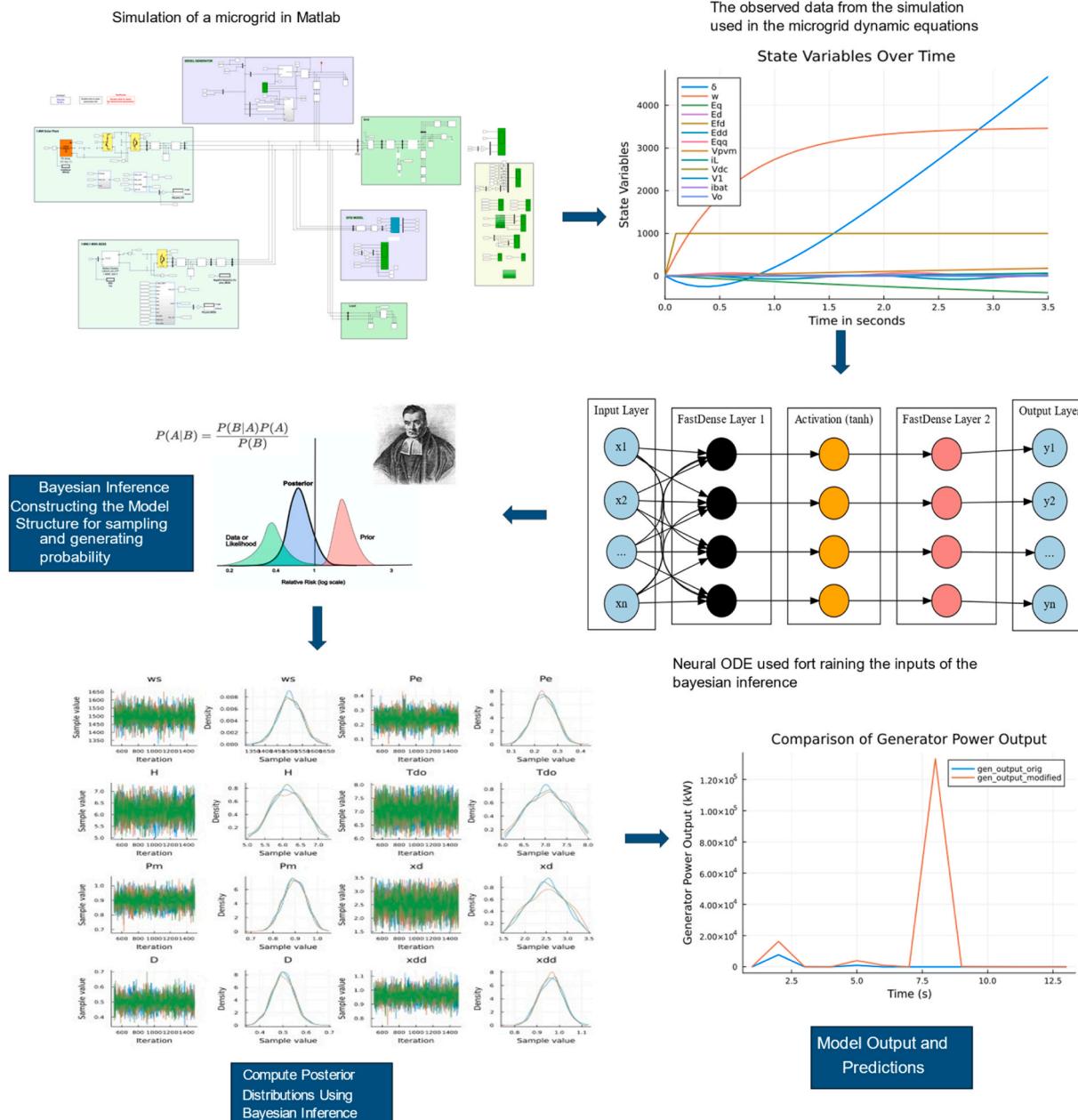


Fig. 1. Flow of the research and Overview of the methodology.

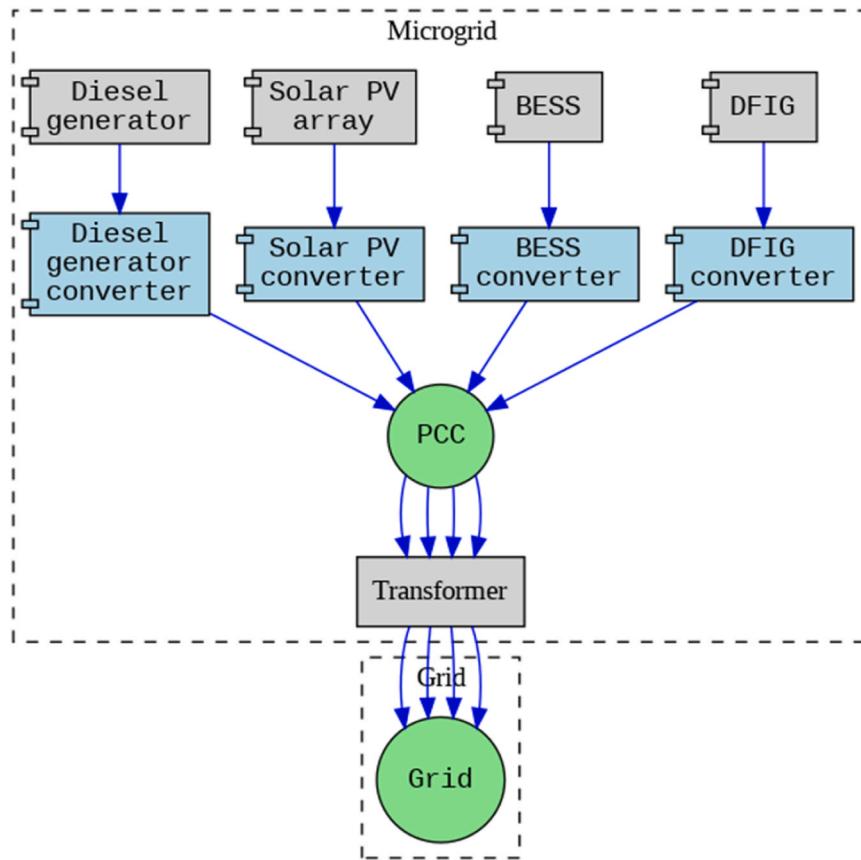


Fig. 2. Simplified Microgrid Architecture.

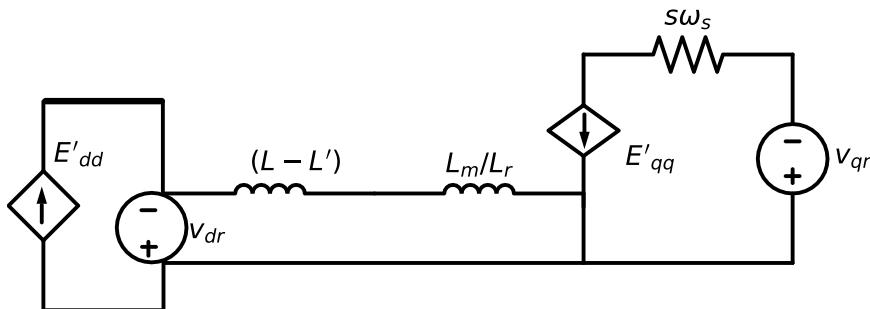


Fig. 3. Equivalent circuit of the DFIG.

The findings and analysis of our research are presented in the results. And in the discussion, we interpreted parameter estimation results analysis, significance model validation parameters, model comparison for model fitting, and sensitivity analysis. Lastly, in conclusion, we showed that our research has achieved its objectives.

2. Methodology

2.1. Overview of the methodology

The methodology illustrated combines Bayesian inference with neural networks to improve the prediction of dynamic system parameters. First, model outputs are analyzed to estimate parameter uncertainty. A neural network model is then used to generate probability distributions for these parameters. The posterior distributions are calculated using Bayesian inference, presenting an accurate estimate of the parameters. Fig. 1 shows the flow of this study research.

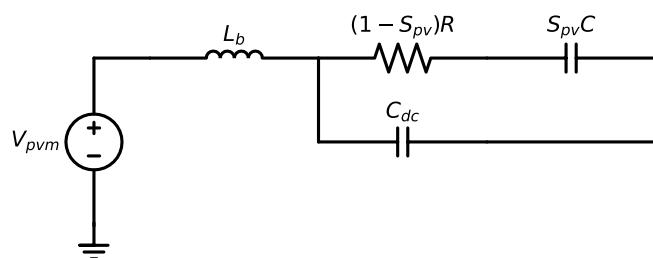


Fig. 4. Equivalent circuit of the solar PV array.

2.2. Dynamic Modeling of Microgrid Components

The study's primary goal is to create and evaluate a novel neural network Bayesian inference model for determining microgrid dynamics parameters. Fig. 2 shows a simplified microgrid architecture that

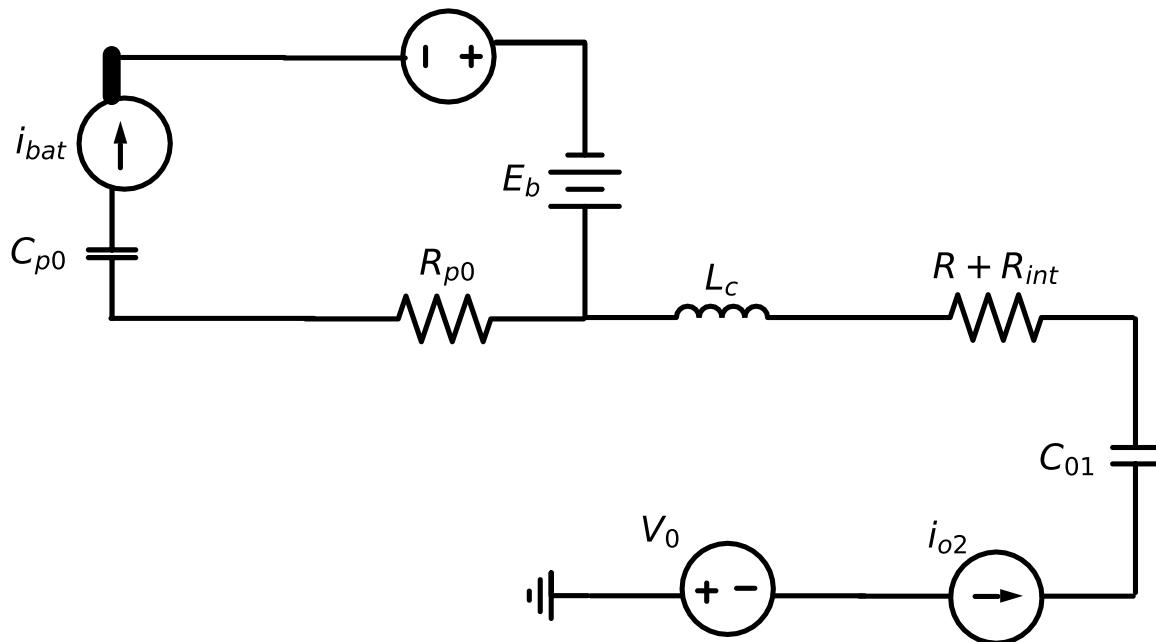


Fig. 5. Equivalent circuit of the BESS component.

consists of a diesel generator, a solar PV array, BESS, and DFIG components. The dynamics of the microgrid components are described by a set of differential equations. These equations capture the interactions of the dynamic behaviors of these components, permitting us to understand their performance in the microgrid system.

2.2.1. Generator Model

The diesel generator contains the sixth-order synchronous model. It considers the dynamics of the damper, field, and stator. The mechanical part, electrical part, and the dynamics field exciter correspond to the simplified synchronous machine as follows [36]:

$$\frac{d\delta}{dt} = w - w_s \quad (1)$$

$$\frac{dw}{dt} = \frac{w_s}{2H}(P_m - \frac{D}{w_s}(w - w_s) - P_e) \quad (2)$$

$$\frac{dE_q'}{dt} = -\frac{1}{T_{d0}}(E_q' - (x_d - x_d')I_d + E_{fd}) \quad (3)$$

$$\frac{dE_d'}{dt} = -\frac{1}{T_{q0}}(E_d' - (x_q - x_q')I_q) \quad (4)$$

$$\frac{dE_{fd}'}{dt} = -\frac{1}{T_A}(E_{fd} - K_A(V_{ref} + V_c - V_t)) \quad (5)$$

2.2.2. DFIG dynamic model

Based on the SRF theory to transform the dynamic variables from the stationary reference to a rotating reference frame for the DFIG dynamics system and with parks transformation neglecting stator voltage, the DFIG rotor voltage dynamics formulas are as follows [37] and the equivalent circuit of the DFIG [38] is shown in Fig. 3:

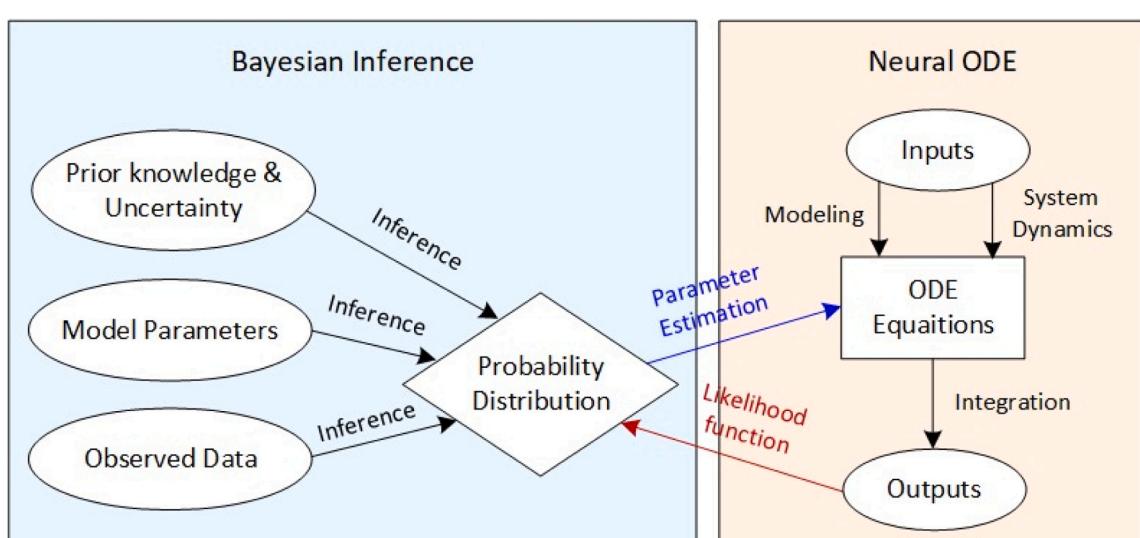


Fig. 6. Methodology of Probability Distribution Generation by the Bayesian Neural ODEs.

Table 2
Initial operating parameters values of the microgrid system.

| Parameter | Value | Parameter | Value |
|-----------|-------------|-----------|---------------|
| Pm | 0.8976 p.u. | ws | 1500 rad/s |
| Pe | 0.2385 p.u. | xdd | 0.96 p.u. |
| Vt | 1.0058 p.u. | xq | 0.794 p.u. |
| H | 6.15 s | xqq | 0.5 p.u. |
| D | 0.5 p.u. | s | 0.50 p.u. |
| xd | 2.5216 p.u. | L | 0.0477 p.u. |
| Ta | 0.01485 s | Rr | 0.006263 p.u. |
| Ka | 200 p.u. | Lr | 0.08187 p.u. |
| Vref | 1.0 p.u. | Lm | 2.92 p.u. |
| Vc | 5 p.u. | vdr | 0.4027 p.u. |
| Tdo' | 7.022 s | vqr | 0.4027 V |
| ws | 1500 rad/s | ipvm | 539.1443 A |
| Tqo' | 2.135 s | iL | 5 A |
| Id | 0.1154 p.u. | Cpvm | 10 F |
| Iq | 3.1689 p.u. | rpv | 0.005 Ω |
| iqs | 0.1973 p.u. | Cdc | 1 F |
| ids | 1.87 p.u. | Spv | 0.50 p.u. |
| Lb | 5 mH | R | 0.002 Ω |
| io1 | 5 A | Rbinit | 0.028 Ω |
| Cpo | 1 F | Eb | 48 V |
| Rpo | 0.75 Ω | Lc | 10 mH |
| Co | 3000 F | Co | 3000 F |
| Sbat | 0.40 | io2 | 8 A |

$$\frac{dE'_{dd}}{dt} = -\frac{1}{T_0'} [E'_{dd} - (L - L')i_{qs}] + sw_s E'_{qq} - (L_m/L_r)v_{dr} \quad (6)$$

$$\frac{dE'_{qq}}{dt} = -\frac{1}{T_0'} [E'_{qq} - (L - L')i_{ds}] - sw_s E'_{dd} - (L_m/L_r)v_{qr} \quad (7)$$

2.2.3. Solar PV array dynamic model

Kirchhoff's law allows the description of the PV cell's U-I equation and by the principle of solar cell operation, the set of differential equations that describe the dynamics of a PV system are as follows [36] and the equivalent of the solar PV array is shown in Fig. 4:

$$C_{pvm} \cdot \frac{dV_{pvm}}{dt} = i_{pvm} - i_L \quad (8)$$

$$L_b \cdot \frac{di_L}{dt} = [V_{pvm} - r_{pv}I_L - (1 - S_{pv})V_{dc}] \quad (9)$$

$$C_{dc} \cdot \frac{dV_{dc}}{dt} = (1 - S_{pv})i_L - \frac{1}{C_{dc}}i_{01} \quad (10)$$

2.2.4. BESS dynamic model

Microgrid-distributed energy resources are intermittent and uncertain. Hence, BESS ensures stability and restores generation immediately after interruption. The BESS is connected to the AC bus grid through a DC/DC/AC converter. The dynamics equations of the BESS are as fol-

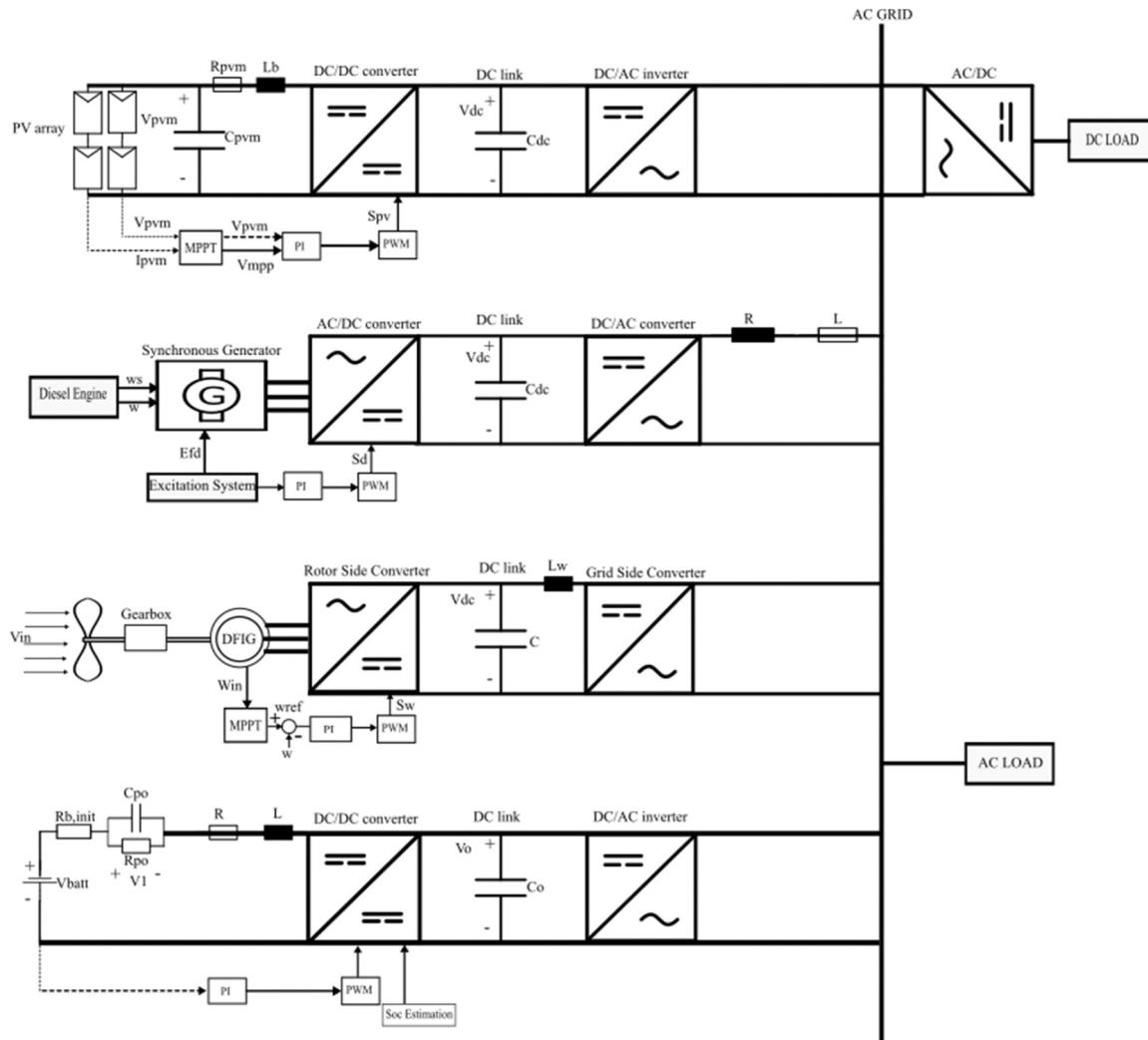


Fig. 7. The Experimented Microgrid.

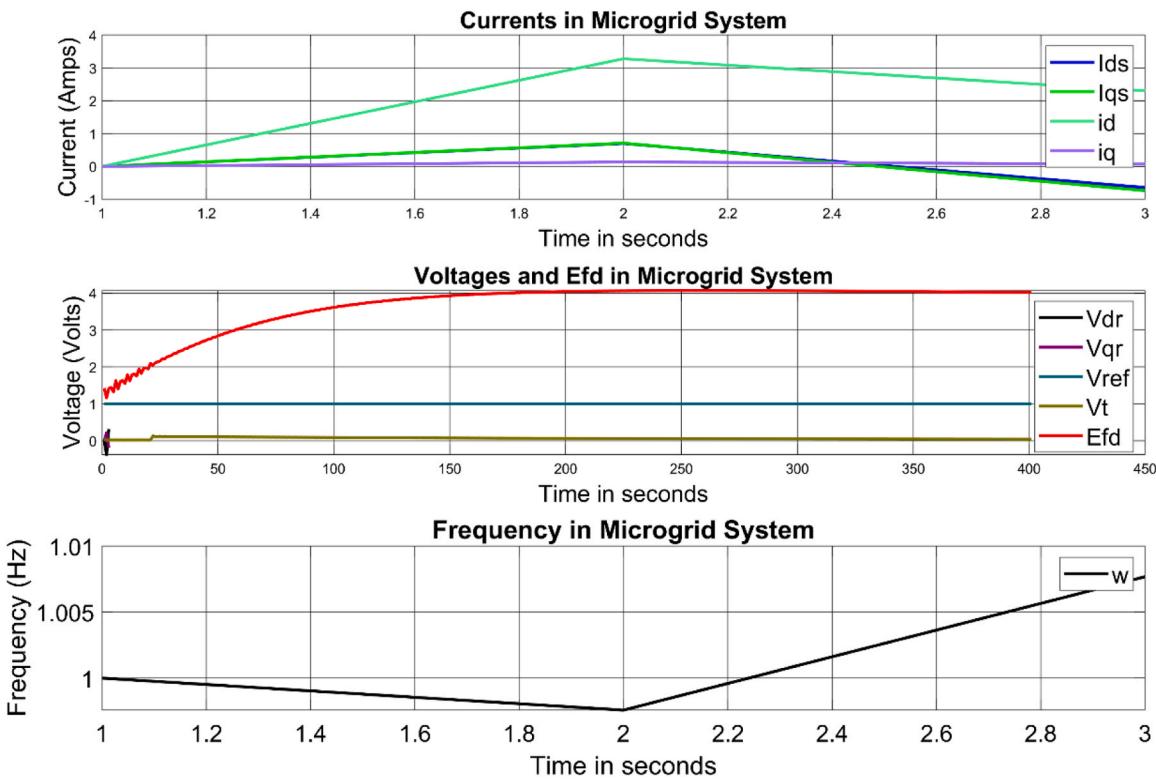


Fig. 8. Initial variables and parameters data problem.

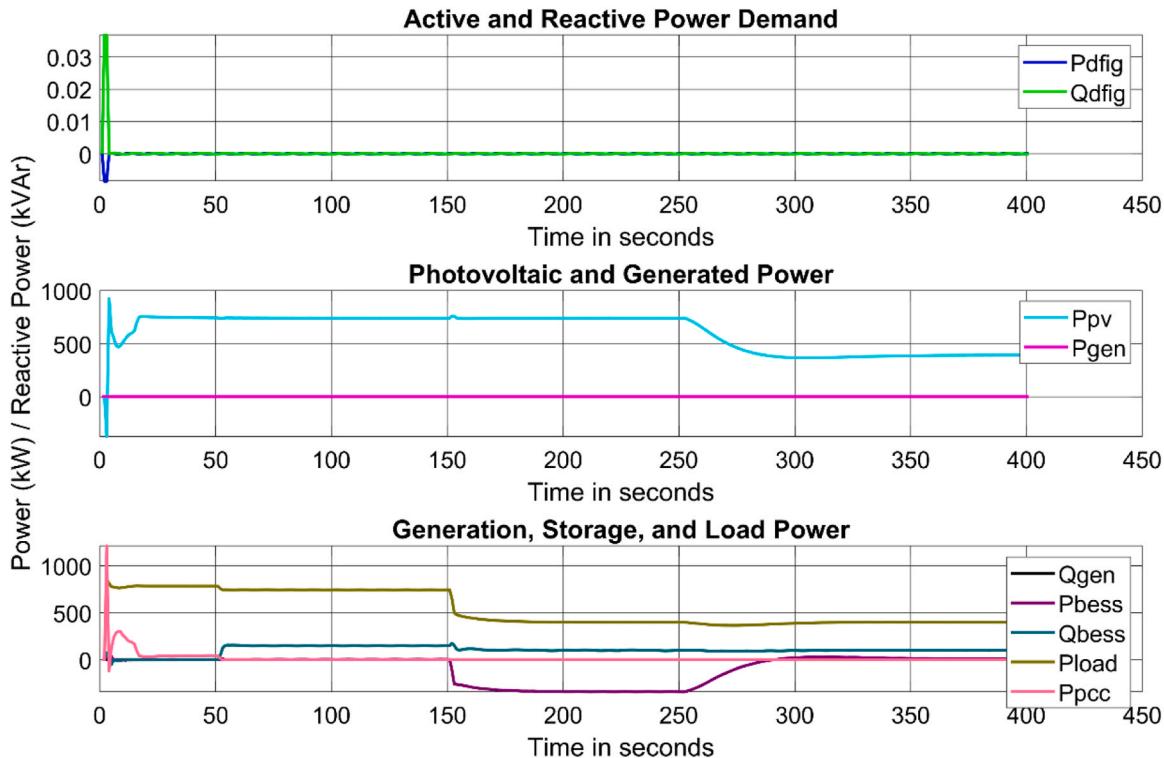


Fig. 9. Generation Power measurements.

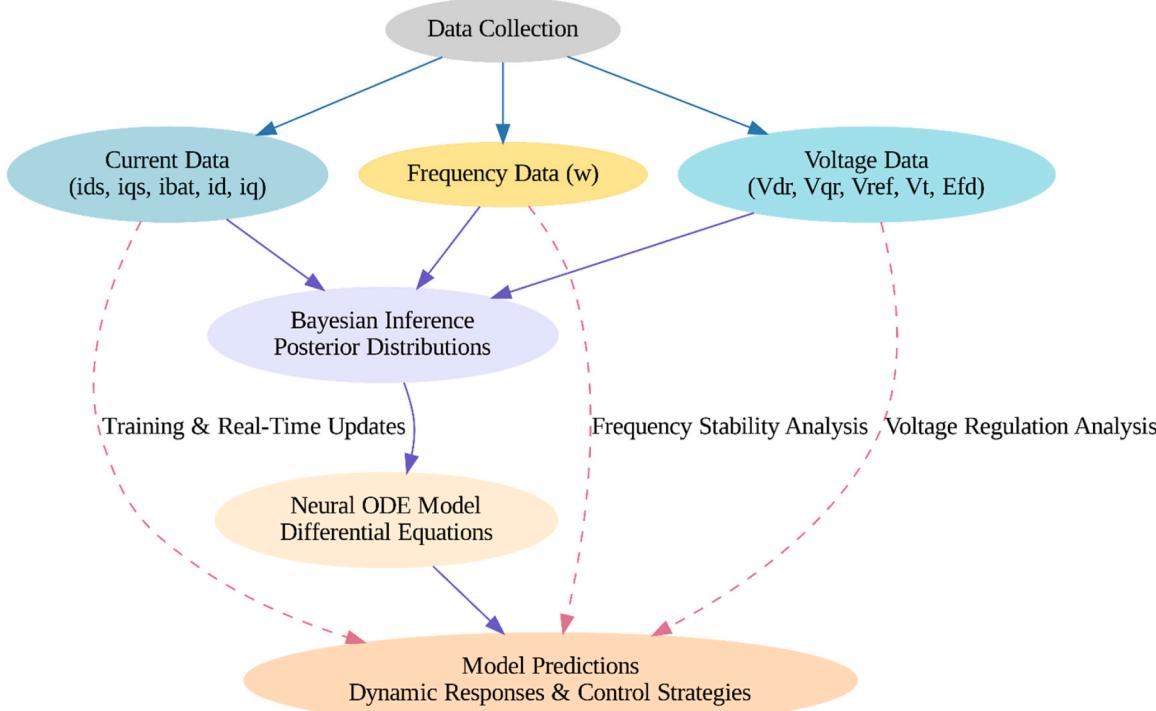


Fig. 10. Integration of data through the methodology.

lows [36] and the equivalent circuit of the BESS is shown in Fig. 5:

$$\frac{dV_1}{dt} = \frac{1}{C_{p0}} \left(i_{bat} - \frac{V_1}{R_{p0}} \right) \quad (11)$$

$$\frac{di_{bat}}{dt} = \frac{1}{L_c} \left[-(R + R_{b,init})i_{bat} + (1 - S_{bat})V_0 - V_1 + E_b \right] \quad (12)$$

$$\frac{dV_0}{dt} = \frac{1}{C_0} [(1 - S_{bat})i_{bat} - i_{o2}] \quad (13)$$

2.2.5. Variable Definitions and Their Roles in Microgrid Dynamics Modeling

In the development of this microgrid dynamic model simulation, variables are described in 13 differential equations from (1)-(13) for capturing and estimating critical parameters of the different microgrid components. Diesel generator parameters such as δ and w , and power metrics like P_m and P_e , important parameters in the generator dynamics are among the described parameters in these equations. Each component has specific variables that direct its operational characteristics. As an illustration, E_{fd} and V_t which are the generator model variables point

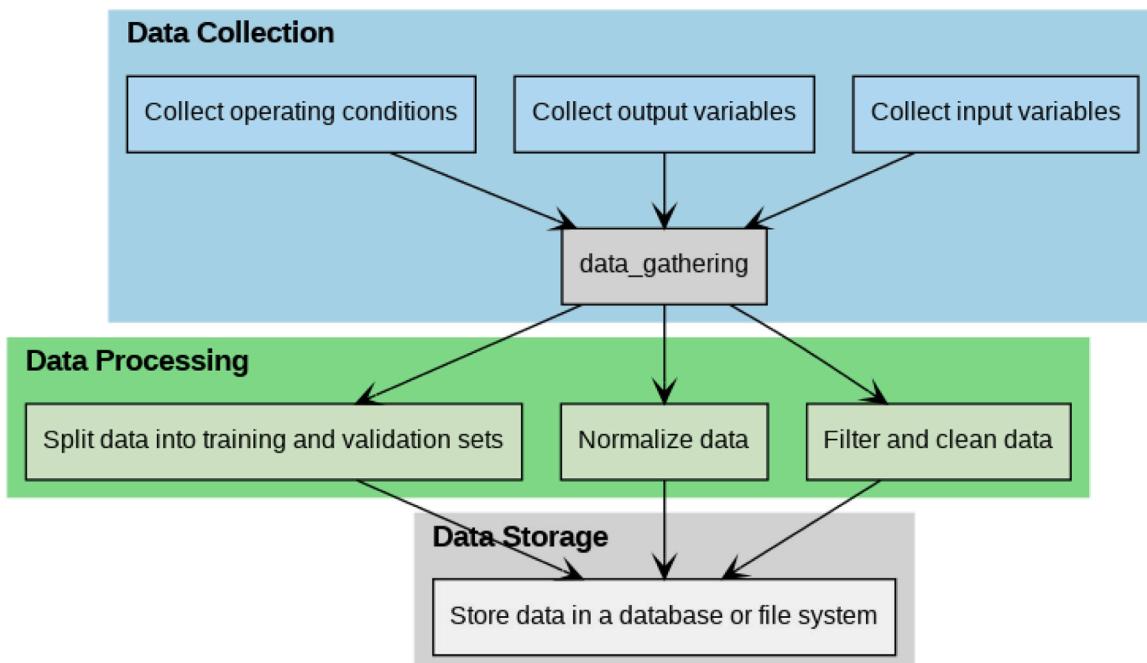


Fig. 11. Data Preparation.

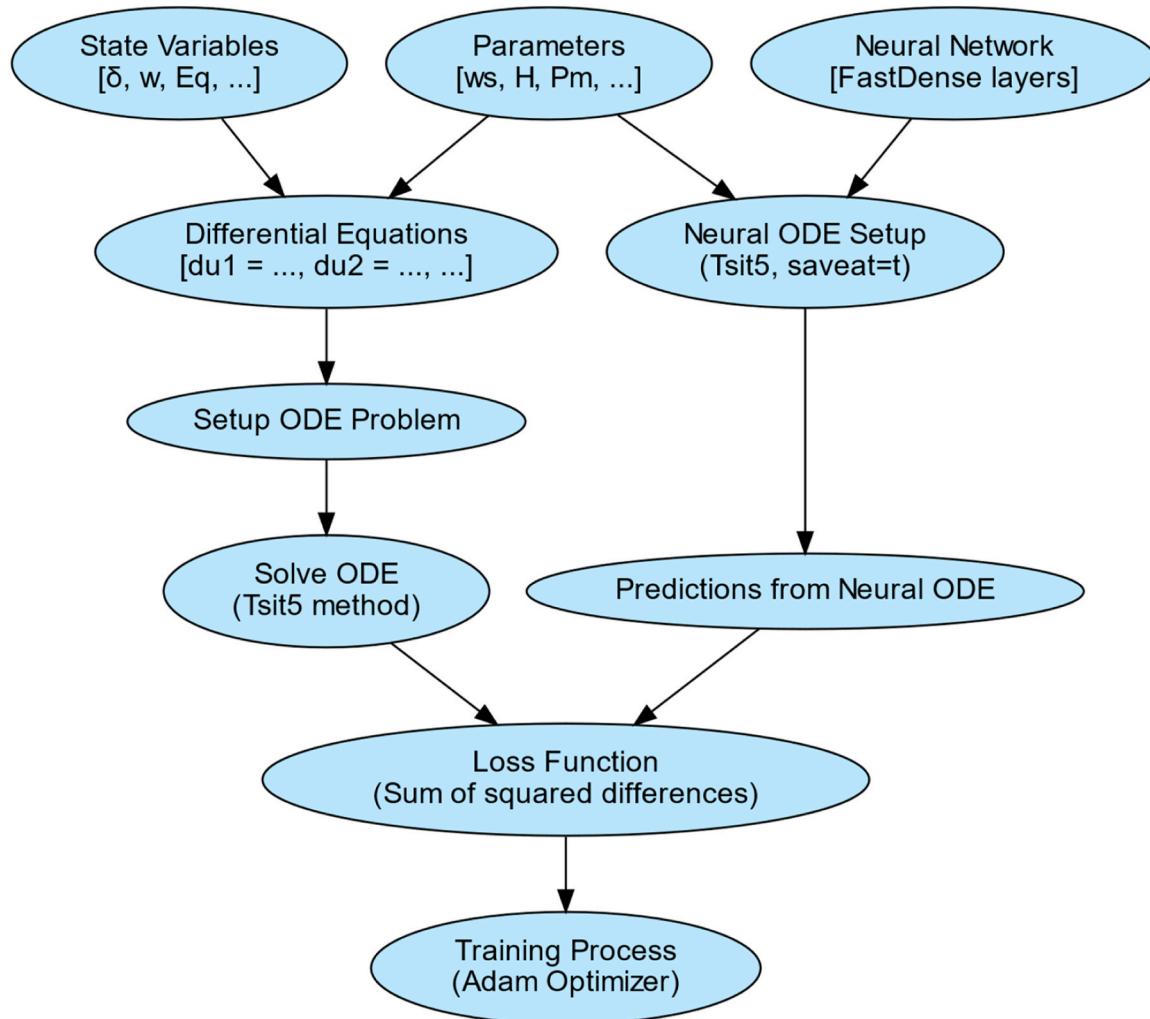


Fig. 12. Neural ODEs and Bayesian Inference Integration Architecture.

to the simulation of the generator response under varying electrical loads. In this way, the DFIG model employs rotor voltages E'_{dd} , E'_{qq} and stator currents i_{qs} , i_{ds} which assess the interfacing with grid dynamics. The energy conversion efficiency of the solar PV array is helped by the voltage across the PV module V_{pvm} , V_{dc} and load current i_L variables. Similarly, energy storage and stability analysis of the BESS dynamics are described through variables like battery voltage V_o and V_1 and battery current i_{bat} . These equations are utilized to simulate the dynamic stability behavior of the microgrid.

2.3. Model Development: Integration of Bayesian Inference and Neural ODEs

2.3.1. The applied tools

The methodology in this study is based on the integration of Bayesian inference and Neural ODEs methods. We employed the Turing.jl [39] package in Julia that offered the tools to define and sample the posterior distributions Bayesian inference. Along the Flux.jl [40], the dynamic behavior of the microgrid components is modeled. The Flux.jl, which is

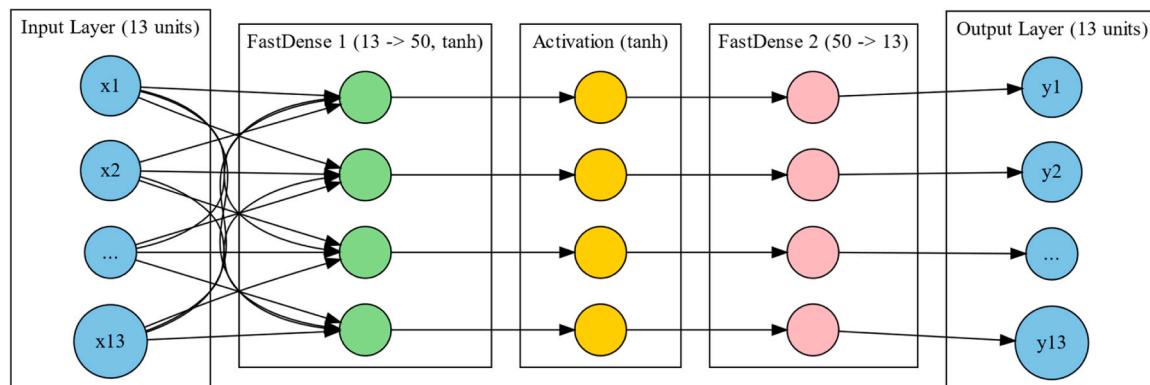


Fig. 13. Neural ODEs Architecture.

also a Julia package for neural network modeling contains wide architectures and optimization algorithms for designing and training complex models. The temporal aspect of the microgrid dynamics is added with the help of the DiffeqFlux.jl [41]. This library allows to model the time-varying behavior of the microgrid and is a package fully implemented in Julia programming.

2.3.2. Model formulation

As displayed in Fig. 6, the Bayesian inference generates a probability distribution by integrating prior knowledge and observed data where the likelihood is derived from the Neural ODEs network. Prior distributions were specified to describe the values of the control parameters, representing our initial knowledge and uncertainty about these values by providing a picture of our beliefs before considering the observed data. Table 2 corresponds to our knowledge and uncertainty about the parameter values serving as a reference for comparison as this table gives the operating initial values known as prior knowledge.

2.3.3. Data processing

The observed data were obtained from a simulation in MATLAB/Simulink of an experimented microgrid presented in Fig. 7 that consists of DFIG, Solar PV array, BESS, DG, and loads connected to the North American grid. The simulated observed data are shown in Fig. 8 and Fig. 9. This data collection consists of gathering and coordinating the data. The time series data shown in Figure are used for training the Bayesian Neural ODE model. I_{ds} and I_{qs} represent the stator current data which gives knowledge of the electromagnetic torque production in synchronous and induction machines. The observed voltage data (V_{dr} , V_{qr} , V_{ref} , V_t , E_{fd}) assess the regulation of voltage in the microgrid. Precisely, V_{ref} and V_t ensure the output voltage in the desired operating limits, and the voltage E_{fd} is used for controlling the output of the synchronous generator. Furthermore, the frequency of w is employed when analyzing the process of synchronization or islanding with the main grid. Deviations of normal frequency can be a signal of a possible outage or failure in the system. Fig. 10 shows how these data are integrated into Bayesian Inference and the Neural ODE model. Fig. 11 illustrates the importance of data in our research since this method requires a substantial amount of information. It highlights the data collection, data processing, and data storage enabling the collection of relevant input and output variables and operating conditions. In the data collection process, the input variables consist of thirteen measured variables, while the output variables are the same but with added noise to simulate real measurement scenarios. The operating conditions refer to the initial values set for each parameter. During data processing, the data is split into training and validation sets, normalized, and cleaned. Finally, the processed data is stored in a database or file system for further analysis and model building.

2.3.4. Neural ODEs approach

The Neural ODEs approach is a neural network that learns the continuous time differential equation evolution of the system that depicts the rate of change of the state variables over time. The diagram in Fig. 12 illustrates the Neural ODE modeling process. State variables and parameters are used to formulate differential equations. These equations are then configured into an ODE problem, and solved using the Tsit5 method, alongside the initialization of a neural network to configure the neural ODE. The neural ODE predictions are compared with the ODE solutions to calculate the loss function, which is minimized in the training process using the Adam optimizer. Fig. 13 visualizes the representation of the structure and the connection of the Bayesian Neural ODEs model between the input layer, hidden layers, and output layer. The state variables are also shown in Figs. 9 and 10 which give a general understanding of how the states interact with each other during the simulation. Neural ODEs model the evolution of system states continuously using differential equations parameterized by a neural network. Inputs include initial states, time points, and neural network parameters. The neural network defines the ODEs, which are solved to predict system states over time. Outputs are

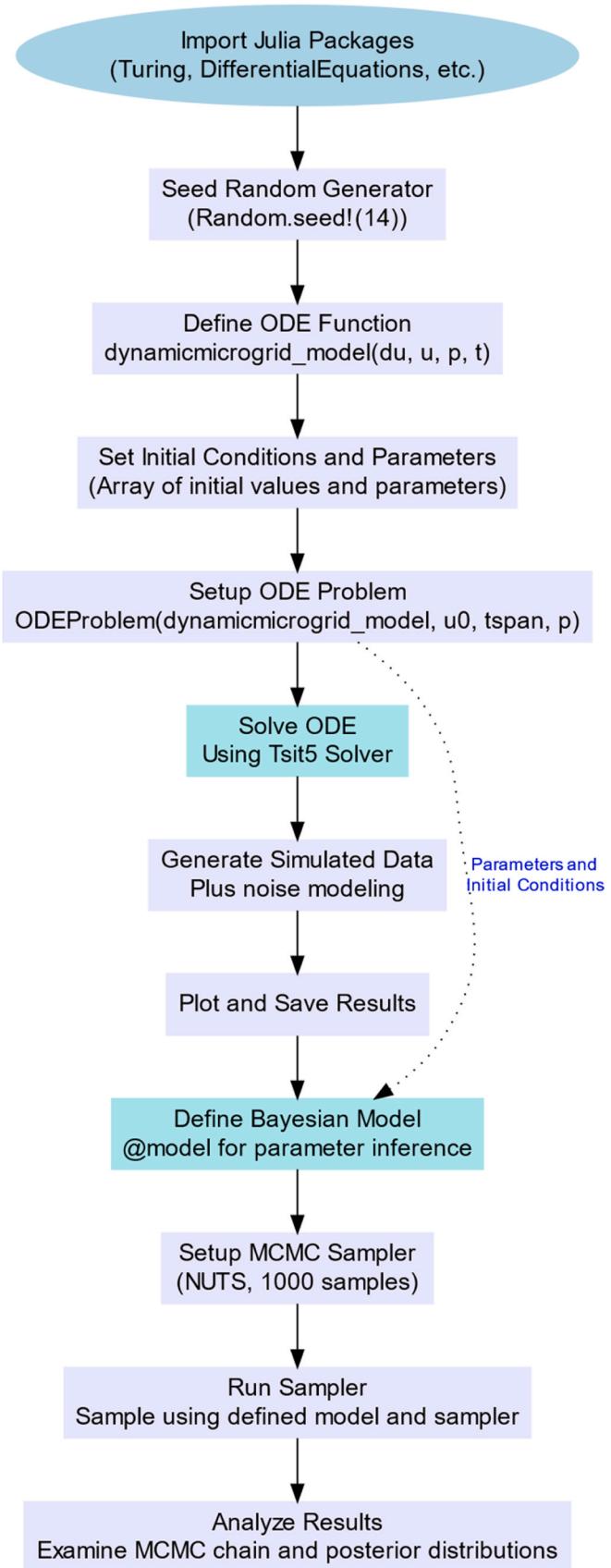


Fig. 14. Proposed Bayesian Neural ODEs method.

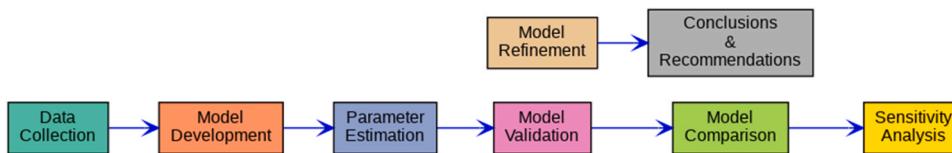


Fig. 15. The process of the research methodology.

the predicted states and a loss function quantifying prediction error. The training process optimizes neural network parameters to minimize this loss, enhancing prediction accuracy.

2.3.5. Bayesian Inference and Neural ODE Formulation

Given the microgrid system described by the 13 equations from Eq.1 to Eq.13. The objective is to estimate the posterior distributions of control parameters for microgrid components by integrating these two methods. The mathematical model that describes the rate of change of a variable x with respect to time:

$$\frac{dx}{dt} = f(x, u, \theta) \quad (14)$$

where x represent the system state variables, u is used as control inputs and θ represent the model parameters. So therefore, the Bayesian inference approach of the parameters posterior distribution given to the observed data and applying the Bayes' theorem can be stated as:

$$p(\theta|D) = \frac{p(D|\theta)p(\theta)}{p(D)} \quad (15)$$

Where $p(\theta)$ is the prior distribution over the parameters, $p(D|\theta)$ is the likelihood of the data given the model parameters, and $p(D)$ is the marginal likelihood. Bayes theorem states that the posterior distribution is proportional to the product of the prior distribution and the likelihood of the data. Then the Neural ODE model approach with the parameters values over time are presented by this expression as follows:

$$\frac{dx(t)}{dt} = f(x(t), \theta(t)) \quad (16)$$

Fig. 14 shows the framework of microgrid dynamic Bayesian Neural ODEs parameter estimation that consists of data collection task, inference modeling and results providing a clear process to parameter uncertainty estimation. This figure shows a workflow for modeling and inferring parameters of a dynamic microgrid using Julia. The steps start by importing the necessary packages, such as Turing.jl and DifferentialEquations.jl. Next, a random generator is initialized, and the ODE function for the dynamic microgrid model is defined. Initial conditions and parameters are set, and an ODE problem is configured with these parameters. The model is solved using the Tsit5 solver, and then simulated data with noise is generated, plotted and saved. Next, a Bayesian model is defined for parameter inference, followed by the configuration and execution of the MCMC sampler (NUTS). Finally, the results are analyzed by examining the MCMC chain and posterior distributions.

The datasets described in Figure are used to train the Bayesian inference to derive posterior distributions of the microgrid control parameters. The observed data are integrated into the Neural ODE framework and each data describes each variable. Accordingly, the model learns from the inherent dynamism of the microgrid components since each variable is representative of the microgrid components. Bayesian Inference allows to update of the estimation of the parameter from the differential equations governing the microgrid dynamics based on this data and further even if new data is implemented. Thereby, the quantification of uncertainties is possible with the observed data. The probabilistic approach guide to understanding the variability and sensitivity of the microgrid control parameters which is given by the integration of the Bayesian Neural ODE. Furthermore, this method adapts to different operating conditions by continuously updating the parameters.

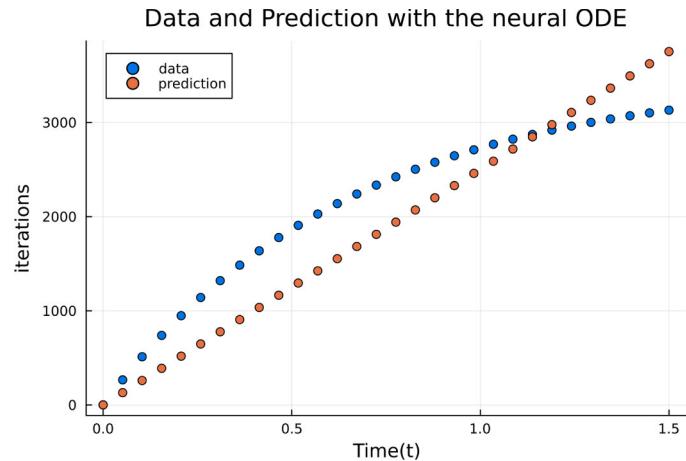


Fig. 16. Neural ODE predictions.

The research methodology outlines the different processes of the research such as data collection, model development, parameter estimation, model validation, model comparison, sensitivity, model refinement if needed, and finally conclusions. **Fig. 15** shows clear conduct of the upcoming results and discussions to communicate the research methodology.

3. Parameter Estimation Results

3.1. Experimental Setup

In this case study, we evaluated the methodology previously mentioned by performing simulations using the stated ODEs differential equations and resolved them statistically. The simulation time was from 0 to 1.5 with 30 consecutive points. With the described model and the initial conditions, the ODE problem was formulated. Then the Tsit5 solver settled the problem. The likelihood of the Bayesian inference of the dynamic microgrid model was incorporated by the neural ordinary differential equations (NeuralODE) approach using the FastChain function from Flux.jl. Additionally, the NeuralODE function from the DiffEqFlux.jl package created a neural network that inputs the system's initial state and the model parameters and outputs the state of the system at different time points. The NeuralODE model was then defined by a loss function that sizes the difference between the predicted solution and the actual simulation data.

The loss function is termed as the kernel of the square-off residuals between the simulation data and the prediction, regularized by the noise standard deviation. In this analysis, the noise standard deviation was supposed to be 0.1. The neural network was trained by the Optim.jl [16] package from Julia to minimize the difference between the output of the neural network and the solution of the differential equation model. The training process was executed in 1000 iterations. A callback function was applied to observe the loss during training and picture the current prediction against the simulation data. The progress of the training was displayed, and a plot showing the data and prediction was updated at each iteration in **Fig. 16**.

Algorithm 1. Bayesian Neural ODE Dynamic Parameter Estimation Framework

Input: - u_0 initial conditions- p parameters initial values - $tspan$ - $tsteps$ - Prior distribution $p(\theta)$ - Number of MCMC samples - Number of hidden layers

Output: - Posterior distribution $p(\theta|y)$

- 1 Initialize neural ODE parameters $\theta_0 \sim p(\theta)$
- 2 Train neural network on measured data y with parameters θ_0 and obtain initial estimates for the parameters θ
- 3 Initialize neural network parameters w with random values
- 4 Set loss function $L = -\log p(y|\theta, w)$
- 5 Sample parameters from prior distribution $\theta \sim p(\theta)$
- 6 Integrate neural ODE with parameters θ to obtain predicted states \hat{x}
- 7 Compute likelihood $p(y|\hat{x}, \theta)$ using the predicted states and measured data
- 8 Compute the prior distribution $p(\theta)$ and posterior distribution $p(\theta|y)$
 - 9 Compute log prior probability: $\log p(\theta)$
 - 10 Compute log likelihood probability: $\log p(y|\hat{x}, \theta)$
 - 11 Compute log posterior probability: $\log p(\theta|y) = \log p(y|\hat{x}, \theta) + \log p(\theta)$
- 12 Compute convergence statistics and return posterior distribution $p(\theta|y)$
- 13 Compute Gelman-Rubin diagnostic statistic R for each parameter θ_i
- 14 Run chains until convergence is achieved for each parameter θ_i
- 15 If \hat{R} is less than 1.1, convergence has been achieved, otherwise, return to step 14.
- 16 Compute effective sample size (ESS) for each parameter θ_i
- 17 Compute Monte Carlo standard error (MCSE) for each parameter θ_i
- 18 Compute the variance of the posterior distribution for each parameter θ_i
- 19 Return the posterior distribution $p(\theta|y)$ with convergence diagnostics including \hat{R} , ESS, and MCSE for each parameter θ_i

End

3.2. Bayesian inference analysis parameter posterior distribution

In this part, we evaluated the probability distribution of the model parameters by assessing the posterior distribution. The posterior distribution is presented in Fig. 17. The plots of this figure showed the trace plots based on the iterations versus the sample value, and density plots based on the sample versus the density of the 43 parameters. Considering the summary statistics such as mean, median, and standard deviation, it is possible to measure the distribution of these parameters and their impact on the model outputs. In this direction, the uncertainty related to the estimated model parameters is then examined with the help of the posterior distribution. By sampling the posterior distribution, we captured the inherent estimation uncertainty regarding the true values of the parameters. This will inform the variability and confidence of the predictions. By the range of plausible parameter values, we also analyzed the reliability and the robustness of the model. The central tendency and the spread of the summary statistics are quantified to facilitate the decision-making in the model findings. Additionally, this summary statistics of the posterior distribution allows us to identify the most influential parameters in the model. A higher posterior density indicates a high value given the observed data suggesting that this parameter is relevant to the model output.

4. Discussion

There are certainly plenty of measures that are useful for helping to measure the efficacy and quality of Bayesian models. Nevertheless, in reality, there is no particular set of rating metrics that are explicitly utilized for verifying Bayesian inference. In the framework of data

analysis, validation reveals the procedure of attesting that the data compiled is precise with the expected application. In this discussion, we will focus on validating the inference results by analyzing the measure and the convergence.

4.1. Model Validation

Being an outcome of the Bayesian approach, the posterior distribution provided a foundational finding. As stated before, it reflects the updated expectations regarding the parameters. To this end, we are going to analyze the dimensions and extent of the posterior distributions. The Turing.jl experiment chain sampling is operated in 10000 iterations. The summary stats of Table 3 are presented to support the convergence and mixing of the chain. In concordance, the density and trace plots of Fig. 17 revealed no apparent patterns or trends that mark a stationary chain.

4.1.1. Mean values and Quantiles

The Turing.jl chain sampling's mean is used as a point estimate to represent the predicted parameter value. The mean value determines whether the parameters are more likely to take a high or small value. From Table 3, we observed the mean values regarding the data central trend. The range of mean values is from 0.001999084 to 539.17431756. This explains that the dataset contains values throughout a wide range of magnitudes. Its average value is the overall mean value. The mean values are also aligned closely to the known values which suggests that the uncertainty is reduced. The parameter values within the 2.50–97.50 % credible interval in Table 4 exhibited a narrow range, indicating that the uncertainty is constrained. Hence, a smaller range in

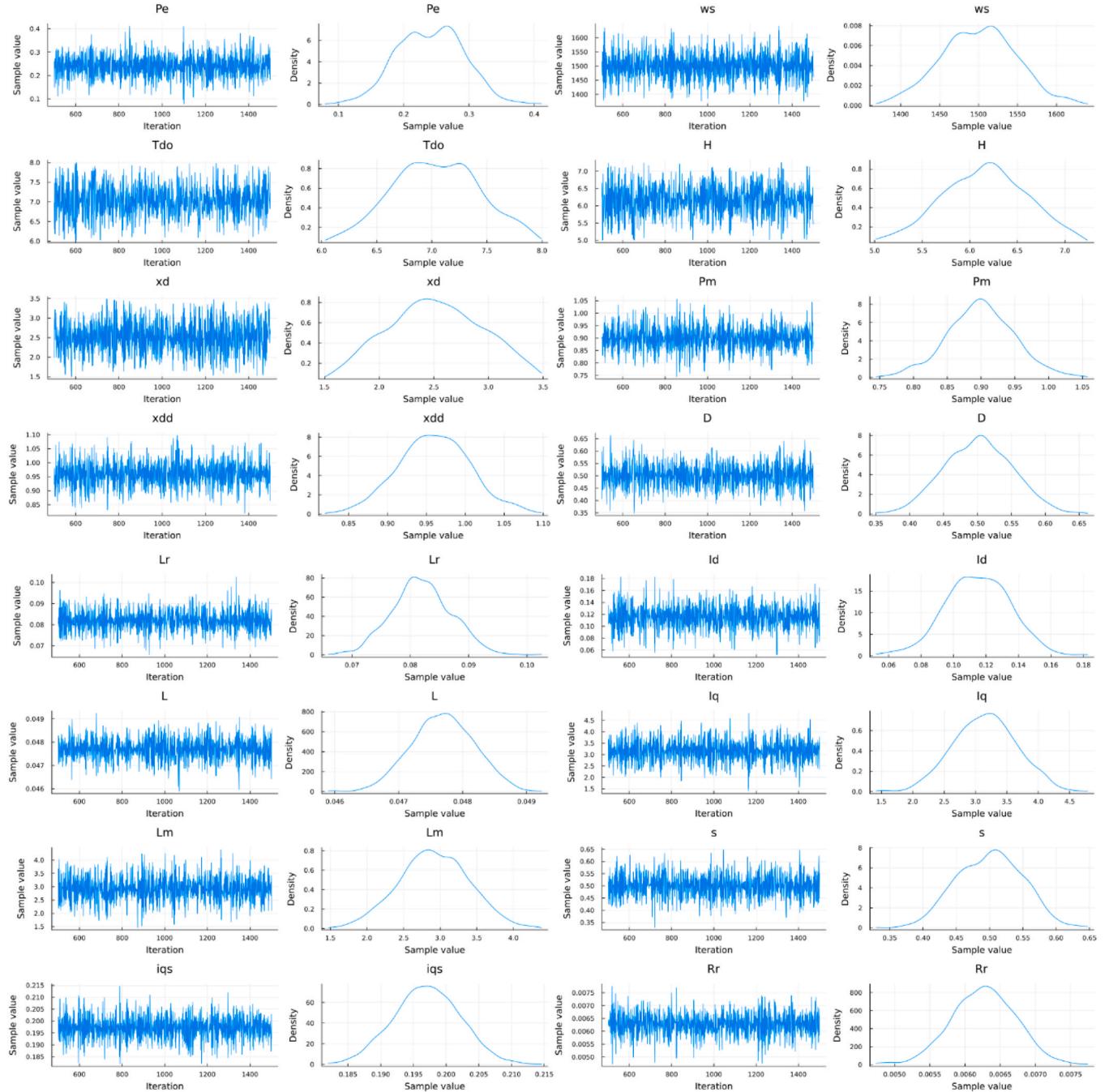


Fig. 17. Trace Plots and density Plots of the Parameter Posterior Distribution predictions.

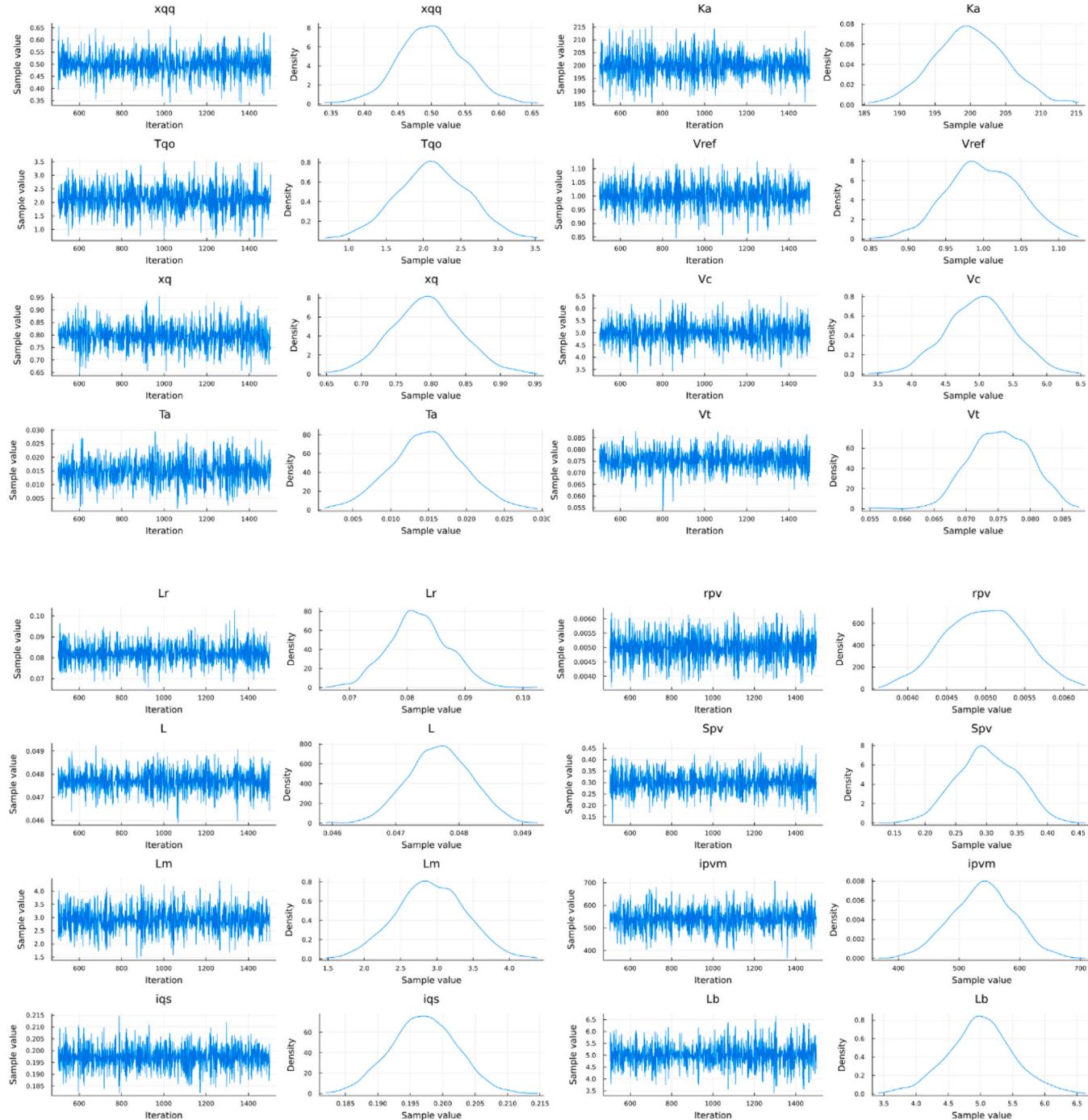


Fig. 17. (continued).

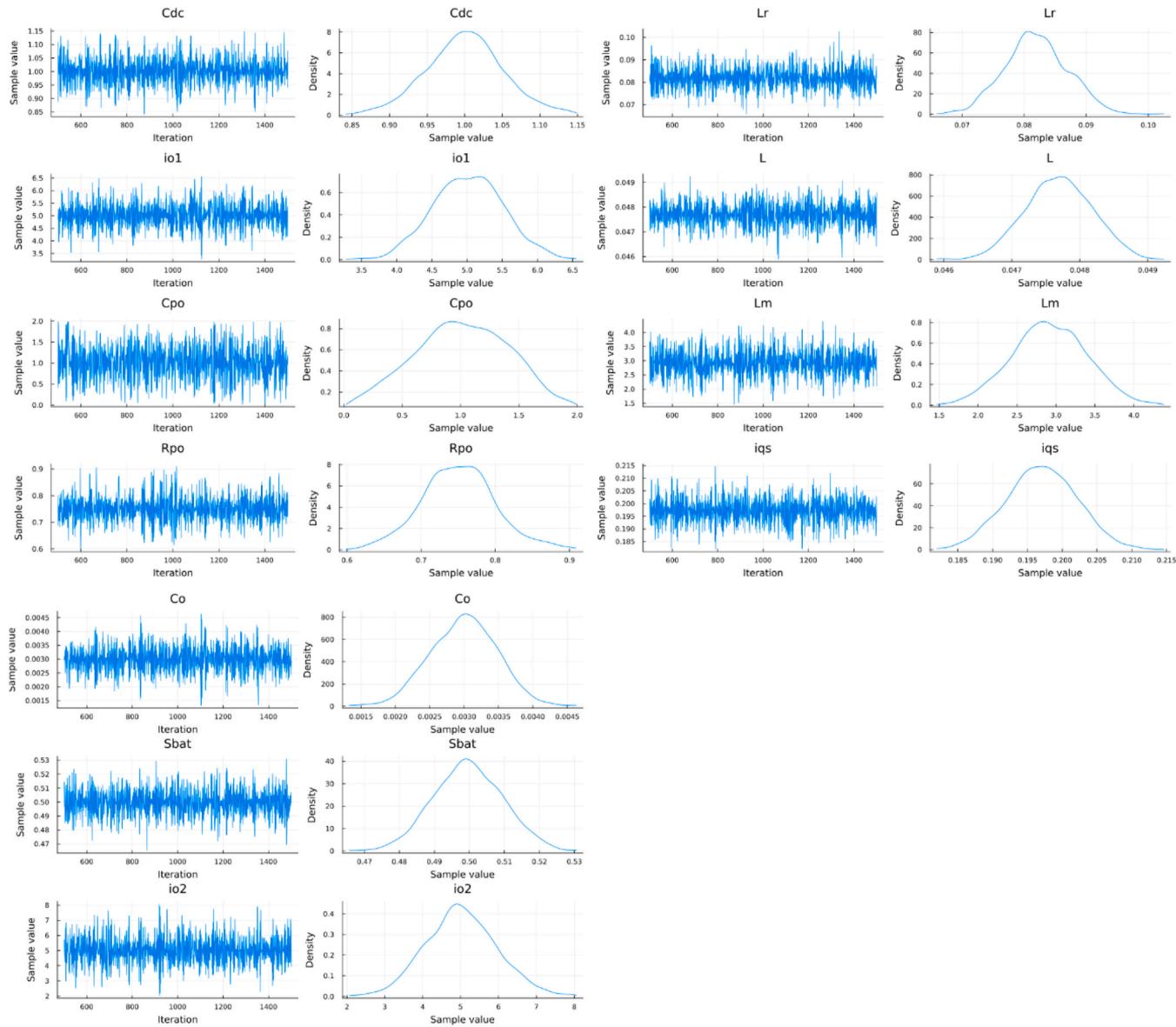


Fig. 17. (continued).

the credible interval suggests that the uncertainty associated with the parameter estimates is reduced and well-controlled. This table details the distribution of each parameter, displaying the quantile statistical results uncertainty estimation at 2.50 %, 25 %, 50 %, 75 %, and 97.50 %. These quantiles illustrate how the parameters are distributed, providing information about their distribution patterns.

4.1.2. MCSE (Markov Chain Monte Carlo Standard Error)

MCSE metric measures the accuracy of statistical models, generally applied in Bayesian models, and informs the degree of variability of the parameter estimates. From the visualization of Table 3, the MCSE values are exceedingly small, which proves the convergence of the MCMC sampling. The MCSE is mathematically expressed as [42]:

$$MCSE = \frac{\sigma^2}{\sqrt{n}}$$

Where σ^2 is the variance of the target distribution and n is the total number of independent MCMC samples.

4.1.3. The Gelman-Rubin diagnostic statistic (Rhat)

The Rhat metric is a tool to evaluate the convergence of the MCMC. In Table 3, the Rhat values for all parameters were shown and revealed that the Rhat values are close to 1 for each parameter. This indicates that the MCMC has converged to the same target distribution. Therefore, these results validate the convergence of the MCMC sampling and enhance the confidence in the accuracy and validity of the presented findings. The Rhat mathematical formulation is expressed as [43]

$$R_{hat} = \sqrt{\frac{\sigma^2}{W}}$$

Where σ^2 is the estimated variance of the target distribution and W is the within-chain variance.

4.1.4. Naïve SE (Standard Error)

An another convergence diagnostic tool, the Naïve SE metric permits to also access to the convergence of the MCMC sampling. In Table 3, the Naïve SE values were computed for each parameter. Likewise, these values are small for all parameters which concludes a stable parameter estimates and shows that MCMC chains explored the parameter space.

Table 3

Summary statistics.

| parameters | mean | std | naive_se | mcse | ess | rhat | ess_per_sec |
|------------|----------|----------|----------|----------|----------|----------|-------------|
| ws | 1500.189 | 48.44952 | 1.532108 | 1.018909 | 2542.336 | 0.999 | 27.45059 |
| H | 6.155078 | 0.464185 | 0.014679 | 0.013314 | 1046.106 | 0.999951 | 11.29521 |
| Pm | 0.89947 | 0.049443 | 0.001564 | 0.001211 | 2629.657 | 0.999273 | 28.39343 |
| D | 0.501251 | 0.050265 | 0.00159 | 0.001192 | 1886.553 | 0.999506 | 20.36985 |
| Pe | 0.241154 | 0.050792 | 0.001606 | 0.000761 | 2545.358 | 0.999088 | 27.48322 |
| Tdo | 7.038533 | 0.415372 | 0.013135 | 0.010359 | 1354.541 | 0.999106 | 14.62551 |
| xd | 2.518163 | 0.443574 | 0.014027 | 0.011004 | 1470.205 | 0.999161 | 15.87437 |
| xdd | 0.960149 | 0.046383 | 0.001467 | 0.000948 | 2215.851 | 1.000337 | 23.9254 |
| xqq | 0.499732 | 0.04879 | 0.001543 | 0.000983 | 2011.072 | 0.999 | 21.71432 |
| Tqo | 2.124334 | 0.509829 | 0.016122 | 0.011804 | 1994.085 | 1.000296 | 21.5309 |
| xq | 0.79339 | 0.050091 | 0.001584 | 0.001348 | 1414.167 | 1.003806 | 15.26931 |
| Ta | 0.014875 | 0.004851 | 0.000153 | 9.61E-05 | 1838.414 | 1.000031 | 19.85007 |
| Ka | 199.9937 | 5.154066 | 0.162986 | 0.083617 | 2222.755 | 0.999004 | 23.99995 |
| Vref | 1.000341 | 0.047837 | 0.001513 | 0.001053 | 1946.008 | 0.999152 | 21.01181 |
| Vc | 5.00925 | 0.495998 | 0.015685 | 0.010315 | 2078.204 | 0.999006 | 22.43917 |
| Vt | 0.075329 | 0.004702 | 0.000149 | 0.000113 | 1531.234 | 1.00007 | 16.53333 |
| Id | 0.114966 | 0.02026 | 0.000641 | 0.000509 | 1629.433 | 0.999426 | 17.59361 |
| Iq | 3.155655 | 0.505054 | 0.015971 | 0.011541 | 1960.488 | 0.999046 | 21.16815 |
| s | 0.500539 | 0.049004 | 0.00155 | 0.001186 | 2263.002 | 0.999661 | 24.43451 |
| Rr | 0.006265 | 0.000456 | 1.44E-05 | 1.00E-05 | 2355.015 | 0.999008 | 25.42801 |
| Lr | 0.081963 | 0.005077 | 0.000161 | 9.36E-05 | 2719.087 | 0.999902 | 29.35903 |
| L | 0.047692 | 0.000493 | 1.56E-05 | 1.21E-05 | 1812.699 | 1.001081 | 19.57241 |
| Lm | 2.915542 | 0.480017 | 0.015179 | 0.008449 | 2005.446 | 0.999866 | 21.65358 |
| iqs | 0.197058 | 0.005068 | 0.00016 | 0.000121 | 2610.875 | 0.99965 | 28.19063 |
| ids | -2.0012 | 0.050394 | 0.001594 | 0.000932 | 2808.767 | 0.999 | 30.32735 |
| vdr | 0.403069 | 0.047613 | 0.001506 | 0.001071 | 1940.441 | 0.999286 | 20.9517 |
| vqr | 0.401632 | 0.049361 | 0.001561 | 0.000963 | 2452.329 | 0.999341 | 26.47875 |
| Cpvm | 10.0009 | 0.483719 | 0.015297 | 0.011096 | 2028.227 | 0.999023 | 21.89955 |
| rpv | 0.004987 | 0.000499 | 1.58E-05 | 1.25E-05 | 2681.242 | 0.999057 | 28.95041 |
| Spv | 0.301676 | 0.049662 | 0.00157 | 0.001003 | 2342.085 | 0.999025 | 25.2884 |
| ipvm | 539.1743 | 50.96189 | 1.611556 | 1.326067 | 1950.483 | 0.999292 | 21.06012 |
| Lb | 5.002968 | 0.510858 | 0.016155 | 0.010492 | 2512.732 | 1.00006 | 27.13094 |
| Cdc | 1.000272 | 0.051919 | 0.001642 | 0.001159 | 2370.291 | 0.999319 | 25.59295 |
| io1 | 5.017142 | 0.501763 | 0.015867 | 0.009295 | 2000.843 | 0.99901 | 21.60388 |
| Cpo | 1.012337 | 0.424997 | 0.01344 | 0.009595 | 1877.671 | 0.999978 | 20.27394 |
| Rpo | 0.750596 | 0.04988 | 0.001577 | 0.000874 | 2660.995 | 0.999 | 28.73179 |
| R | 0.001999 | 0.000105 | 3.32E-06 | 2.12E-06 | 2327.687 | 0.999014 | 25.13294 |
| Rbinit | 0.027828 | 0.00485 | 0.000153 | 0.000116 | 1772.405 | 1.001129 | 19.13734 |
| Eb | 47.82876 | 4.421644 | 0.139825 | 0.094614 | 1712.863 | 0.999358 | 18.49444 |
| Lc | 9.951012 | 1.93678 | 0.061246 | 0.048783 | 1712.996 | 0.999156 | 18.49589 |
| Co | 0.002996 | 0.000486 | 1.54E-05 | 9.35E-06 | 2267.964 | 0.999009 | 24.48808 |
| Sbat | 0.499676 | 0.009832 | 0.000311 | 0.000183 | 3000 | 0.999535 | 32.39216 |
| io2 | 5.024143 | 0.941093 | 0.02976 | 0.020366 | 2822.636 | 0.999789 | 30.4771 |

The Standard Error is mathematically expressed as [43]

$$SE = \frac{\text{Var}(x)}{C.S}$$

Where $\text{Var}(x)$ is the variance of the posterior samples, C is the number of chains, and S is the number of samples.

4.1.5. Effective Sample Size (ESS)

ESS determines whether the chains have converged towards the targeted distribution in the posterior distribution of the computational MCMC. The ESS is a measure of independent information in Bayesian inference. It defines the number of independent samples in the MCMC. Furthermore, convergence analysis is the tool that measures the speed of the MCMC algorithm. Its results can be summarized in a Table or a Figure. Table 3 presents the ESS values for each parameter. General knowledge like a rule of thumb tells that ESS values should exceed one hundred for a parameter. If there are some possibilities that an ESS value is less than one hundred, it is obligated to increase or adjust the number of MCMC samples. Ideally, high ESS values demonstrate that the parameters of each sample are independent and contribute information to the posterior distribution. When the ESS values are low, it means they are highly autocorrelated they can cause slower convergence and less accurate inference. Fig. 18 shows a stable pattern that the convergence has been achieved. In this graph, the ESS values are seen to be superior to 1000. Fig. 18 also shows the arrangement of the parameters ESS

values in descending order to identify the parameters with the highest ESS values. A high ESS value suggests that MCMC has converged well for that parameter and is likely to have a more accurate estimation compared to the parameter with the lower ESS values. The ESS is mathematically expressed as [42]:

$$ESS = \frac{N}{1 + \sum_{t=1}^{\infty} \rho_t}$$

Where N is the total number of MCMC samples and ρ_t is the autocorrelation at lag t.

4.2. Investigation of the significance of model parameters

In this part, our aim is to determine the most significant parameters of our model using z-scores. We employed the statistical mean values and standard deviations from the summary statistics with parameter names. The z-scores are formulated by the absolute mean value division with the standard deviation. The z-scores present the measurements of how many standard deviations are away from the mean to show the relative parameter significance. Fig. 19 depicts the bar plot in descending order of the z-score parameter significance. This plot shows the magnitude of the z-scores of each parameter. The higher values are the ones with the greatest significance and are displayed with their parameter names on the y-axis while the corresponding z-scores are on

Table 4
Quantiles of the parameter estimation.

| parameters | 2.50 % | 25.00 % | 50.00 % | 75.00 % | 97.50 % |
|------------|----------|----------|----------|----------|----------|
| ws | 1405.812 | 1467.281 | 1501.21 | 1533.012 | 1597.875 |
| H | 5.201458 | 5.838326 | 6.172142 | 6.485292 | 7.053772 |
| Pm | 0.798391 | 0.866876 | 0.899014 | 0.932547 | 0.999711 |
| D | 0.404189 | 0.466043 | 0.502436 | 0.536302 | 0.59936 |
| Pe | 0.143858 | 0.205313 | 0.242494 | 0.277696 | 0.334145 |
| Tdo | 6.229618 | 6.745793 | 7.0328 | 7.323195 | 7.843888 |
| xd | 1.703214 | 2.201099 | 2.501477 | 2.844206 | 3.364202 |
| xdd | 0.867515 | 0.928936 | 0.960509 | 0.99038 | 1.056087 |
| xqq | 0.403954 | 0.467479 | 0.498767 | 0.530217 | 0.600056 |
| Tqo | 1.107717 | 1.78697 | 2.121307 | 2.45518 | 3.112163 |
| xq | 0.695102 | 0.760458 | 0.792969 | 0.8265 | 0.893045 |
| Ta | 0.0053 | 0.011854 | 0.014884 | 0.017972 | 0.024571 |
| Ka | 190.1141 | 196.6595 | 199.905 | 203.4283 | 209.8541 |
| Vref | 0.904867 | 0.966881 | 0.997815 | 1.034905 | 1.093809 |
| Vc | 4.060329 | 4.673513 | 5.017929 | 5.338664 | 5.952995 |
| Vt | 0.066715 | 0.072076 | 0.075428 | 0.078773 | 0.083994 |
| Id | 0.074363 | 0.101334 | 0.114909 | 0.129236 | 0.153509 |
| Iq | 2.178212 | 2.806536 | 3.162397 | 3.49403 | 4.106946 |
| s | 0.408452 | 0.465114 | 0.502399 | 0.534724 | 0.590888 |
| Rr | 0.005354 | 0.005957 | 0.006268 | 0.006579 | 0.007138 |
| Lr | 0.072588 | 0.078641 | 0.08182 | 0.085184 | 0.091811 |
| L | 0.046744 | 0.047368 | 0.047701 | 0.04803 | 0.048635 |
| Lm | 1.962374 | 2.592321 | 2.905402 | 3.234788 | 3.858686 |
| iqs | 0.187362 | 0.193731 | 0.197073 | 0.200519 | 0.206915 |
| ids | -2.09648 | -2.03763 | -2.00187 | -1.96433 | -1.90712 |
| vdr | 0.307936 | 0.374964 | 0.402072 | 0.43252 | 0.498526 |
| vqr | 0.305315 | 0.366189 | 0.400016 | 0.437224 | 0.497654 |
| Cpvm | 9.068878 | 9.674214 | 9.998849 | 10.32227 | 10.93388 |
| rpv | 0.003991 | 0.004633 | 0.004993 | 0.005336 | 0.005941 |
| Spv | 0.211073 | 0.267055 | 0.299638 | 0.338573 | 0.393645 |
| ipvm | 437.9343 | 504.8973 | 539.9853 | 572.0655 | 639.9073 |
| Lb | 3.973653 | 4.677163 | 5.005766 | 5.321701 | 6.050855 |
| Cdc | 0.897286 | 0.966497 | 1.000445 | 1.033583 | 1.105907 |
| io1 | 4.060576 | 4.67519 | 5.027587 | 5.352816 | 6.000636 |
| Cpo | 0.173805 | 0.725593 | 1.008791 | 1.3164 | 1.826358 |
| Rpo | 0.651167 | 0.717737 | 0.749566 | 0.782474 | 0.858784 |
| R | 0.001783 | 0.001927 | 0.002001 | 0.00207 | 0.002198 |
| Rbinit | 0.018763 | 0.024687 | 0.027678 | 0.031184 | 0.037245 |
| Eb | 39.70005 | 44.61252 | 47.80307 | 51.00079 | 56.15412 |
| Lc | 6.396617 | 8.566738 | 9.926544 | 11.24014 | 13.92293 |
| Co | 0.002054 | 0.00266 | 0.002999 | 0.003324 | 0.003916 |
| Sbat | 0.480514 | 0.492863 | 0.499425 | 0.506573 | 0.518596 |
| io2 | 3.246765 | 4.395867 | 4.994904 | 5.623047 | 6.922971 |

the x-axis. In the Figure display, the 10 top parameters and key factors of the model's behavior are observed which can be used in future research for potential research areas or model improvements and refinements.

4.3. Model Comparison

We tested the method by comparing five different likelihood functions to gauge which dynamic microgrid model fits the observed data. This test is indispensable for statistical analysis and inference. We can assess the corresponding fit of different models or hypotheses to the data and decide which model is more convincingly backed by the evidence. These likelihood functions in Table 5 were used in the framework of the Neural ODEs model. Table 5 presents the beforehand described likelihood functions formulas where P is the pdf of each distribution and noise_std is the noise of the standard deviation. These different likelihood functions are used to capture the characteristics of the data and to ease the comparison between the models. Then the accurate likelihood function distribution that assesses the goodness of fit is decided.

4.3.1. The Mean Squared Error (MSE) likelihood

The MSE is a type of likelihood that quantifies the squared differences between the observed data and the predictions. The residuals are calculated by the differences of the data and the predictions. The pdf helps to establish the likelihood. The negative logarithm of the pdf

provides the final value of the likelihood [44].

4.3.2. The Negative Log-Likelihood (NLL) likelihood

The NLL is the sum of the pdf log values of a normal for each residual. The calculation of the residuals and likelihood by the pdf is the same as the MSE but the difference is that NLL computes directly the negative logarithm of the likelihood [45].

4.3.3. The Poisson likelihood function

The Poisson likelihood function considers that the observed data follows a Poisson distribution. The likelihood is given here by the sum of the log pdf values of the Poisson distribution for each predicted value [46].

4.3.4. The Student's Likelihood function

This function considers that the residuals follow the t-student's distribution. Also, in the next step, it evaluates the log pdf of the t-student distribution with 5 degrees of freedom for each residual. Then, the sum of the negative logarithm of the sum of these pdf provided the likelihood value [47].

4.3.5. The Laplace likelihood function

In this likelihood, the residuals between the observed data and the predictions follow a Laplace distribution [48]. The calculation of the

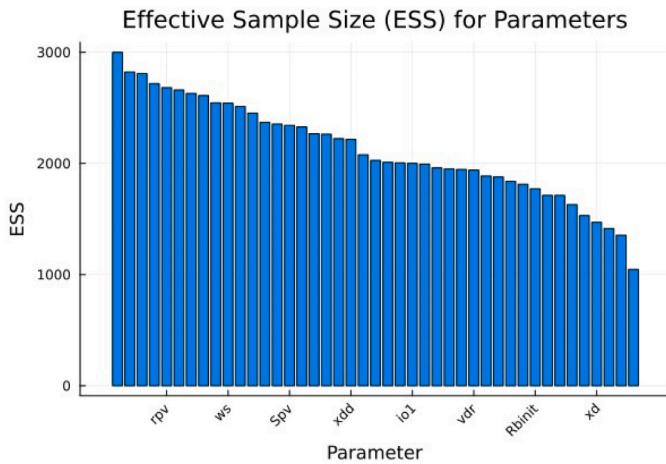


Fig. 18. ESS values visualization in Descending Order.

residuals and likelihood by the pdf is the same as the MSE and NLL. And as with the MSE, the negative logarithm of the pdf provides the final value of the likelihood.

In each of the 6 scenarios, we changed each time the likelihood function in the code to observe and compare the results. The deviance, p_DIC which is the variance of the log-likelihood, and then DIC which stands for deviance information criterion across the six models are contrasted. Additionally, it is also important to tell that the DIC metric is the combination of the deviance and p_DIC metrics. Our actual model is Model 1 followed by the inclusion of varying likelihoods of the other 5 models. The deviance across the six models is presented in Fig. 20. However, Model 2, Model 4, Model 5, and Model 6 exhibited higher p_DIC values as observed in Fig. 21, and Model 1 and Model 3 showed the lowest p_DIC values. As shown in Fig. 22, Model 1 and Model 3 have the lowest DIC. Therefore, Model 1 and Model 3 have the best fit to the data among the evaluated models. Model 4 and Model 6 also got the second lowest DIC after Model 1 and Model 3 which shows that they have a good fit to the data but not as good as Model 1 and Model 3. These results highlight the goodness of fit by the DIC metrics and model complexity by the p_DIC metric. These insights are used by researchers to decide the appropriate model for their selected applications. To support the goodness of fit of these results comparisons, it is intended to compare the Akaike Information Criterion (AIC) and the Bayesian Information Criterion (BIC) of the 6 different statistical models. In the given models, AIC and BIC values are calculated. The AIC is calculated by the sum of the computed total log-likelihood and the number of parameters. The BIC is also calculated by the same principle but with a penalty for the number of parameters. Fig. 23 presents a bar plot that represents each model with a different color to visualize the models with the lowest AIC and BIC. From Fig. 19, Models 3, 4, and 6 have the lowest AIC and BIC values. Lower AIC and BIC values suggest a better trade-off between fit and model complexity compared to the other models. In all, Model 1 and Model 3 have the best fit to the data, and Model 3, Model 4, and Model 6 have a good fit with lower model complexity. To summarize, the difference lies in accuracy versus simplicity. Research context will guide to choice between these two approaches.

4.4. Sensitivity analysis

4.4.1. Assessing System Performance and Stability with Parameter Adjustments

To assess the influence of the prior distributions of the inference, a sensitivity analysis is conducted. The parameter ws was increased to 2000, representing an increase in the nominal frequency. This change may have implications for the system's overall performance and stability. Therefore, two scenarios are created: an original scenario and the

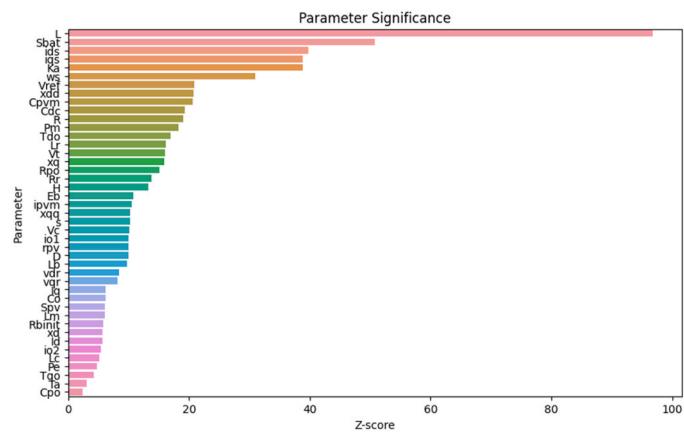


Fig. 19. Bar Plot in descending order of the z-scores parameter significance.

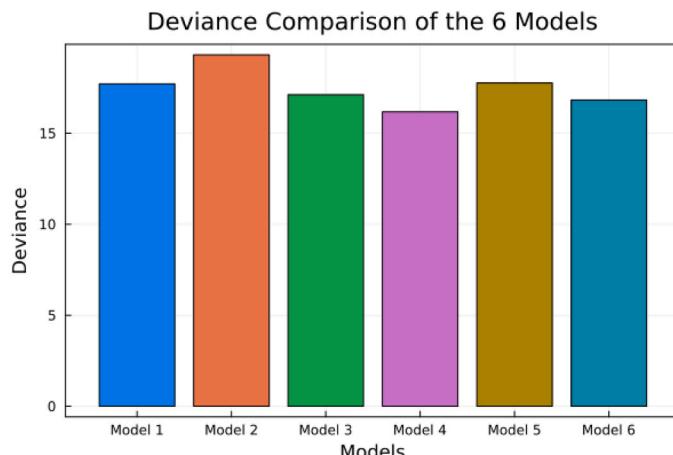
Table 5
Likelihood functions.

| Likelihood Function | Likelihood Formula |
|-------------------------------|---|
| Mean Squared Error | $-\log(P(\text{MSE})) = -\log(P(\text{Normal}(0, \text{noise_std}), \text{mse}))$ |
| Negative Log-Likelihood (NLL) | $-\sum(\log(P(\text{NLL})) = -\sum(\log(P(\text{Normal}(0, \text{noise_std}), \text{residuals})))$ |
| Poisson Likelihood | $-\sum(\log(P(\text{Poisson}(\text{predictions}), \text{data})))$ |
| Student's t Likelihood | $-\sum(\log(P(T(5), \text{residuals})))$ |
| Laplace Likelihood | $-\sum(\log(P(\text{Laplace}(0, \text{noise_std}), \text{residuals})))$ |

one modified with an updated prior. Then the posterior samples are extracted in each case from the posterior chains. Fig. 24 presents the trace plots posterior distributions. It shows that trace plots in the modified case completely moved upwards. Furthermore, the mean and standard deviation in each scenario are calculated and represented as horizontal lines in the plots to visualize the central tendency of the posterior distributions. A huge difference in mean and standard deviation values is observed. The mean goes from 1500 to around 1790. Similarly, the standard deviation changed from 50 to 10. In the modified scenario, a slightly shifted and narrower posterior distribution has been observed compared to the original one, as observed in Fig. 25. The uncertainty which is here represented by the updated prior value is controlled. The system is so robust that it detects the little and smallest change in one parameter. From this, it is knowledgeable that a change in initial parameters can alter the parameter values as seen in Fig. 24 and Fig. 25. Similarly, we can modify the parameters of the microgrid voltage rating, and parameter of the state of charge of the battery to assess the impact of these changes on the system's performance, stability, and efficiency. These results of the parameter variations are important in the design and operation of the system under different operating conditions. To sum up, this sensitivity analysis showed the importance of the prior data in Bayesian inference, supports the robustness of the findings, and contributes to clearly understanding the topic.

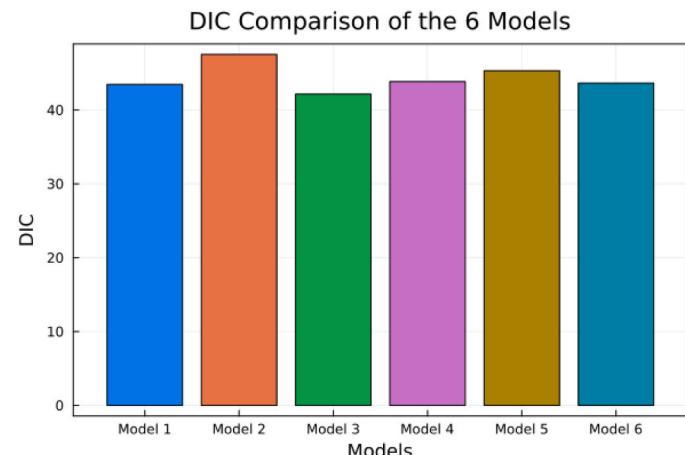
4.4.2. Impact of Parameter Modifications on Microgrid Dynamics

In this analysis, it was important to know the effect of changes in parameter values for microgrid dynamic analysis. In this case, the study was based on the modification of parameters such as the storage battery capacity, which is defined by the Cpo parameter, and also on modifications to the nominal frequency w and the exciter time constant Ta, to determine their influence and the change into the diesel generator power. As illustrated in Figs. 26 and 27, the original models in both cases show significant fluctuations in a spike of around 10 seconds when the model parameters are modified. These observations are responses to the system dynamics and parameterization.

**Fig. 20.** Deviance Comparison for the 6 models.

4.5. Evaluating the Contributions, Advantages, and Limitations of Integrating Neural ODEs with Bayesian Inference

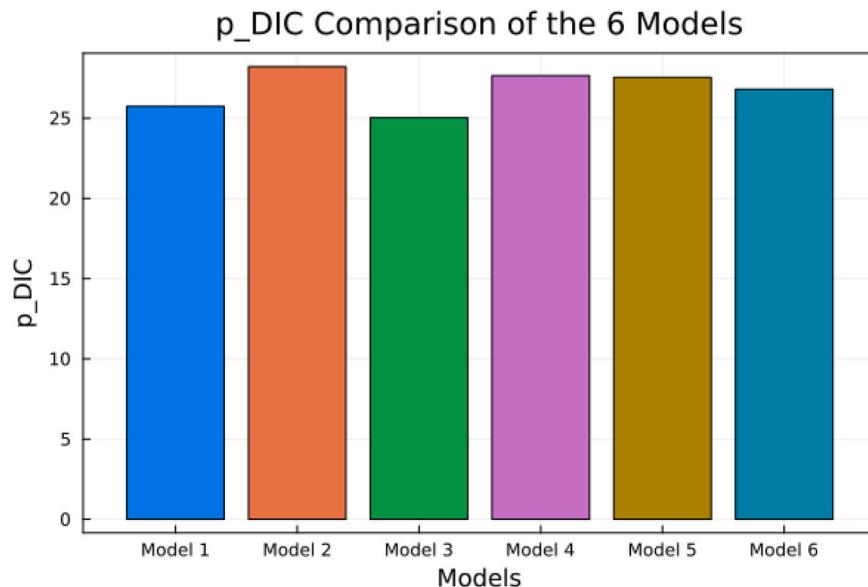
In recent years, major advances have been made in the development of dynamic analysis models for microgrid systems. Traditional methods based on the mathematics of differential equations and sometimes with the possibility of integration with machine learning methods have been used to study the dynamics of microgrids[49]. These traditional methods have limitations such as the need to fix certain important parameters. These methods are certainly functional but generally lack the flexibility to adapt to real data and are difficult to adapt to estimate uncertainties in large and complex systems such as microgrids containing intermittent energy sources. In contrast, Neural ODE is an innovative emergent model that corresponds to a change and can provide solutions and propose parameter adjustments to models with data-driven learning. By inserting neural networks into differential equations, Neural ODE distinguishes itself from other methods with its structure that easily captures dynamism with precision[50]. This study was therefore implemented with the help of Turing.jl, a Julia programming package that enables probabilities to be programmed and Bayesian Inference to be implemented in the Neural ODE, to obtain an estimate of each parameter in the system of differential equations. In this integration, Bayesian Inference enables uncertainties to be quantified

**Fig. 22.** DIC Comparison for the 6 models.

with the help of a priori information[51]. The NUTS algorithm facilitates the sampling of posterior distributions[52]. Also, the introduction of noise in the data simulation and the Bayesian inference makes this study similar to a real-time measurement. The approach used in this paper is consequently aligned with the model projection of the probability studies and data assimilation which focus is creating models that are correct and resilient to uncertainties[53]. But now computational applications introduce difficulties such as increasing processing times and potential problems with model interpretation. The complexity of Neural ODE and Bayesian inference, which is an intensive model to train, will cause multiple iterations to train the model for processing and analysis[50]. Furthermore, neural networks, such as neural ODEs, are a black box that obscures the validation of model results and decision-making, then can have repercussions on critical infrastructures such as microgrids[49]. In all, Fig. 28 summarizes the key points of Neural ODEs with Bayesian Inference for instance parameter flexibility, accuracy, and real-data handling and its downgrade points such as computational load, cost efficiency, and transparency.

5. Conclusion

The effects of greenhouse gases have been felt in recent years, prompting the world to turn to renewable energy sources. Nevertheless,

**Fig. 21.** p_DIC Comparison for the 6 models.

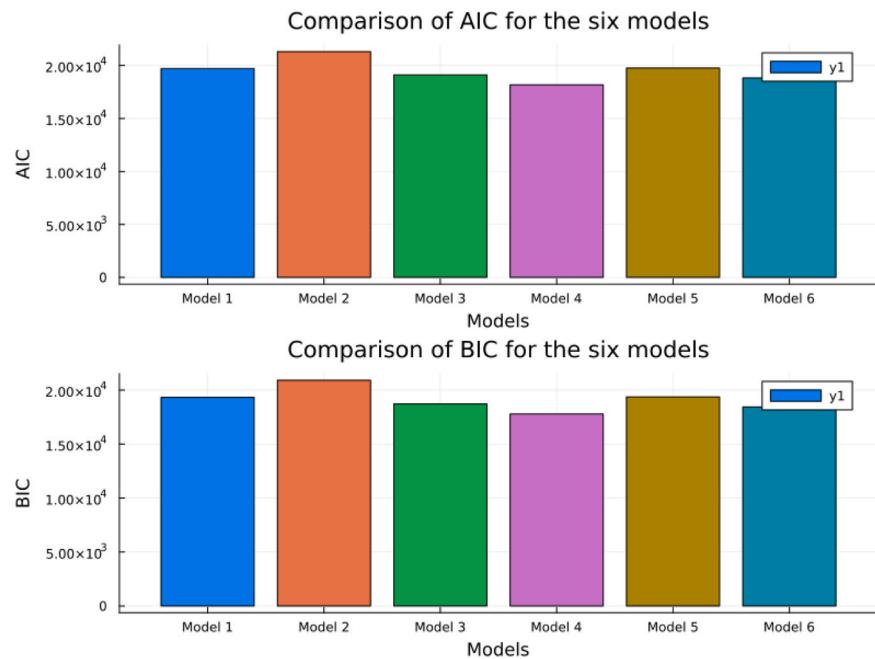


Fig. 23. Comparison of AIC and BIC for the 6 models.

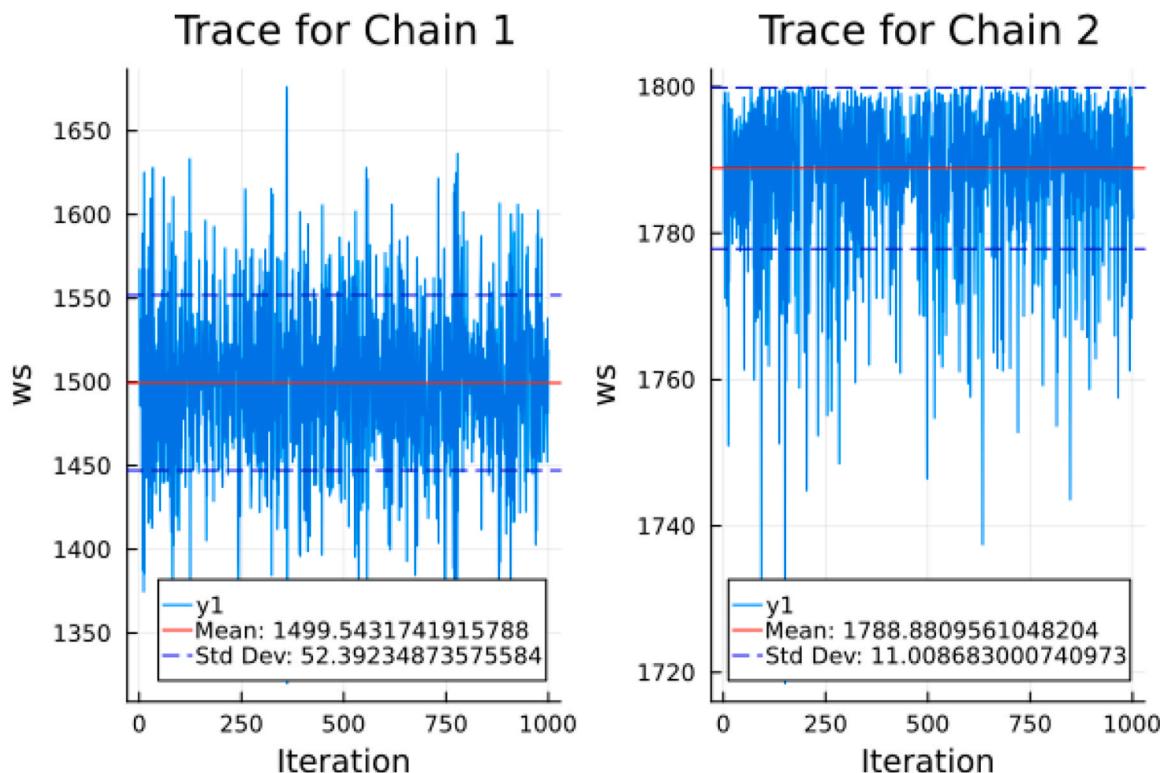


Fig. 24. Trace Plots Comparison for the original and modified posterior chain distribution.

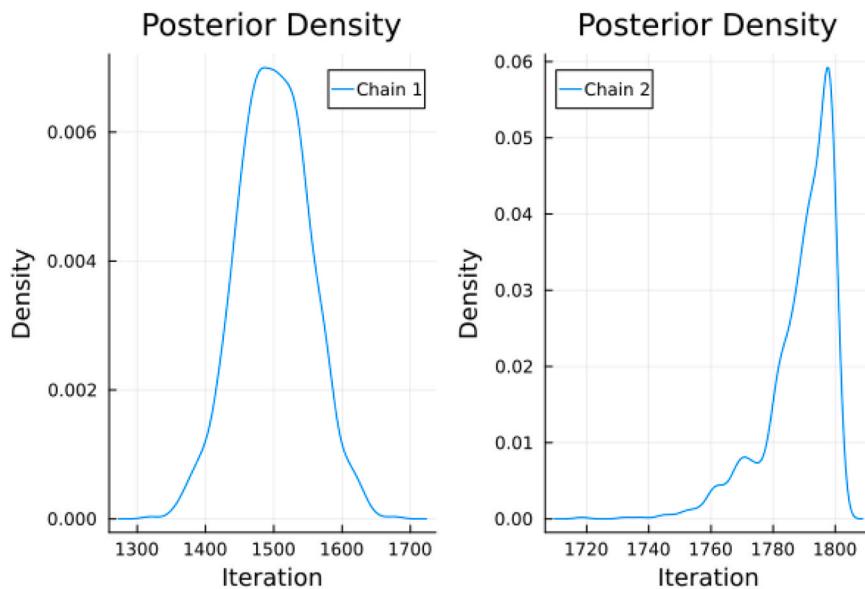


Fig. 25. Density Plots Comparison for the original and modified posterior chain distribution.

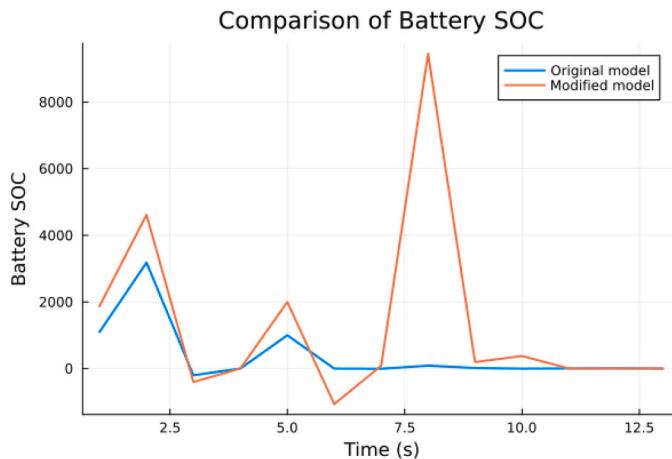


Fig. 26. Comparison of Battery SOC.

renewable resources such as solar and wind power are highly volatile. For this reason, research has turned to optimizing the uses of these sources. One of the initial priorities is to manage and regulate the energy derived from renewable sources effectively. For optimized control, it is crucial to understand the variables and parameters implicated in dynamic models such as microgrids. In this context, this research paper proposes addressing microgrid control parameter estimation by integrating Bayesian inference with the Neural ODE method. This methodology was valuable in terms of contributing to the microgrid decision-making capabilities under different operating conditions. We modeled the posterior probability distribution of each of the parameters of a set of differential equations that defines the dynamic microgrid system by performing simulations and resolving the uncertainty statistically. The convergence diagnostics statistics (Rhat, Naïve SE, ESS, and MCSE) demonstrated strong evidence that the MCMC sampling has effectively converged and that the uncertainty is constrained. The credibility of this analysis outcomes validates the reliability of the obtained parameter estimates. The potential of this research may be explored in future research and applied to other domains with similar dynamic modeling requirements.

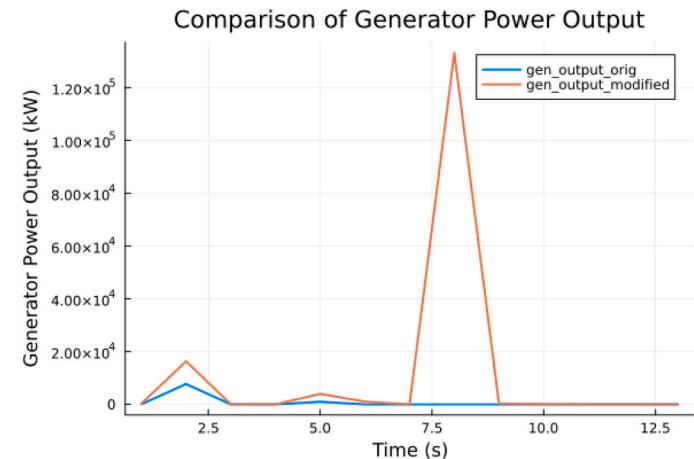


Fig. 27. Comparison of Generator Power Output.

CRediT authorship contribution statement

Fathi Farah Fadoul: Conceptualization, Data curation, Formal analysis, Funding acquisition, Investigation, Methodology, Project administration, Resources, Software, Validation, Visualization, Writing – original draft, Writing – review & editing. **Ramazan Çağlar:** Conceptualization, Investigation, Methodology, Supervision, Validation, Visualization, Writing – review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

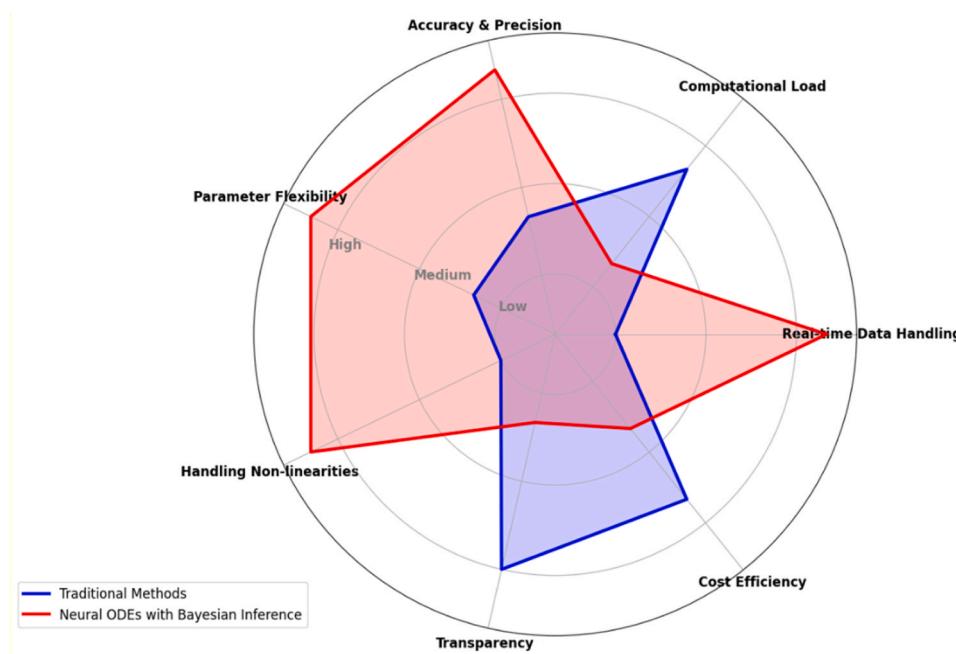


Fig. 28. Comparison of Traditional Methods and Neural ODEs with Bayesian Inference.

References

- [1] D. S. T and O.D.W.H. of T.S.T.U.G.T.S.-S.A.S.M.J.H.I.F.E.A. Laura Cozzi, “(www.iea.org)”, For the first time in decades, the number of people without access to electricity is set to increase in 2022.
- [2] Paris: IEA, “International Energy Agency (IEA) World Energy Outlook.”
- [3] World Bank, “The Cost of Energy Infrastructure: A Global Review. Washington, DC: World Bank.”
- [4] United Nations Development Programme (UNDP), “Energy Access for All: The Energy Progress Report 2019. New York: UNDP.”
- [5] World Bank, “The State of the World’s Infrastructure 2019.”
- [6] United Nations Environment Programme (UNEP), “Emissions Gap Report 2019.”
- [7] International Renewable Energy Agency (IRENA), “Renewables 2022: Global Status Report. Abu Dhabi: IRENA.”
- [8] World Bank, “Global Trends in Renewable Energy Investment 2021. Washington, DC: World Bank.”
- [9] Navigant Research, “Microgrids Market: Global Industry Analysis and Opportunity Assessment, 2019-2029. Boston, MA: Navigant Research.”
- [10] H. Jiayi, J. Chuanwen, X. Rong, A review on distributed energy resources and MicroGrid, in: Renewable and Sustainable Energy Reviews, vol. 12, Elsevier Ltd, 2008, pp. 2472–2483, <https://doi.org/10.1016/j.rser.2007.06.004>.
- [11] Y. Li, F. NEJABATKHAH, Overview of control, integration and energy management of microgrids, *J. Mod. Power Syst. Clean. Energy* vol. 2 (3) (Jan. 2014) 212–222, <https://doi.org/10.1007/s40565-014-0063-1>.
- [12] F. Ramahatana, J. Le Gal La Salle, P. Lauret, M. David, A more efficient microgrid operation through the integration of probabilistic solar forecasts, *Sustain. Energy, Grids Netw.* vol. 31 (Sep. 2022), <https://doi.org/10.1016/j.segan.2022.100783>.
- [13] S. Mohseni, M.S. Pishvaae, R. Dashti, Privacy-preserving energy trading management in networked microgrids via data-driven robust optimization assisted by machine learning, *Sustain. Energy, Grids Netw.* vol. 34 (Jun. 2023), <https://doi.org/10.1016/j.segan.2023.101011>.
- [14] A. Lawson, M. Goldstein, C.J. Dent, Bayesian framework for power network planning under uncertainty, *Sustain. Energy, Grids Netw.* vol. 7 (Sep. 2016) 47–57, <https://doi.org/10.1016/j.segan.2016.05.003>.
- [15] S.M. Mirafabzadeh, M. Longo, High-resolution PV power prediction model based on the deep learning and attention mechanism, *Sustain. Energy, Grids Netw.* vol. 34 (Jun. 2023) 101025, <https://doi.org/10.1016/j.segan.2023.101025>.
- [16] F.F. Fadoul, A.A. Hassan, R. Caglar, Assessing the feasibility of integrating renewable energy: decision tree analysis for parameter evaluation and LSTM forecasting for solar and wind power generation in a campus microgrid, *IEEE Access* vol. 11 (2023), <https://doi.org/10.1109/ACCESS.2023.3328336>.
- [17] N. Petra, C.G. Petra, Z. Zhang, E.M. Constantinescu, M. Anitescu, A Bayesian approach for parameter estimation with uncertainty for dynamic power systems, *IEEE Trans. Power Syst.* vol. 32 (4) (Jul. 2017) 2735–2743, <https://doi.org/10.1109/TPWRS.2016.2625277>.
- [18] M. Pandey, B. Lie, Bayesian Inference for thermal model of synchronous generator part i: parameter estimation, *IEEE Access* vol. 10 (2022) 103529–103537, <https://doi.org/10.1109/ACCESS.2022.3209232>.
- [19] M. Sharifzadeh, A. Sikinioyi-Lock, N. Shah, Machine-learning methods for integrated renewable power generation: a comparative study of artificial neural networks, support vector regression, and Gaussian Process Regression, *Renew. Sustain. Energy Rev.* vol. 108 (Jul. 2019) 513–538, <https://doi.org/10.1016/j.rser.2019.03.040>.
- [20] C. Cai, H. Liu, Y. Tao, Z. Deng, W. Dai, J. Chen, Microgrid equivalent modeling based on long short-term memory neural network, *IEEE Access* vol. 8 (2020) 23120–23133, <https://doi.org/10.1109/ACCESS.2020.2966238>.
- [21] P.N. Papadopoulos, T.A. Papadopoulos, P. Crolla, A.J. Roscoe, G.K. Papagiannis, G. M. Burt, Black-box dynamic equivalent model for microgrids using measurement data, *IET Gener., Transm. Distrib.* vol. 8 (5) (2014) 851–861, <https://doi.org/10.1049/iet-gtd.2013.0524>.
- [22] B. Zaker, G.B. Gharehpetian, M. Karrari, Small signal equivalent model of synchronous generator-based grid-connected microgrid using improved Heffron-Phillips model, *Int. J. Electr. Power Energy Syst.* vol. 108 (Jun. 2019) 263–270, <https://doi.org/10.1016/j.ijepes.2019.01.016>.
- [23] I.Y. Chung, W. Liu, D.A. Cartes, K. Schoder, Control parameter optimization for a microgrid system using particle swarm optimization, *2008 IEEE Int. Conf. Sustain. Energy Technol., ICSET 2008* (2008) 837–842, <https://doi.org/10.1109/ICSET.2008.4747124>.
- [24] B. Zaker, G.B. Gharehpetian, M. Karrari, A novel measurement-based dynamic equivalent model of grid-connected microgrids, *IEEE Trans. Ind. Inf.* vol. 15 (4) (Apr. 2019) 2032–2043, <https://doi.org/10.1109/TII.2018.2856852>.
- [25] F. Conte, F. D’Agostino, S. Massucco, F. Silvestro, C. Bossi, M. Cabiat, Experimental validation of a dynamic equivalent model for microgrids, *IEEE Trans. Ind. Appl.* vol. 57 (3) (May 2021) 2202–2211, <https://doi.org/10.1109/TIA.2021.3064522>.
- [26] M. Ahmeid, M. Armstrong, S. Gadoe, M. Al-Greer, P. Missailidis, Real-time parameter estimation of DC-DC converters using a self-tuned kalman filter, *IEEE Trans. Power Electron.* vol. 32 (7) (Jul. 2017) 5666–5674, <https://doi.org/10.1109/TPEL.2016.2606417>.
- [27] F.S. Cattivelli, C.G. Lopes, A.H. Sayed, Diffusion recursive least-squares for distributed estimation over adaptive networks, *IEEE Trans. Signal Process.* vol. 56 (5) (2008) 1865–1877, <https://doi.org/10.1109/TSP.2007.913164>.
- [28] “Exploring neural ordinary differential equations for time series forecasting applications | Eki.Lab.” Accessed: Jun. 17, 2024. [Online]. Available: (https://kimerics.github.io/blog/2022/07/11/neural_ode/).
- [29] “[2401.01836] Neural Control: Concurrent system identification and control learning with neural ODE.” Accessed: Jun. 17, 2024. [Online]. Available: (<https://arxiv.labs.arxiv.org/html/2401.01836>).
- [30] “Feasibility Study of Neural ODE and DAE Modules”.
- [31] R. Dandekar *et al.*, “Bayesian Neural Ordinary Differential Equations”.
- [32] “Bayesian Neural Ordinary Differential Equations - Raj Dandekar.” Accessed: Jun. 17, 2024. [Online]. Available: (https://rajdandekar.github.io/publications/Bayesian_NeuralODE).
- [33] R. Bergna, F. Opolka, P. Liò, and J.M. Hernandez-Lobato, “Graph Neural Stochastic Differential Equations,” Aug. 2023, Accessed: Jun. 17, 2024. [Online]. Available: (<http://arxiv.org/abs/2308.12316>).
- [34] J. Antorán, S. Padhy, R. Barbano, E. Nalisnick, D. Janz, and J.M. Hernández-Lobato, “Sampling-based inference for large linear models, with application to linearised Laplace,” Oct. 2022, Accessed: Jun. 17, 2024. [Online]. Available: (<http://arxiv.org/abs/2210.04994>).
- [35] P. Hegde, Ç. Gataç Yıldız, H. Lähdesmäki, S. Kaski, and M. Heinonen, “Variational multiple shooting for Bayesian ODEs with Gaussian processes”.

- [36] T.K. Roy, M.A. Mahmud, A.M.T. Oo, M.E. Haque, K.M. Muttaqi, N. Mendis, Nonlinear adaptive backstepping controller design for islanded DC microgrids, IEEE Trans. Ind. Appl. vol. 54 (3) (May 2018) 2857–2873, <https://doi.org/10.1109/TIA.2018.2800680>.
- [37] H. Huang, P. Ju, X. Pan, Y. Jin, X. Yuan, Y. Gao, Phase-amplitude model for doubly fed induction generators, J. Mod. Power Syst. Clean. Energy vol. 7 (2) (Mar. 2019) 369–379, <https://doi.org/10.1007/s40565-018-0450-0>.
- [38] M. Demirol, R. Caglar, T.N. Demirol, Wind farm dynamic analysis in terms of Turkish grid codes, 2016 Int. Conf. Probabilistic Methods Appl. Power Syst., PMAPS 2016 - Proc., Inst. Electr. Electron. Eng. Inc. (Dec. 2016), <https://doi.org/10.1109/PMAPS.2016.7764211>.
- [39] H. X. K, G.Z. Ge, Turing: a language for flexible probabilistic inference. Proc. Twenty-First Int. Conf. Artif. Intell. Stat., <i>Proc. Mach. Learn. Res. (2018).
- [40] Mike Innes, Flux: elegant machine learning with julia, J. Open Source Softw. (2018).
- [41] Christopher Rackauckas and Mike Innes and Yingbo Ma and Jesse Bettencourt and Lyndon White and Vaibhav Dixit, “DiffEqFlux.jl - {A} Julia Library for Neural Differential Equations,” CoRR, vol. abs/1902.02376, May 2019.
- [42] K. Xu, “Probabilistic Programming in Julia New Inference Algorithms,” 2016.
- [43] R. McElreath, Statistical rethinking: a Bayesian course with examples in R and Stan, Chapman & Hall/CRC texts in statistical science series, Ideas. Repec. Org. (2016) 470.
- [44] M. Mahmood, The mean squared error of the likelihood based estimating equations in the general linear model, Model Assist. Stat. Appl. vol. 10 (3) (Jul. 2015) 221–229, <https://doi.org/10.3233/MAS-150326>.
- [45] A. Espuña Fontcuberta, “Analyzing the Negative Log-Likelihood Loss in Generative Modeling ALEIX ESPUÑA FONTCUBERTA,” 2022.
- [46] M. (2021) Taboga, “Maximum likelihood - MATLAB Example”, Lectures on probability theory and mathematical statistics. Kindle Direct Publishing,” Online appendix. (<https://www.statlect.com/fundamentals-of-statistics/maximum-likelihood-matlab-example>).
- [47] “Chapter 3 Robustness Properties of the Student t Based Pseudo Maximum Likelihood Estimator.”
- [48] N. Balakrishnan and C.D. Cutler, “Maximum Likelihood Estimation of Laplace Parameters Based on Type-II Censored Samples.”
- [49] X.-P. Wang, Y. Song, M. Irving, Modern power systems analysis, Mod. Power Syst. Anal. (2008), <https://doi.org/10.1007/978-0-387-72853-7>.
- [50] R.T.Q. Chen, Y. Rubanova, J. Bettencourt, and D. Duvenaud, “Neural Ordinary Differential Equations”.
- [51] B. Carpenter, et al., Stan: A probabilistic programming language, J. Stat. Softw. vol. 76 (1) (Jan. 2017) 1–32, <https://doi.org/10.18637/JSS.V076.I01>.
- [52] F. Dominici, J.J. Faraway, M. Tanner, J. Zidek, and P.J. Smith, “Bayesian Data Analysis CHAPMAN & HALL/CRC Texts in Statistical Science Series Series Editors Analysis of Failure and Survival Data”.
- [53] D. Koller and N. Friedman, “Probabilistic Graphical Models: Principles and Techniques”.