Parameter Estimation of Power Electronic Converters with Physics-Informed Machine Learning

Zhao, Shuai; Peng, Yingzhou; Zhang, Yi; Wang, Huai

Published in:
IEEE Transactions on Power Electronics

DOI (link to publication from Publisher):
10.1109/TPEL.2022.3176468

Publication date:
2022

Document Version
Publisher's PDF, also known as Version of record

Parameter Estimation of Power Electronic Converters With Physics-Informed Machine Learning

Shuai Zhao, *Member, IEEE*, Yingzhou Peng, *Member, IEEE*, Yi Zhang, *Member, IEEE*, and Huai Wang, *Senior Member, IEEE*

***Abstract*—Physics-informed machine learning (PIML) has been emerging as a promising tool for applications with domain knowledge and physical models. To uncover its potentials in power electronics, this article proposes a PIML-based parameter estimation method demonstrated by a case study of dc–dc Buck converter. A deep neural network and the dynamic models of the converter are seamlessly coupled. It overcomes the challenges related to training data, accuracy, and robustness which a typical data-driven approach has. This exemplary application envisions to provide a new perspective for tailoring existing machine learning tools for power electronics.**

***Index Terms*—Buck converter, deep neural network, prognostics and health management, physics-informed machine learning (PIML), condition monitoring.**

I. Introduction

IN-SITU component electrical parameters can be used for adaptive control, fault diagnosis, and condition monitoring of power electronic converters [1], [2]. One research direction is to indirectly estimate the parameters based on existing sensors and available information in the converters, without the introduced hardware interference and cost [3]. The efforts made can be categorized into *purely data-driven methods* [4]–[7] and *hybrid data-driven approaches* [8]–[11], depending on whether the system physical models are utilized or not.

The purely data-driven method is designed to learn underlying functional relationships between the easily accessible signals and component parameters to be monitored, and a vast amount of machine learning (ML) tools [12], [13] can be used to establish such functional relationships. Given the established functional relationship and easily accessible signals, the component parameters will be estimated indirectly. For example, in [4], a feed-forward neural network is applied to learn the nonlinear relationship between the capacitance of the dc-link capacitor and the harmonics of the dc-link ripple voltage. Once the neural network is well trained, the capacitance will be inferred from the harmonics of ripple voltage indirectly. In [5], to monitor the capacitor aging status in the dc filters, an adaptive neuro-fuzzy inference system algorithm is trained to establish the functional relationship between the algorithm inputs, including the converter input voltage and voltage crossing the dc filters, and the algorithm output as the capacitor aging index in the dc filters. In this way, the capacitor aging index is indirectly obtained with easily accessed voltage information. One of the key merits of the purely data-driven method is that these methods are flexible and easy to implement [3]. Nevertheless, to our best knowledge, these purely data-driven solutions have been rarely implemented in industrial fields, due to the following common challenges. First, the power electronics is not a data-intensive field to date, such as image recognition, where the data-driven method has shown great success based on solid data foundations (e.g., "ImageNet" with million samples). Instead, the data collection in power electronics is prohibitively resource-consuming with limited sampling frequency and accuracy. As a result, ML tools may not be well trained and thus the benefits cannot be fully unleashed. Second, considering the uncertainty sources (high-level heterogeneity, parameter variations, diversified mission profiles, etc.) in power electronic systems, the underlying assumption of the identical data distributions between the training and testing datasets are difficult to fulfill. Thus, the method generalization is relatively poor especially for the cases where few data are available in the training dataset. More often than not, it results in the models that are less robust for external disturbance. Third, power electronic systems are implemented for mission-critical applications in most cases, with almost zero-tolerance to unpredictable results. However, most of the existing purely data-driven solutions suffer from the "black-box" feature. The predictions are less accountable and may be subjected to physical inconsistency, which could be unacceptable for practical engineers.

One of the favorable features of power electronic systems is that there are already certain aspects of physical models and domain knowledge. It includes deterministic functional relationships of underlying system behaviors, e.g., symmetry, invariance, algebraic equations, logical rules, principles (e.g., Ohm's law, Kirchhoff circuit laws, differential equations) [14], etc. Therefore, it is natural to leverage the efficient deterministic principles and flexible data-driven approaches in a synergistic manner for complementing each other, as hybrid data-driven methods. In contrast to purely data-driven methods, the system models are partially known in advance as the prior information.

Manuscript received November 8, 2021; revised January 27, 2022 and March 30, 2022; accepted May 15, 2022. Date of publication May 20, 2022; date of current version June 24, 2022. This work was supported by the Villum Foundation through the project of Light-AI for Cognitive Power Electronics. Recommended for publication by Associate Editor A. Yazdani. *(Corresponding authors: Shuai Zhao; Yingzhou Peng.)*

The authors are with the Department of Energy, Aalborg University, 9220 Aalborg, Denmark (e-mail: szh@et.aau.dk; ype@et.aau.dk; yiz@et.aau.dk; hwa@et.aau.dk).

This article has supplementary material provided by the authors and color versions of one or more figures available at https://doi.org/10.1109/TPEL.2022.3176468.

Digital Object Identifier 10.1109/TPEL.2022.3176468

The unknown model coefficients, which are the component parameters to be monitored, will be estimated with the measurable data. By searching an optimal set of model coefficients with optimization tools, an accurate model is deemed to be established once the errors between model outputs and the physical systems are minimized. For example, in [15], the component parameter estimation is formulated as a multivariable optimization task of a state-space model of a Buck converter. The matrices of the state-space model are decomposed with eigenvectors and eigenvalues with unknown component parameters, and these parameters are optimized by using the biogeography-based optimization method. In [16], a generalized gradient descent algorithm is applied to the parameter estimation of an interleaved boost converter based on the system dynamic model. Usually, the hybrid data-driven methods are more accurate and robust, while the existing studies still suffer from the following limitations. First, in most existing studies, the data collection requires injecting a pseudorandom binary sequence into the controller [3]. Considering the controller is one of the crucial parts, its safety and reliability are the highest priority, especially for mission-critical applications. From the industrial product perspective, proactively injecting the disturbance into the controller is not favorable or even not allowed for condition monitoring purposes. Second, advanced frequency domain analysis, such as s- or z-domain transformation, is required in most cases. These analysis techniques are challenging to be scalable for large complex systems [3]. Third, due to the intrinsic nonlinearity, the model discretization techniques are necessary for simplifying the model. The inaccuracy resulting from the discrete form is more significant for complex systems and will reduce the estimation accuracy [17].

To date, the notion of physics-informed machine learning (PIML) [14], [18], [19] is gaining widespread attention. Compared to the existing ML tools, it is imperative to integrate physics models into data-driven pipelines, so as to enforce the learning process toward physically consistent solutions. It provides a new paradigm to integrate the physical models and data in the data-driven methods, with great benefits to reduce data requirements, reduce the computation burden, and augment the prediction transparency [13]. This new paradigm has achieved success in complex engineering fields and profoundly outperformed conventional methods, such as fluid mechanics [20], power systems [21], etc. Several salient tools have been developed for PIML in the literature. For example, with the automatic differentiation capability of a neural network, the physical knowledge can be explicitly incorporated to the loss of a supervised learning task during the training stage. As a result, a physics-informed neural network (PINN) [22] is proposed and implemented as an open-source tool based on Tensorflow. In [23], the infinite derivative property of Gaussian processes is exploited for developing the physics-informed Gaussian processes (PIGP). The Gaussian process priors can be modified according to the particular forms of physical models, so that the data and physical knowledge are integrated during the learning stage. Compared to PINN, the advanced uncertainty quantification feature of Gaussian processes can be also extended to the PIGP. In addition, several pioneering open-source toolboxes have been prepared to accelerate the PIML implementation, such as DeepXDE [24] based on Tensorflow, SciANN [25] based on Keras, and SimNet [26] developed by NVIDIA, which are expected to deliver great benefits to the domains with plentiful physical models and fewer data.

This article aims to deliver these benefits of PIML to the power electronic field. As an exemplary application, it proposes a new method of parameter estimation of a dc–dc Buck converter based on PIML. The Buck converter physical knowledge is seamlessly coupled into the training of a deep neural network to mitigate the common challenges in data-driven parameter estimation. The code and data accompanying this article are available in the Supplementary Material and also on GitHub at[1]. The presented study has the following contributions and advantages.

1) It proposes a new scheme to synergistically combine the data and physical models in the parameter estimation tasks for power electronic applications.
2) The implemented PIML for Buck converter requires a relatively small size of training data, more importantly, which are readily available feedback signals for control purposes.
3) The complex s- or z-domain transformation and discretization techniques are not required in the proposed method. Moreover, in principle, the method could be scalable for more complex power electronic systems with a larger number of parameters compared to the Buck converter.

The rest of this article is organized as follows. Section II presents the methodology, including the physical model of a dc–dc Buck converter, idea and framework of the PIML method, and data preparation procedure. Section III and Section IV verify the proposed method in the simulation and experiment testing, respectively. Finally, Section V concludes this article.

II. Methodology

A. System Dynamic Modeling of Buck Converter

Taking advantage of complementary strengths of physical models and data, this article proposes a new parameter estimation method based on the modern PIML tool, simultaneously exploiting the powerful universal approximation capability of deep learning, and the efficient deterministic physical models. The dc–dc Buck converter, which is a fundamental energy conversion system topology in power electronics, will be applied as an exemplary application to uncover the potentials of PIML in power electronics.

Fig. 1 shows the circuity topology of a dc–dc Buck converter. The system dynamic model of the Buck converter has been well-established in the literature as (1) [8]. In (1), $A = (S \cdot R_{dson} + R_L + R_C R/(R_C + R))$. S is the switch state of the power semiconductor device that is 1 when it is ON or 0, otherwise, i_L is the inductor current, v_C is the dc capacitor voltage, v_o is the output voltage, R is the load resistance, C is the capacitance of the dc-link capacitor, R_C is the corresponding equivalent series resistance, V_{in} is the input voltage, and V_F is

[1][Online]. Available: https://github.com/ms140429/PIML_Converter

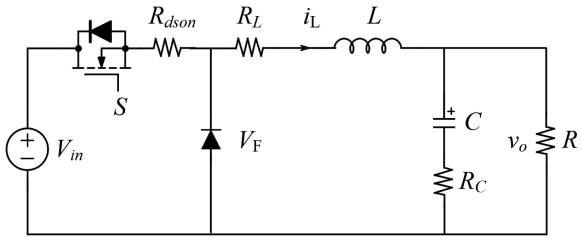


Fig. 1. Circuit topology of the dc–dc Buck converter.

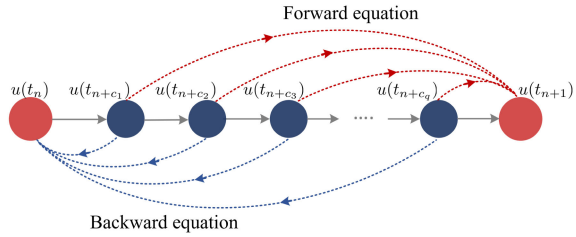


Fig. 2. Explicit relationship between the observable states $u(t_n), u(t_{n+1})$, and intermediate latent states $u(t_{n+c_i})$ through backward equation (4) and forward (7).

the diode forward voltage. In essence, the converter is operated with a controllable pulse width modulation (PWM) sequence. From a dynamic system perspective, the system behavior of the converter is deemed as periodic repetitions of multiple PWM cycles with the switch state $S = 1$ and $S = 0$. According to (1), specifically, at the beginning of each PWM cycle, the inductor current i_L and output voltage v_o will start from their initial states and then evolve toward certain ending states, given an operation time Δt with $S = 0$. Subsequently, these ending states will serve as the initial states for the rest PWM time period $T - \Delta t$, when $S = 1$. The above two steps will be repeated alongwith the converter operation.

$$\begin{bmatrix} \frac{di_L}{dt} \\ \frac{dv_C}{dt} \\ v_o \end{bmatrix} = \begin{bmatrix} -\frac{A}{L} & -\frac{1}{L}\left(\frac{R}{R_C+R}\right) \\ \frac{1}{C}\left(\frac{R}{R_C+R}\right) & -\frac{1}{C}\left(\frac{1}{R_C+R}\right) \\ \frac{R_C R}{R_C+R} & \frac{R}{R_C+R} \end{bmatrix} \times \begin{bmatrix} i_L \\ v_C \end{bmatrix} + S\begin{bmatrix} \frac{v_{\text{in}}}{L} \\ 0 \\ 0 \end{bmatrix} + (1-S)\begin{bmatrix} \frac{-V_F}{L} \\ 0 \\ 0 \end{bmatrix}. \quad (1)$$

B. Physics-Informed Machine Learning Concept

According to (1), the parameter estimation of the converter is essentially a multivariable optimization task of differential equations. Existing numerical differential equation solvers in the literature, however, are not applicable here due to the distinctive feature of condition monitoring applications, e.g., stiffness behavior in the system model, sparse and noisy data, high accuracy requirement, etc. Using a neural network for parameter estimation of differential equations goes back to the 1990s [27]. However, the power and potentials of such ideas were not unleashed due to the limited computation hardware by then. The situation has been remarkably improved to date, both in terms of the computation hardware and open-source toolbox. A growing body of literature indicates that the classical ML methods are developed to incorporate partial differential equations of dynamic systems in the data-driven solutions or discovery [22], [23]. In this article, we resort to this modern scientific computation framework for the parameter estimation of power electronic systems. Formally, consider a dynamic system with the following differential equations:

$$u_t + \mathcal{N}[u;\lambda] = 0, x \in \Omega, t \in [0,T] \quad (2)$$

and

$$f := u_t + \mathcal{N}[u;\lambda] \quad (3)$$

where $u = u(\boldsymbol{x}, t)$ denotes the solution of differential equations, $\boldsymbol{x}$ are the space coordinates, and t is the time coordinate. The abovementioned form of system dynamic model is generic and covers a wide range of tasks. Regarding (2), there are two relevant tasks as 1) *data-driven solution:* given system parameter λ, what is the solution of u; and 2) *data-driven discovery:* given the system observed data u, what are the system parameters λ that can generate such data.

For the parameter estimation task in the converter, u can be either the inductor current i_L or the output voltage v_o, and the parameter set λ are the component parameters to be monitored, including inductance L, resistance of inductor R_L, capacitance C, equivalent series resistance R_C, and ON-state resistance R_{dson} of the power semiconductor device. Moreover, other information including input voltage V_{in} and diode forward voltage V_F can be incorporated as the unknown parameters in the estimation procedure as well. As a result, it aims to estimate the component parameters given the data of the inductor current i_L and output voltage v_o, i.e., data-driven discovery.

C. Implicit Runge–Kutta Time-Stepping Scheme

Consider a general case of a dynamic system governed by (2). For simplicity, the space coordinates $\boldsymbol{x}$ are neglected in u hereafter. For data collection, suppose that the system is observed with the state $u(t_n)$ at the time t_n and the state $u(t_{n+1})$ at time t_{n+1}, where $t_{n+1} = t_n + \Delta t$. Note that the time period Δt is unnecessarily equal for any t_n. Meanwhile, as shown in Fig. 2, assume that there are q intermediate states $u(t_{n+c_i})$ during the Δt time period, where $t_{n+c_i} = t_n + c_i \Delta t$, $c_i \in [0,1], i = 1, \ldots, q$. In contrast to the observable states $u(t_n)$ and $u(t_{n+1})$, note that these intermediate states are unobservable and latent. According to [22], the observable states $u(t_n)$ and $u(t_{n+1})$ and the intermediate states $u(t_{n+c_i})$ can be explicitly coupled in the framework of the implicit Runge–Kutta method with $q-$stages [28]. Formally, their functional relationship can

be derived as

$$u_i(t_n) = u(t_{n+c_i}) + \Delta t \sum_{j=1}^{q} a_{ij}\mathcal{N}[u(t_{n+c_j}); \lambda] \quad (4)$$

$$u_i(t_{n+1}) = u(t_n) - \Delta t \sum_{j=1}^{q} b_j\mathcal{N}[u(t_{n+c_j}); \lambda] \quad (5)$$

where $i, j = 1, \ldots, q$, $u_i(t_n) = u(t_n)$, and $u_i(t_{n+1}) = u(t_{n+1})$. The parameter set $\{a_{ij}, b_j, c_j\}$ can be explicitly calculated given a certain step number q in the implicit Runge–Kutta method. Specifically, it is generally provided as a Butcher tableau [28] with the following form

$$\begin{array}{c|c} \boldsymbol{c} & \boldsymbol{A} \\ \hline & \boldsymbol{b}^T \end{array} = \begin{array}{c|cccc} c_1 & a_{11} & a_{12} & \ldots & a_{1q} \\ c_2 & a_{21} & a_{22} & \ldots & a_{2q} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ c_q & a_{q1} & a_{q2} & \ldots & a_{qq} \\ \hline & b_1 & b_2 & \ldots & b_q \end{array} \quad (6)$$

where $\boldsymbol{A} = (a_{ij})_{i,j=1,2,\ldots,q} \in \mathbb{R}^{q\times q}$, $\boldsymbol{b}, \boldsymbol{c} \in \mathbb{R}^{q\times 1}$. By incorporating (4) into (5), the relationship between the ending states $u(t_{n+1})$ and the intermediate states $u(t_{n+c_i})$ can be derived as

$$u_i(t_{n+1}) = u(t_{n+c_i}) + \Delta t \sum_{j=1}^{q} (a_{ij} - b_j)\mathcal{N}[u(t_{n+c_j}); \lambda]. \quad (7)$$

Therefore, the relationship between $u(t_n)$ and the intermediate latent states $u(t_{n+c_i})$ is governed by the backward equation (4), and such a relationship in terms of $u(t_{n+1})$ is governed by the forward equation (7). For illustration, the functional relationship between the observable states and the intermediate latent states is shown in Fig. 2. In this way, $u(t_n)$ and $u(t_{n+1})$ are correlated together through depending on the intermediate latent states $u(t_{n+c_i})$. Notably, the parameter λ is involved in this constructed functional relationship.

Likewise, the abovementioned idea is applied to characterize the dynamics of the inductor current and output voltage of the Buck converter in this article. The training data are selected as the peak values of the inductor current and output voltage when the transient behaviors occur, as these data are more informative and sensitive to the converter parameter variations. As it will be seen later in Section II-E, such a data collection mechanism is efficient and of the data-light feature. Take the inductor current as an example. Specifically, consider that the inductor current is only available at the peak points, i.e., the lower peak current $i_L(t_n)$ and upper peak current $i_L(t_{n+1})$. The time between the peak points is denoted as Δt, i.e., $t_{n+1} = t_n + \Delta t$. For the q-steps implicit Runge–Kutta time-stepping scheme, suppose that the intermediate latent states between the peaks of the inductor current are $i_L(t_{n+c_k})$, $t_{n+c_k} = t_n + c_k\Delta t$, $k = 1, \ldots, q$, $c_k \in [0, 1]$. From a dynamic system perspective, given the initial state of the inductor current $i_L(t_n)$, the converter is expected to cross each intermediate states $i_L(t_{n+c_k})$, respectively, and then reach the ending state $i_L(t_{n+1})$ during Δt. For illustration, the state evolution mechanism of the inductor current is shown in Fig. 3. The abovementioned analysis applies to the output voltage v_o as well.

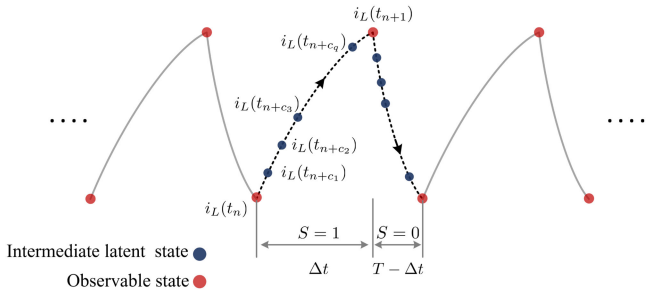


Fig. 3. Intermediate latent states of inductor current i_L in the scheme of implicit Runge–Kutta time-stepping scheme, where $t_{n+c_k} = t_n + c_k\Delta t$, $k = 1, \ldots, q$. This setting also applies to the output voltage v_o.

Collectively, the intermediate latent states for the converter are $[i_L(t_{n+c_1}), \ldots, i_L(t_{n+c_q}), v_o(t_{n+c_1}), \ldots, v_o(t_{n+c_q})]$. The system dynamic model (1) can be further derived according to the standard form (2). Specifically, for the inductor current i_L

$$\frac{di_L}{dt} + \mathcal{N}[i_L; \boldsymbol{\theta}] = 0 \quad (8)$$

$$\mathcal{N}[i_L; \boldsymbol{\theta}] = \frac{\begin{bmatrix} (S \cdot R_{dson} + R_L)i_L + v_o \\ -S \cdot V_{\text{in}} + (1 - S) \cdot V_F \end{bmatrix}}{L} \quad (9)$$

and for the output voltage v_o

$$\frac{dv_o}{dt} + \mathcal{N}[v_o; \boldsymbol{\theta}] = 0 \quad (10)$$

$$\mathcal{N}[v_o; \boldsymbol{\theta}] = \frac{v_o + C \cdot R_C \cdot R \cdot \mathcal{N}[i_L; \boldsymbol{\theta}] - R \cdot i_L}{C \cdot (R_c + R)} \quad (11)$$

where $\boldsymbol{\theta} = \{L, R_L, C, R_C, R_{dson}, V_{\text{in}}, V_F, R\}$ are the unknown component/system parameters to be estimated. In this way, the system dynamics of the Buck converter are derived in the framework of the implicit Runge–Kutta method. The relationships of the initial and ending states of the inductor current/output voltage are explicitly constructed, which is significant for coupling the data and the physics model. Given the data of initial and ending states of i_L and v_o, how to approximate the intermediate latent states is a crucial step for the proposed method, which will be detailed subsequently.

D. Physics-Informed Neural Network for Buck Converter

Deep learning is a powerful tool to approximate any nonlinear relationship between the model inputs and outputs. By exploiting the universal approximation capability, a deep neural network is applied here to construct and predict the intermediate latent states, given the information of the initial and ending states of the inductor current and output voltage. These approximated intermediate states will be utilized as the prior information for the subsequent physical models (8) and (10) [22].

Fig. 4 shows the configuration of the PINN for the dc–dc Buck converter. It consists of two parts, i.e., the data-driven part and physical model part. For the data-driven part, the inputs of the deep neural network include the initial states of the inductor current $i_L(t_n)$ and output voltage $v_o(t_n)$, the switch state S, and the period of time Δt from the initial states to the

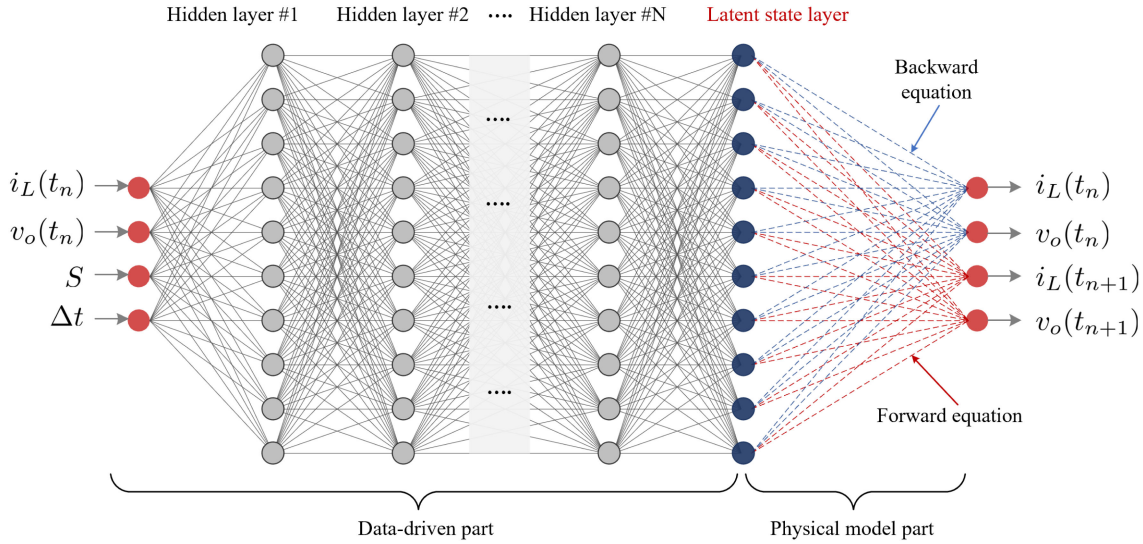


Fig. 4. Structure of the PINN for the dc–dc Buck converter. The intermediate latent state layer is coupled with the initial states $i_L(t_n)$ and $v_o(t_n)$ and ending states $i_L(t_{n+1})$ and $v_o(t_{n+1})$ through backward equation (4) and forward equation (7), where $t_{n+1} = t_n + \Delta t$.

ending states of the inductor current and output voltage. The deep neural network can be flexibly configured with a different number of hidden layers and neurons in each layer for sufficient network expressivity. The output of the data-driven part is the intermediate latent states of inductor current and the output voltage $[i_L(t_{n+c_1}), \ldots, i_L(t_{n+c_q}), v_o(t_{n+c_1}), \ldots, v_o(t_{n+c_q})]$, which are unobservable and used as the inputs for the physical model part. The physical model part is configured according to the backward equation (4) and forward equation (7), which are embodied with the dynamic model of the Buck converter in (8) and (10). The outputs of the physical part are the initial and ending states of the inductor current and output voltage. It is worth mentioning that connections between the hidden layers in the data-driven part are determined by the numerical weights $\mathbf{w}$ of a deep neural network. In contrast, the connections between the latent state layer and the final outputs are facilitated with the backward equation (4) and forward equation (7) with the physical implications of converter dynamics (8) and (10). The data-driven part and physical part are, therefore, seamlessly coupled as a PINN for the Buck converter. Essentially, this configuration learns the dynamic behavior of the Buck converter system in a period of time Δt and $T - \Delta t$ for the switching states $S = 0$ and $S = 1$, respectively, through a supervised learning task.

In this structure, the unknown component parameters $\boldsymbol{\theta}$ to be estimated in the physical model are combined with the weights $\mathbf{w}$ and biases $\mathbf{b}$ in the data-driven part. Notably, the physical part involved with (8) and (10) is explicit and differentiable. As a result, the supervised learning error at the final output can be propagated through the physical-model part and the data-driven part, respectively, for updating the learning gradients. Therefore, the chain rule of the typical back-propagation training scheme can be readily extended to incorporate the physical part, so that all the parameters in the PINN can be simultaneously trained through a standard back-propagation method. Note that there are usually up to thousands of parameters in $\{\mathbf{w}, \mathbf{b}\}$ in a typical deep neural network. Therefore, it is worth mentioning that in this configuration, there is almost no limitation on the number of the parameters in the physical model, which indicates the method scalability. As a result, the parameter estimation of the dc–dc converter can be completed through the neural network training. Specifically, the error function of the PINN is formulated as

$$E(\boldsymbol{\Theta}) = \sum_n \left[(i_L(t_n) - \hat{i}_L(t_n))^2 + (i_L(t_{n+1}) - \hat{i}_L(t_{n+1}))^2\right] + \sum_n \left[(v_o(t_n) - \hat{v}_o(t_n))^2 + (v_o(t_{n+1}) - \hat{v}_o(t_{n+1}))^2\right] \tag{12}$$

where the parameter set of the PINN is $\boldsymbol{\Theta} = \{\mathbf{w}, \mathbf{b}, \boldsymbol{\theta}\}$, $\{\mathbf{w}, \mathbf{b}\}$ are the parameters of the data-driven part of the deep neural network, and $\boldsymbol{\theta}$ are the parameters of the physical model of the Buck converter. This hybrid design can seamlessly combine the data of the physics model of the Buck converter, modulating the training phase efficiently so that it is enforced to the underlying physical principles and constraints of the converter.

E. Configuration of Data Acquisition and Sharing

It is noticeable that the PINN is designed to learn the dynamic behavior of the Buck converter in the period of time Δt. The selection of the Δt is particularly important considering the implementation constraints in field applications, e.g., data access feasibility, sampling frequency, reliable sampling, noise, etc. To this end, a peak-to-peak sampling mechanism is proposed in this article to enable efficient data acquisition. Fig. 5 shows the proposed peak-to-peak sampling mechanism. In this setting, each collected peak data point plays two roles and can be shared between the backward and forward equations. Specifically, each low-peak data point can be utilized for either the initial state of the converter when $S = 1$ or the ending state when $S = 0$.

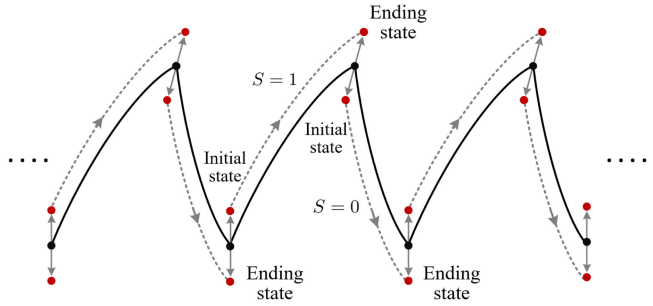


Fig. 5. Configuration of the data acquisition and data sharing.

TABLE I
SPECIFICATIONS OF THE DC–DC BUCK CONVERTER IN THE SIMULATION

V_{in}	V_{ref}	f_{sw}	C/R_C
48 V	24 V	20 kHz	164.5 μF / 0.201 Ω
L/R_L	R_{dson}	V_F	$R_1/R_2/R_3$
725 μH / 0.314 Ω	0.221 Ω	1 V	10.2 / 3.1 / 6.1 Ω

Likewise, the high-peak data point is either the ending state of the converter when $S = 1$ or the initial state when $S = 0$. One of the major benefits of such peak-to-peak sampling method is that it will significantly reduce the data requirements for the training, i.e., the sampling frequency will be significantly reduced to $2f_{\text{sw}}$, which is low and can be readily implemented on typical power electronic systems. Meanwhile, the control signal, i.e., the switch state $S = 0$ or $S = 1$, can be implicitly identified from the peak values and therefore it is unnecessary to collect the control signals.

III. Accuracy and Robustness Analysis

To verify the accuracy and robustness of the proposed method, a dc–dc Buck converter is implemented in MATLAB to generate the simulation data. The specifications of the Buck converter are given in Table I. Notably, the converter simulation for the data generation is very flexible, where various external disturbance factors in practice can be manually injected for testing accuracy and robustness, such as the analog-to-digital converter (ADC) quantization error, noise immunity capability, and synchronization error between the inductor current and output voltage. Thus, in addition to the clean data generated by the converter directly, the abovementioned three aspects can be comprehensively tested for method verification.

A. Method Accuracy and Robustness

For the training data, the peak-to-peak values of the inductor current and output voltage when transient behaviors occur are collected. The converter transient behaviors may be due to the converter start-up, load-changing, etc. Notably, these typical operational modes are common and such data collection scheme is readily applicable to industrial implementations. Taking the traction inverter in the railway application as an example, the data of the transient states of the train start-up, accelerating, decelerating, etc., are easy to obtain. Note that the collecting of peak-to-peak values would be challenging if an appropriate controller is designed to suppress the overshoot and the following oscillations. In this case, the time period of the transient behavior will be shorter, suggesting that the number of peak-to-peak points of the ripple signals would be small. While the degradation of power electronic devices is usually very slow (e.g., months), the data collection procedure can wait for a longer time for capturing more transient behaviors due to the load changes. It will have sufficient time to prepare the dataset, especially considering the data-light feature of the proposed method. Moreover, it is noted that the data collection is not only limited to the transient behaviors due to the load changes. Other factors (e.g., the variation of input voltage V_{in} and system startup/shutdown) that can make the converter change from the stable state to transient state can be exploited as well.

In this simulation case, the peak values of the inductor current and output voltage of the transient state due to arbitrary load changes (e.g., $R_0 \to R_1$, $R_1 \to R_2$, $R_2 \to R_3$) are used. For each transient case, the peak-to-peak values of 120 switching periods are collected from the time of the load changing to that of the converter running into the steady state. As a result, for the three load-changing cases, the dataset $\mathcal{D} = \{i_L(t_n), i_L(t_{n+1}), v_o(t_n), v_o(t_{n+1}), \Delta t\}_n$ is consisting of peak-to-peak samples in 360 periods, which is a small dataset for deep neural network training and indicates the data-light feature of the proposed method. It is worth mentioning that the data volume of $\mathcal{D}$ is constant for the parameter estimation task at any condition monitoring times. It will not be progressively increasing alongwith the long-term system operation.

The PINN-based parameter estimation method is implemented by using Python with Tensorflow backend. The deep neural network is configured as five layers with 50 neurons in each layer, and such a structure selection will be justified later in Section III-B. The network is trained by using the Adam optimizer followed by a full-batch L-BFGS optimizer [22], with the default parameter settings (the learning rate $\eta = 0.001$, and exponential decay rates $\beta_1 = 0.9$ and $\beta_2 = 0.999$). The epoch of the Adam optimizer is set as 200 000. Based on the fact that the theoretical truncation error of the implicit Runge–Kutta method is $\mathcal{O}(\Delta t^{2q})$ [28], the q of the time-stepping scheme is elaborately selected to suppress the truncation error below the machine accuracy. It can be determined as

$$q = 0.5 \log(\epsilon) / \log(\Delta t). \tag{13}$$

In this case, the largest Δt will be the period of the PWM signal, i.e., 50 μs. For a 64-bit operation system, the machine precision is $\epsilon = 2.2e - 16$. To mitigate the truncation error, the q should be at least $2 > 1.82$. In the following testing, q is set as 20 to suppress the truncation error factor and ensure the data-driven structure robustness as well. This setting also aims to suppress the effect of q in the following analysis to a large extent. In addition to the directly simulated data, three disturbance aspects are verified based on emulating the data characteristics in the field operation, including the ADC quantization error, noise immunity capability, and synchronization error.

1) *ADC quantization error:* One of the key differences between the simulation and the experimental data is the

TABLE II
PERCENTAGE ERROR (%) OF COMPONENT PARAMETERS IN DIFFERENT TESTING CASES

Error (%) / Cases (mean, %)	L	R_L	C	R_C	R_{dson}	R_D	R_1	R_2	R_3	V_{in}	V_F
Clean data (0.1)	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
ADC error (0.1)	0.1	0.2	0.1	0.1	0.5	0.1	0.1	0.1	0.1	0.1	0.2
Sync error (1.6)	0.4	0.5	0.1	5.7	0.1	0.3	0.1	0.1	0.1	0.2	8.8
5 noise (0.8)	0.1	0.6	0.1	2.8	0.8	0.1	0.1	0.1	0.1	0.1	3.4
10 noise (2.1)	0.3	1.1	0.7	5.5	0.6	0.4	0.1	0.3	0.2	0.3	12.3
ADC-Sync-5noise (3.6)	0.8	5.9	1.0	5.2	11.0	1.1	0.1	0.1	0.1	0.3	11.4
ADC-Sync-10noise (4.9)	1.0	13.0	1.1	4.4	27.3	3.6	0.1	0.2	0.2	0.1	1.6

(ADC error: ADC quantization error; Sync error: Synchronization error). The data-driven part is the deep neural network configured as five layers with 50 neurons in each layer. All the results below 0.1% are rounding up to 0.1% for clarity.

quantization error when using ADC for the data sampling. Thus, the simulation data are further processed with a 12-bit ADC quantization to emulate the sampling procedure in practice. Considering the signal scope of i_L and v_o, the quantization error for the inductor current is $10/(2^{12}-1) = 2.4$ mA and that of the output voltage is $30/(2^{12}-1) = 7.3$ mV.

2) *Noise immunity capability:* This affecting factor is tested by adding the random Gaussian noise in i_L and v_o. Specifically, the standard derivation σ for 1 time noise in the inductor current i_L is set as the ADC precision $10/(2^{12}-1) = 2.4$ mA and that of the output voltage v_o is set as $30/(2^{12}-1) = 7.3$ mV. According to the 3σ-rule of the Gaussian random variable, for one time noise case, the approximated scope of current noise is around $[-7.2, 7.2]$ mA, and that of the voltage noise is around $[-21.9, 21.9]$ mV. Likewise, multiple levels of noisy data are generated, including five times (the noise scale of the i_L is $[-36, 36]$ mA and that of the v_0 is $[-109.5, 109.5]$ mV) and 10 times (the noise scale of the i_L is $[-72, 72]$ mA and that of the v_0 is $[-219, 219]$ mV). Considering the ripple scope in the i_L is (0.49, 0.97) A and that of the v_o is (0.0021, 0.42) V, the abovementioned settings are applicable in practice.

3) *Synchronization error between i_L and v_o:* In practice, the sampling hardware may not be able to sample the peaks of i_L and v_o, simultaneously, as designed. There may be a synchronization error when sampling these two signals. Considering the period of the PWM signal 50 μs, a random synchronization error is introduced to emulate this disturbance factor. Specifically, a uniform random variable with scope [0, 2] us is added between the sampling time of i_L and v_o for simulating this scenario.

With the abovementioned experimental settings, the training of the PINN is completed in around 15 min on a regular computation platform (Intel Xeon CPU E5-2620, 2.4 GHz). It is worth mentioning that the proposed method is designed for the degradation condition monitoring. It is expected to be implemented on cloud-edge-based computation platforms. For example, the edge platform will send the data (e.g., the inductor current i_L and the output voltage v_o) to the cloud platform. Afterward, the proposed method will be executed on the powerful cloud platform for the parameter estimation. Since the degradation in power electronic systems is very slow (e.g., months) in practice, the execution time on the cloud platform is not quite critical. In the future, the computation efficiency can be elaborately optimized for real-time applications (e.g., adaptive control design).

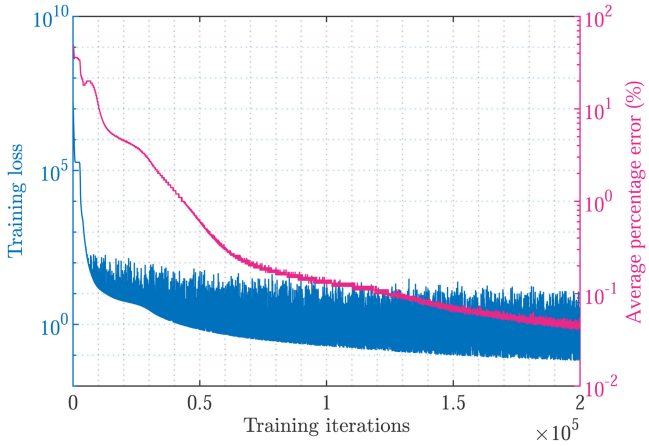


Fig. 6. Convergence of the training process of the PINN and the average percentage error of parameter estimation.

Various combinations of the disturbance factors are tested and results are listed in Table II, with the average of all parameter estimations in each testing case. For illustration, Fig. 6 shows the convergence process of the clean data case. Note that the training loss and the average percentage error are shown in the logarithmic scale. It can be seen that the average percentage error progressively decreases with the decreasing of the training error. In the end, the training error variation is stable and the average estimation error is within 0.1%. Several salient facts are identified as follows. First, for the clean data case without any disturbance factors, the proposed method is very accurate and the estimated percentage errors are below 0.1% for all the parameters, while based on a limited training dataset of only 360 samples. It suggests that incorporating physical knowledge into the typical ML pipeline can significantly reduce the requirements on data volume. Second, compared to the clean data case, it is noted that the factor of the ADC quantization error is almost negligible. The estimation accuracy of the R_C and V_F is relatively vulnerable to the synchronization error and the noise factors, while they are still accurate for condition monitoring applications. Third, the R_L and R_{dson} are relatively vulnerable to the composite disturbance factors, as shown in the last two testing cases. While compared to the individual R_L and R_{dson}, it can be seen that the sum of R_L and R_{dson} is more accurate

TABLE III
AVERAGE PERCENTAGE ERROR (%) OF ALL PARAMETERS FOR THE CLEAN DATA TESTING CASE IN TERMS OF DIFFERENT LAYERS AND NUMBER OF NEURONS IN THE DATA-DRIVEN PART

Layers \ Neurons	10	30	50	70	90	110	130
1	5.4	2.5	2.4	0.5	0.4	2.1	2.1
3	3.8	0.2	0.1	0.1	0.1	0.1	0.1
5	30.6	0.1	0.1	0.1	0.1	0.1	0.1
7	4.1	0.1	0.1	0.1	0.1	0.1	0.1
9	1.4	0.1	0.1	0.1	0.1	0.1	0.1

All the results below 0.1% are rounding up to 0.1% for clarity.

TABLE IV
AVERAGE PERCENTAGE ERROR (%) OF ALL PARAMETERS FOR THE POOR DATA TESTING CASE

Layers \ Neurons	10	30	50	70	90	110	130
1	9.2	8.4	8.1	7.7	8.2	7.9	6.2
3	9.4	6.9	5.2	5.2	4.8	5.0	4.9
5	37.7	5.0	4.9	5.1	5.0	4.8	5.0
7	8.6	4.9	4.7	5.0	4.8	4.8	4.8
9	9.9	5.4	5.0	4.9	4.9	4.7	4.8

(ADC quantization error, synchronization error, and 10 times noise data) in terms of different layers and number of neurons in the data-driven part.

and stable in terms of different affecting factors. As a result, the indicator $R_D = R_L + R_{dson}$ is chosen for the condition monitoring of the inductor and power semiconductor device together, which is aligned with the fact found in [8] as well. As a whole, from the perspective of the average percentage error, the maximum average error is less than 5% in all testing cases, indicating that the proposed method is accurate and robust for the parameter estimation task of the dc–dc Buck converter.

B. Optimal Network Structure Design for Simulation

In addition to the disturbance factors in the data, the accuracy regarding the structure of the deep neural network is investigated as well. Theoretically, the neural network in the data-driven part should be sufficiently expressive for modeling the complex intermediate states. Thus, it is necessary to test the network structure in terms of the different number of hidden layers and neurons in each layer. In the testing, the number of layers and neurons is increased gradually to gain more expressivity. The method is tested based on the clean data and the poor data (ADC quantization error, Synchronization error, and 10 times noise) in Table II, and the estimation results are presented in Tables III and IV, respectively. The averaged percentage error of these parameters is used to quantify the neural network performance. For the clean data testing case, it is noted that when the neuron size is larger than 30 and the number of layers is larger than three, the average percentage error is almost stable and is below 0.2%, which indicates that the network expressivity is sufficient for this parameter estimation task. For rest cases with smaller neuron sizes or fewer hidden layers, the estimated average percentage error is higher and inconsistent due to the limited expressivity of the data-driven part. The abovementioned analysis also applies to the case of the poor data testing case, where the average

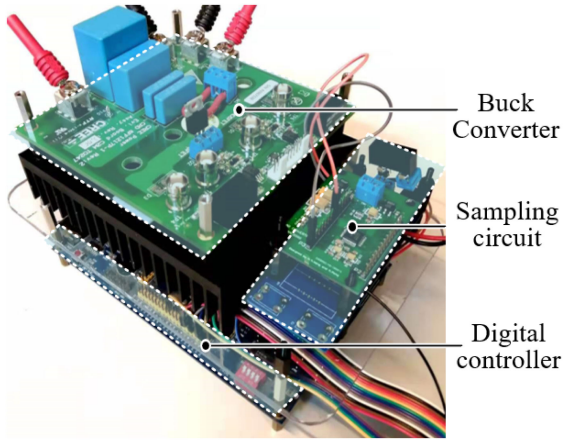


Fig. 7. Testing hardware of the dc–dc Buck converter [8].

percentage error is almost stable at around 7% when the neuron size is larger than 30 and the number of layers is larger than three. As a result, considering the computation efficiency and accuracy, the deep neural network in the data-driven part is chosen as five layers with 50 neurons in each layer for the abovementioned simulation testing in Section III-A.

As an applicable design rule, it can start with a network structure with the number of neurons and number of layers to a large extent, e.g., 9×130. Afterward, progressively reducing the number of neurons and number of layers to check the stability of the estimation accuracy, as in Table III. Once a rough lowest boundary is determined (e.g., 3×30), a slightly larger network structure (e.g., 5×50) can be applied as the final network design.

IV. EXPERIMENTAL VERIFICATION

Fig. 7 shows the Buck converter prototype for the case study. This testing platform is based on a previous work [8], where the capacitor and power semiconductor device can be easily replaced to emulate the parameter shifts. The switching frequency f_{sw} is 20 kHz. The training dataset is collected by a 12-bit oscilloscope (HDO4054 A). In each switching cycle, only the peak-to-peak values of the inductor current and output voltage are sampled for the training dataset.

A. Experimental Setting

Multiple testing groups under a variety of operating conditions and component combinations are investigated. Note that the capacitor and power semiconductor device are the most vulnerable components in power electronic systems. In the hardware testing, the degradation behavior of the converter is simulated by manually changing power semiconductors (M1: 0.225 Ω, M2: 0.152 Ω, M3: 0.072 Ω) and capacitors (C1: 164.5 μF, 0.201 Ω, C2: 160.7 μF, 0.217 Ω, C3:156.8 μF, 0.234 Ω, C4:152.9 μF, 0.253 Ω), forming different testing groups (e.g., C1M1). Other component/system specifications are the same as those used in the simulation as in Table I. Note that these parameters are measured offline. For example, the inductance

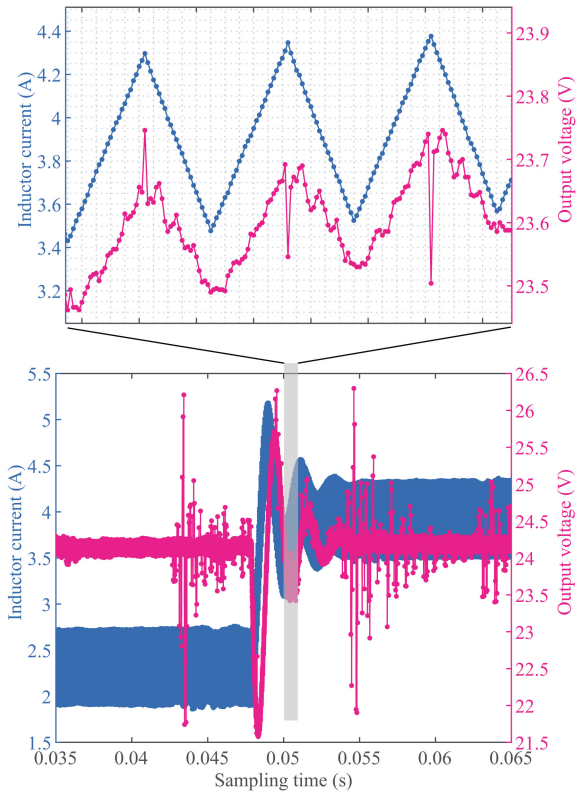


Fig. 8. Data acquisition of the inductor current and output voltage signals.

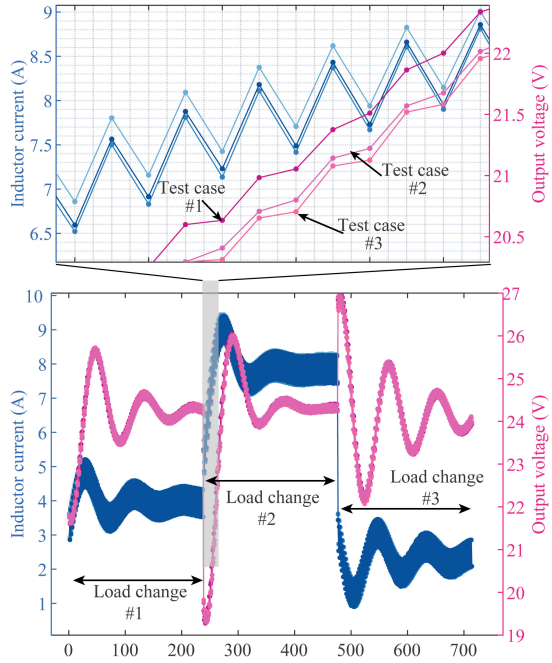


Fig. 9. Data collections of i_L and v_o when the converter transient behaviors occur due to the load changing. The data collection in each testing group is independently repeated three times for repeatability verification.

is measured as 7.25 μH and R_L is 0.314 Ω at the switching frequency with the LCR meter.

B. Data Acquisition and Preprocessing

During the normal operation, the load is manually changed to obtain the transient signals of the inductor current and output voltage. For illustration, the collected inductor current and output voltage are shown in Fig. 8. In contrast to the simulation data, it is obvious that there is a high-level noise in the hardware testing data, although the oscilloscope is used for data sampling. These transient signals are collected when manually and arbitrarily changing the load three times, e.g., $R_0 - R_1$, $R_1 - R_2$, and $R_2 - R_3$, from the occurrence of the transient behavior until the converter runs to the stable state.

As mentioned, only the peaks of the inductor current and output voltage are used. Each transient signal is collected with 120 peak-to-peak pairs, as a total of 360 data pairs are prepared for the training dataset. In the testing, the capacitors and power semiconductor devices are manually replaced for emulating the component degradation behavior. Moreover, two different power rating conditions (i.e., 48–24 V and 24–12 V) are also applied for testing the operational rating factors. The different combinations of the components and power rating form multiple testing groups. Each testing group consists of three testing cases, where the data collection procedure will be repeated three times, so that the data of the inductor current and output voltage are independently collected by multiple times. It will be applied to verify the method's repeatability. For illustration, the data of the three testing cases at one testing group are given in Fig. 9. As expected, it is noted that there is observable data discrepancy between the three testing cases. As it will be shown later, such difference will only affect the estimation accuracy to a very limited extent, which indicates the good repeatability and robustness of the proposed method.

C. Estimation Result Analysis

Multiple testing groups in terms of various component combinations and power ratings are set for testing the accuracy and robustness, as in Table V. For each testing group, note that the parameter estimation process is independently repeated based on the three different training data samples. For the hardware testing, the deep neural network in the data-driven part is set as five layers with 50 neurons in each layer, based on the optimal network structure searching results later in Section IV-D.

Before the experiment, the component parameters are measured with the LCR meter offline. Considering the variation factors between the offline static testing and the field operational stage online, there are errors and discrepancies between the LCR measured results and the component parameters in the operational converter. To suppress this factor, it is more reasonable to use the percentage variation to illustrate the estimation accuracy. Take the case of 48/24 V as an example. The testing group of C1M1 is considered as a benchmark standard and its parameter variations of both the LCR measured values and the estimated values with the proposed method, are deemed as 0%. Meanwhile, their normalized values are deemed as 100%. Subsequently, the rest testing groups are compared with the C1M1 group to calculate their parameter variations and normalized

TABLE V
PERCENTAGE VARIATIONS OF THE ESTIMATED PARAMETERS AT MULTIPLE EXPERIMENTAL SETTINGS UNDER POWER RATINGS OF 48/24 V AND 24/12 V (C: μF, R_C: Ω, R_D: Ω). THE LCR VALUES ARE CALCULATED WITH LCR METER

48/24V	C1M1			C1M2			C1M3			C2M3			C3M3			C4M3		
	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D
LCR Value	164.5	0.201	0.539	164.5	0.201	0.466	164.5	0.201	0.386	160.7	0.217	0.386	156.8	0.234	0.386	152.9	0.253	0.386
Variation (%)	0	0	0	0	0	12.9	0	0	28.0	2.3	8.0	28.0	4.7	16.4	28.0	7.0	25.9	28.0
Est. variation (%)	0	0	0	0.2	0.1	15.5	0.2	1.4	53.3	2.6	11.0	39.0	5.6	22.7	32.3	7.1	32.9	46.5

24/12V	C1M3			C2M3			C3M3			C4M3			C4M2			C4M1		
	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D	C	R_C	R_D
LCR Value	164.5	0.201	0.386	160.7	0.217	0.386	156.8	0.234	0.386	152.9	0.253	0.386	152.9	0.253	0.466	152.9	0.253	0.539
Variation (%)	0	0	0	0.3	8.0	0	4.7	16.4	0	7.0	25.9	0	7.0	25.9	20.7	7.0	25.9	39.6
Est. variation (%)	0	0	0	1.9	5.1	20.3	4.0	13.0	1.9	7.3	16.5	13.1	7.3	16.5	11.3	6.0	14.4	33.0

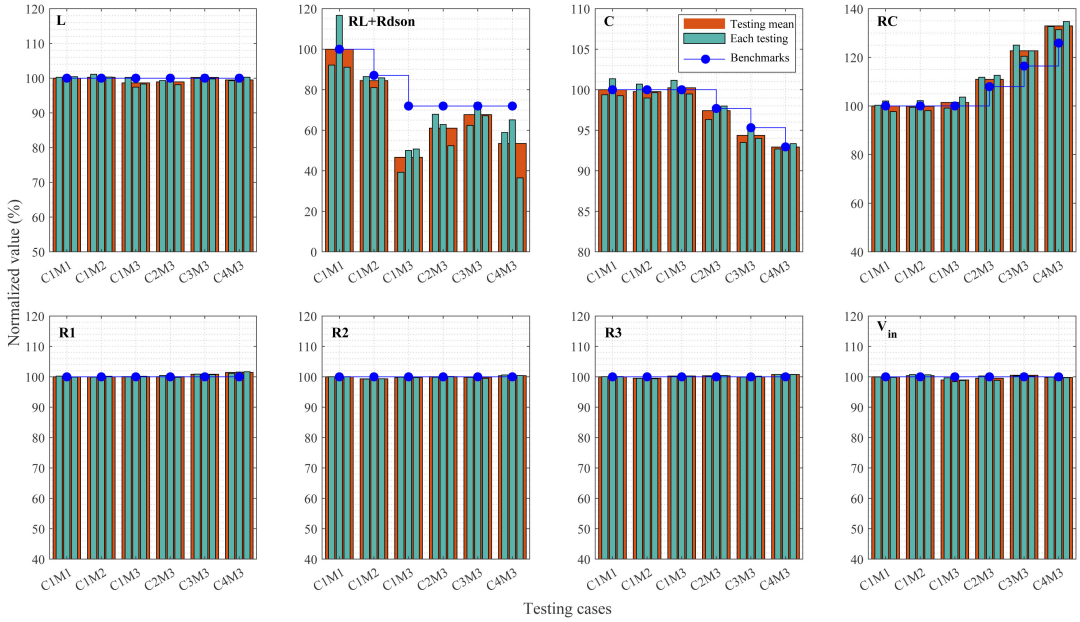


Fig. 10. Normalized value of the estimated parameters for the dc–dc Buck converter under 48–24 V rating case. R_1, R_2, and R_3 are the loads used for generating transient signals. The results of each testing group are normalized in terms of the benchmark group C1M1.

values, both for the cases of the LCR measured values and the estimated results. The corresponding percentage variations and the normalized values are then compared to validate the method's effectiveness. The results are presented in Table V, and the normalized value of the 48–24 V case is shown in Fig. 10 for illustration.

The comparison results show that the proposed method is robust and accurate for monitoring the parameter variations. Several key facts can be identified from the testing experiment. First, based on Table V, it can be seen that the estimation percentage variations are accurate and close to the benchmark values, especially for the C and R_C, for both the power rating conditions 48/24 V and 24/12 V. Second, it is found that the estimated variations of the input voltage V_{in} and the loads $R_1, R_2,$ and R_3 are very accurate, as shown in Fig. 10. Third, in each testing group, the proposed method can provide almost identical results with good repeatability in each testing case, which can be obviously shown in Fig. 10. Compared to the previous studies [8], it is evident that the proposed method is more accurate and robust, while based on only a limited dataset with fewer sensor input signals (e.g., the input voltage V_{in} and load R are not required).

It is noted that the estimation of R_D is not quite accurate compared to other estimated system parameters (e.g., input voltage V_{in} and the loads $R_1, R_2,$ and R_3). For example, in the testing group C4M3 at 48–24 V, the estimated variation of R_D is 46.5%, which is not quite close to the benchmark variation 28.0%. One of the major reasons is that the two components (e.g., R_{dson} and R_L) are only applied as the constant unknown variables in the model. Nevertheless, in the field operation, R_{dson} is highly affected by the operational conditions, e.g., temperature, current, etc., and those factors have not been considered in the converter physical model (1). Moreover, the R_L may be varied with different inductor current i_L, which is neglected in the model as well. In addition, from the converter-level perspective, there are another two error sources. On the one hand, the physical model is designed to closely characterize the converter dynamics. However, there may be some discrepancies between the physical model and real converter due to the neglected high-order parasitic factors. On the other hand, the data acquisition

TABLE VI
AVERAGE PERCENTAGE ERROR (%) OF ALL PARAMETERS IN THE HARDWARE TESTING IN TERMS OF DIFFERENT LAYERS AND NUMBER OF NEURONS IN THE DATA-DRIVEN PART

Layers \ Neurons	10	30	50	70	90	110	130
1	14.7	13.0	13.1	13.2	13.1	13.5	13.4
3	12.7	10.5	10.6	10.7	10.7	10.7	10.6
5	36.8	10.7	10.7	10.7	10.7	10.7	10.7
7	14.6	10.0	10.7	10.7	10.7	10.7	10.7
9	17.8	10.7	10.7	10.7	10.7	10.7	10.7

process will inevitably include external disturbance noise, ADC quantization error, synchronization error, etc., which will reduce the estimation accuracy as well.

D. Optimal Network Structure Design for Hardware Testing

Similar to Section III-B, based on the hardware testing data, the robustness in terms of the neural network structure is tested by gradually increasing the hidden layer number and the size of neurons. The testing is based on the testing data of group C1M1. It is worth mentioning that the average percentage error in terms of the LCR meter measured values are used to quantify the performance. Thus, the estimated percentage errors will be higher than that in Table IV due to the uncertainty sources discussed in Section IV-C. The results are given in Table VI. It is noted that when the neuron size is larger than 30 and the number of layers is larger than three, the average percentage error is almost stable and is below 11%, which indicates that the network expressivity is sufficient for this task. As a result, a 5-layers deep neural network with 50 neurons in each layer is used for the hardware testing data.

E. Discussions and Potentials

The proposed method can provide highly accurate and robust estimation results of the dc–dc Buck converter. As an exemplary application, the PIML tool has shown great benefits in the condition monitoring of a dc–dc Buck converter. It is worth mentioning that this tool can be extended to other applications involved with plentiful physical models, e.g., model predictive control, electromagnetic interference design, etc. Although it has shown promising results, yet there are still some open problems to be answered in future works.

1) How to determine the optimal weighting hyperparameters between terms in the loss function (12).
2) Why there is no suboptimal solution regarding the physical part.
3) Is there any formal design guideline of the deep neural network and what is the lowest network configuration in the data-driven part.
4) Is there any other more informative signal pattern for the parameter estimation, in addition to the peak-to-peak of the transient behaviors.
5) How to further reduce the data volume and finding a theoretical limitation on the number of transient signals when preparing the training data. More directions are open to explore.

V. CONCLUSION

This article proposes a new parameter estimation method for power electronic systems based on the PIML. A PINN is specifically tailored through seamlessly integrating a deep neural network and the dynamic model of a Buck converter. As for the data-light feature, the proposed method can be well-trained based on a small training dataset of only 360 data samples, which can be easily prepared in practice. The simulation and hardware implementation results verify the performance in accuracy and robustness of the proposed parameter estimation method. We expect this work will serve as an inspiration for PIML in power electronic applications to uncover the potentials of integrating physics into data-driven approaches.

REFERENCES

[1] S. Zhao, S. W. Chen, F. Yang, E. Ugur, B. Akin, and H. Wang, "A composite failure precursor for condition monitoring and remaining useful life prediction of discrete power devices," *IEEE Trans. Ind. Informat.*, vol. 17, no. 1, pp. 688–698, Jan. 2021.
[2] S. Zhao, Y. Peng, F. Yang, E. Ugur, B. Akin, and H. Wang, "Health state estimation and remaining useful life prediction of power devices subject to noisy and aperiodic condition monitoring," *IEEE Trans. Instrum. Meas.*, vol. 70, Jan. 2021, Art. no. 3510416.
[3] M. Al-Greer, M. Armstrong, M. Ahmeid, and D. Giaouris, "Advances on system identification techniques for DC-DC switch mode power converter applications," *IEEE Trans. Power Electron.*, vol. 34, no. 7, pp. 6973–6990, Jul. 2019.
[4] H. Soliman, I. Abdelsalam, H. Wang, and F. Blaabjerg, "Artificial neural network based DC-link capacitance estimation in a diode-bridge front-end inverter system," in *Proc. IEEE 3rd Int. Future Energy Electron. Conf.*, 2017, pp. 196–201.
[5] T. Kamel, Y. Biletskiy, and L. Chang, "Capacitor aging detection for the DC filters in the power electronic converters using ANFIS algorithm," in *Proc. IEEE 28th Can. Conf. Elect. Comput. Eng.*, 2015, pp. 663–668.
[6] H. Soliman, P. Davari, H. Wang, and F. Blaabjerg, "Capacitance estimation algorithm based on DC-link voltage harmonics using artificial neural network in three-phase motor drive systems," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2017, pp. 5795–5802.
[7] H. Soliman, H. Wang, and F. Blaabjerg, "A review of the condition monitoring of capacitors in power electronic converters," *IEEE Trans. Ind. Appl.*, vol. 52, no. 6, pp. 4976–4989, Nov./Dec. 2016.
[8] Y. Z. Peng, S. Zhao, and H. Wang, "A digital twin based estimation method for health indicators of DC-DC converters," *IEEE Trans. Power Electron.*, vol. 36, no. 2, pp. 2105–2118, Feb. 2021.
[9] M. Milton, C. De La O, H. L. Ginn, and A. Benigni, "Controller-embeddable probabilistic real-time digital twins for power electronic converter diagnostics," *IEEE Trans. Power Electron.*, vol. 35, no. 9, pp. 9850–9864, Sep. 2020.
[10] M. Garaj, K. Y. Hong, H. S.-H. Chung, J. Zhou, and A. W.-L. Lo, "Photovoltaic panel health diagnostic system for solar power plants," in *Proc. IEEE Energy Convers. Congr. Expo.*, 2018, pp. 1–6.
[11] W. Wang, A. C. Liu, H. S. Chung, R. W. Lau, J. Zhang, and A. W. Lo, "Fault diagnosis of photovoltaic panels using dynamic current-voltage characteristics," *IEEE Trans. Power Electron.*, vol. 31, no. 2, pp. 1588–1599, Feb. 2016.
[12] S. Zhao, F. Blaabjerg, and H. Wang, "An overview of artificial intelligence applications for power electronics," *IEEE Trans. Power Electron.*, vol. 36, no. 4, pp. 4633–4658, Apr. 2021.
[13] S. Zhao and H. Wang, "Enabling data-driven condition monitoring of power electronic systems with artificial intelligence: Concepts, tools, and developments," *IEEE Power Electron. Mag.*, vol. 8, no. 1, pp. 18–27, Mar. 2021.
[14] L. von Rueden *et al.*, "Informed machine learning–a taxonomy and survey of integrating knowledge into learning systems," *IEEE Trans. Knowl. Data Eng.*, to be published, doi: 10.1109/TKDE.2021.3079836.

[15] B. X. Li and K. S. Low, "Low sampling rate online parameters monitoring of DC-DC converters for predictive-maintenance using biogeography-based optimization," *IEEE Trans. Power Electron.*, vol. 31, no. 4, pp. 2870–2879, Apr. 2016.
[16] J. Poon, P. Jain, C. Spanos, S. K. Panda, and S. R. Sanders, "Fault prognosis for power electronics systems using adaptive parameter identification," *IEEE Trans Ind. Appl.*, vol. 53, no. 3, pp. 2862–2870, May/Jun. 2017.
[17] F. Alonge, F. D'Ippolito, and T. Cangemi, "Identification and robust control of DC/DC converter hammerstein model," *IEEE Trans. Power Electron.*, vol. 23, no. 6, pp. 2990–3003, Nov. 2008.
[18] G. E. Karniadakis, I. G. Kevrekidis, L. Lu, P. Perdikaris, S. Wang, and L. Yang, "Physics-informed machine learning," *Nature Rev. Phys.*, vol. 3, no. 6, pp. 422–440, May 2021.
[19] K. E. Willcox, O. Ghattas, and P. Heimbach, "The imperative of physics-based modeling and inverse theory in computational science," *Nature Comput. Sci.*, vol. 1, no. 3, pp. 166–168, Mar. 2021.
[20] M. Raissi, A. Yazdani, and G. E. Karniadakis, "Hidden fluid mechanics: Learning velocity and pressure fields from flow visualizations," *Science*, vol. 367, no. 6481, pp. 1026–1030, Feb. 2020.
[21] G. S. Misyris, A. Venzke, and S. Chatzivasileiadis, "Physics-informed neural networks for power systems," in *Proc. IEEE Power Energy Soc. Gen. Meeting*, 2020, pp. 1–5.
[22] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations," *J. Comput. Phys.*, vol. 378, pp. 686–707, Feb. 2019.
[23] M. Raissi, P. Perdikaris, and G. E. Karniadakis, "Machine learning of linear differential equations using gaussian processes," *J. Comput. Phys.*, vol. 348, pp. 683–693, Nov. 2017.
[24] L. Lu, X. Meng, Z. Mao, and G. E. Karniadakis, "DeepXDE: A deep learning library for solving differential equations," *J. SIAM Rev.*, vol. 63, no. 1, pp. 208–228, Feb. 2021.
[25] E. Haghighat and R. Juanes, "SciANN: A keras/tensorflow wrapper for scientific computations and physics-informed deep learning using artificial neural networks," *Comput. Methods Appl. Mechanics Eng.*, vol. 373, Jan. 2021, Art. no. 113552.
[26] O. Hennigh *et al.*, "NVIDIA SimNet: An Ai-Accelerated multi-physics simulation framework," in *Proc. Int. Conf. Comput. Sci.*, 2020, pp. 447–461.
[27] I. E. Lagaris, A. Likas, and D. I. Fotiadis, "Artificial neural networks for solving ordinary and partial differential equations," *IEEE Trans. Neural Netw.*, vol. 9, no. 5, pp. 987–1000, Sep. 1998.
[28] A. Iserles, *A First Course in the Numerical Analysis of Differential Equations*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

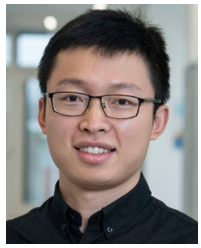
Shuai Zhao (Member, IEEE) received the B.S., M.S., and Ph.D. degrees in information and telecommunication engineering from Northwestern Polytechnical University, Xi'an, China, in 2011, 2014, and 2018, respectively.

He is currently a Postdoctoral Researcher with the Center of Reliable Power Electronics (CORPE), Department of Energy, Aalborg University, Aalborg, Denmark. From September 2014 to September 2016, he was a Visiting Ph.D. student with the Department of Mechanical and Industrial Engineering, University of Toronto, Toronto, ON, Canada. In August 2018, he was a Visiting Scholar with the Power Electronics and Drives Laboratory, Department of Electrical and Computer Science, University of Texas at Dallas, Richardson, TX, USA. His research interests include physics-informed machine learning, system informatics, condition monitoring, diagnostics and prognostics, and tailored AI tools for power electronic systems.

Dr. Zhao was the recipient of the scholarship from China Scholarship Council.

Yingzhou Peng (Member, IEEE) received the B.S. degree in electrical engineering from Harbin Engineering University, Harbin, China, in 2014, the M.S. degree in power electronics from Chongqing University, Chongqing, China, in 2017, and the Ph.D. degree in power electronics from Aalborg University, Aalborg, Denmark, in 2020.

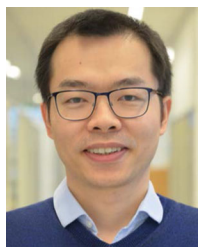
He is currently a Postdoc with Aalborg University, Aalborg, Denmark. In 2020, he was a Visiting Researcher with the Electrical Power and Energy Conversion Lab, Cambridge University, Cambridge, U.K. His research interests include the failure mechanisms analysis of power electronic components, and the improvement of the robustness and reliability of power converters by means of condition monitoring.



Yi Zhang (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from the Harbin Institute of Technology, China, in 2014 and 2016, respectively, and the Ph.D. degree from Aalborg University, Aalborg, Denmark, in 2020.

He is currently a DFF International Postdoc affiliated with both Aalborg University and the Massachusetts Institute of Technology, Cambridge, MA, USA. From November 2018 to February 2019, he was a Visiting Scholar with the Georgia Institute of Technology, Atlanta, GA, USA. His research interests include the reliability and multiphysics modeling of power semiconductor devices and power converters.

Dr. Zhang was the recipient of the First Place Prize Paper Award for 2020 in the IEEE TRANSACTIONS ON POWER ELECTRONICS in 2021, and the IEEE Power Electronics Society Ph.D. Thesis Talk Award Winner in 2020.



Huai Wang (Senior Member, IEEE) received the B.E. degree in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2007 and the Ph.D. degree in power electronics from the City University of Hong Kong, Hong Kong, China, in 2012.

He is currently a Professor with AAU Energy, Aalborg University, Aalborg, Denmark, where he leads the Group of Reliability of Power Electronic Converters and the mission on digital transformation and AI. He was a Visiting Scientist with the ETH Zurich, Zürich, Switzerland, from August to September 2014, and with the Massachusetts Institute of Technology, Cambridge, MA, USA, from September to November 2013. He was with the ABB Corporate Research Center, Baden, Switzerland, in 2009. His research interests include the fundamental challenges in modeling and validating power electronic component failure mechanisms and application issues in system-level predictability, condition monitoring, circuit architecture, and robustness design.

Dr. Wang was the recipient of the Richard M. Bass Outstanding Young Power Electronics Engineer Award from the IEEE Power Electronics Society in 2016 and the 1st Prize Paper Award from the IEEE TRANSACTIONS ON POWER ELECTRONICS in 2021. He is currently an Associate Editor for the *Journal of Emerging and Selected Topics in Power Electronics* and IEEE TRANSACTIONS ON POWER ELECTRONICS.