A Novel Nonlinear Deep Reinforcement Learning Controller for DC–DC Power Buck Converters

Meysam Gheisarnejad, Hamed Farsizadeh, and Mohammad Hassan Khooban, *Senior Member, IEEE*

***Abstract*—The nonlinearities and unmodeled dynamics inevitably degrade the quality and reliability of power conversion, and as a result, pose big challenges on higher-performance voltage stabilization of dc–dc buck converters. The stability of such power electronic equipment is further threatened when feeding the nonideal constant power loads (CPLs) because of the induced negative impedance specifications. In response to these challenges, the advanced regulatory and technological mechanisms associated with the converters require to be developed to efficiently implement these interface systems in the microgrid configuration. This article addresses an intelligent proportional-integral based on sliding mode (SM) observer to mitigate the destructive impedance instabilities of nonideal CPLs with time-varying nature in the ultralocal model sense. In particular, in the current article, an auxiliary deep deterministic policy gradient (DDPG) controller is adaptively developed to decrease the observer estimation error and further ameliorate the dynamic characteristics of dc–dc buck converters. The design of the DDPG is realized in two parts: (i) an actor-network which generates the policy commands, while (ii) a critic-network evaluates the quality of the policy command generated by the actor. The suggested strategy establishes the DDPG-based control to handle for what the iPI-based SM observer is unable to compensate. In this application, the weight coefficients of the actor and critic networks are trained based on the reward feedback of the voltage error, by using the gradient descent scheme. Finally, to investigate the merits and implementation feasibility of the suggested method, some experimental results on a laboratory prototype of the dc–dc buck converter, which feeds a time-varying CPL, are presented.**

***Index Terms*—Buck converter, constant power load (CPL), deep deterministic policy gradient (DDPG), sliding mode (SM) observer, ultralocal model (ULM).**

Nomenclature

P	Rated power of the CPL (W).
v_{ref}	Reference voltage (V).
v_o	Output voltage (V).
v_C	Capacitor voltage (V).
i_{CPL}	Current of the CPL (A).
v_{CPL}	Voltage of the CPL (V).
$x_1,\, x_2$	Average of the inductor current and capacitor voltage.
i_L	Inductor current (A).
v_C	Capacitor voltage of the dc–dc buck converter (V).
L	Inductance of the dc–dc buck converter (mH).
C	Capacitance of the dc–dc buck converter (μF).
χ	Subset of R^2.
$\dot{y}^*(t)$	Desired reference.
$\Theta(\circ)$	Unknown functional.
Φ	Unknown dynamics.
$\hat{\Phi}$	Unknown dynamics estimation.
$\zeta(t)$	Estimation error.
σ	Observer design coefficient.
Λ	Nonphysical design coefficient.
λ	Derivative order of the plant output.
$\mathcal{S}$	State space.
$\mathcal{A}$	Action space.
P	Reward function.
γ	Discount factor.
$(s_t,\, a_t)$	State-action pair.
$\pi(a_t \vert s_t)$	Policy function.
G_t	Accumulated discounted reward.
$\mathcal{Q}^{\pi}(s_t,\, a_t)$	Action-value function.
$\mathcal{Q}(s, a \vert \theta^{\mathcal{Q}})$	Output of the critic-network.
$\mu(s \vert \theta^{\mu})$	Output of the actor-network.
$\theta^{\mathcal{Q}}$	Weight coefficient of the critic-network.
θ^{μ}	Weight coefficient of the actor-network.
$\mathcal{Q}'(s, a \vert \theta^{\mathcal{Q}'})$	Output of critic target.
$\mu'(s \vert \theta^{\mu'})$	Output of actor target.
$L(\theta^{\mathcal{Q}})$	Loss function.

Abbreviations

CPL	Constant power load.
MG	Microgrid
ULM	Ultralocal model.
iPI	Intelligent proportional-integral.
RL	Reinforcement learning.

Manuscript received October 31, 2019; revised February 20, 2020, April 3, 2020, and May 24, 2020; accepted June 12, 2020. Date of publication July 1, 2020; date of current version April 27, 2021. *(Corresponding author: Mohammad Hassan Khooban.)*

Meysam Gheisarnejad is with the Department of Electrical Engineering, Najafabad Branch, Islamic Azad University, Isfahan, Iran (e-mail: me.gheisarnejad@gmail.com).

Hamed Farsizadeh is with the Shiraz University of Technology, Shiraz 25529, Iran (e-mail: hthfarsi@gmail.comand).

Mohammad Hassan Khooban is with the DIGIT, Department of Engineering, Aarhus University, Finlandsgade 22, Aarhus 8200, Denmark (e-mail: khooban@ieee.org).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIE.2020.3005071

MDP	Markov decision process.
DQN	Deep Q network.
DDPG	Deep deterministic policy gradient.
ReLU	Rectified linear unit.
SISO	Single input/single output.
FL	Fuzzy logic.
SDP	Semidefinite programming.
SMC	Sliding mode control.
MPC	Model predictive control RBF.
NN	Radial basis function neural network.
PHiL	Power hardware-in- the loop.

I. Introduction

DC MICROGRIDS (MGs) consists of an autonomous cluster of microsources and backup devices, which are connected to the source bus by the power electronic circuits [1]–[3]. In the MG context, fuel cells, in particular, are known as one of the most profitable renewable source technologies due to their great density of power generation, high efficiency, and durability. In practical MGs, the fuel cells can be coupled with power electronic systems as a typical case to ameliorate the efficiency and reliability of the whole MG system. The successful implementation of fuel cells in some real-word plants, over recent years, like marine systems [4]–[6] and spacecraft [7] has been found in the literature.

The controllers of each distributed generations (DGs) in dc MGs with fuel cell are regulated to control either a current, voltage or desired power. With the advances in digital control and interface converters, these power systems can track the nonperiodic/periodic reference signals and reject any disturbances with an insignificant deterioration of output [8], [9]. However, in an MG with the fuel cell, many loads (e.g. inverter motor drives, power supplies, etc. [10]–[12]) with the tightly controlled loads act as constant power loads (CPLs), which impose a nonlinear and destabilizing impact on the converters due to inverse voltage phenomena. The fuel cell in a special dc MG is regarded as a DG while the loads connected to the MG system is regarded as CPLs. Based on the concepts of the nonlinear sliding surface, a robust sliding mode controller (SMC) has been developed in [13] to alleviate the instability effects of CPL in a dc–dc boost converter. It is assumed in this article that the dc–dc converter is supplied by a solar panel, fuel cell or other renewable resources. To deliver reliable energy to the propulsion of a marine MG, an intelligent control strategy is suggested in [14] for a hybrid fuel cell/battery structure of a dc islanded MG feeding CPLs.

To mitigate the destruction feature of the CPLs, various control methodologies are developed by ameliorating the control law of injecting power and load (or source) converter, such as active disturbance rejection control [15], backstepping algorithm [12] and semidefinite programming [8]. Due to the negative impedance specification of CPLs, designing a robust controller for the converter power systems feeding such loads is a difficult task as it requires an exact mathematical model of the plant to be controlled. Based on the known restricted bound, model-based techniques, such as terminal SMC [16], linear matrix inequality [17], and robust nonfragile [18] have been established in the field of dc converters with promising results. However, the unmodeled dynamics, that cannot be expressed using deterministic mathematical approaches, make the control design of dc–dc converter too difficult and do not allow the model-based techniques to project precisely or perform satisfactorily. The motivation to establish a model-independent controller for the dc–dc converters is that, in practice, such systems are faced with inaccuracies in their models, quantization impacts and uncertainties. To deal with this issue, the model-independent techniques have been formalized to tackle with uncertain or partially uncertain dynamics while preserving the light computational burthen of proportional integral derivative (PID) by adopting a so-called intelligent PID or iPID. The intelligent techniques are developed based on the ultra-local model (ULM) estimation; where this dynamic approximation is updated based only on the knowledge of the input-output (I/O) measurements. In [19] and [20], the uncertain dynamics are estimated on a very short time interval by a ULM that is updated online based on the algebraic derivation techniques. However, the algebraic estimators are insufficient to guarantee a good reference tracking in each time instant of implementation. The reason is that due to the measurement noises and controller errors, the algebraic derivation techniques are unable to derive the tracking error to zero quickly. In response to the estimation error, an auxiliary controller [e.g., sliding mode control (SMC) [21] and fuzzy logic (FL) [22], etc.] is often added to the standard iPID-based ULM control to compensate the uncertain or changeable components. However, these mentioned works are appropriate only for a particular cycle period and suffer the lack of online learning ability and poor generalization properties. In [23], a radial basis function neural network (RBF-NN) is introduced into the model-independent-based iPID control. This additional observer is not only used to achieve the tracking trajectory efficiently but also has a capability of self-learning.

But as mentioned earlier, the destructive effect of unknown powers of CPLs highly threaten the stability of the dc–dc converters, which lead to the NNs or RBF-NNs become ineffective to approximate the uncertain model due to their limited learning capability. Developing “deep Q network (DQN)” , as an advanced reinforcement learning (RL), can be an effective approach for addressing the limitations of the earlier adopted methodologies [24]. Using deep NNs and RL, DQN can learn an action-value function by employing a Bellman equation in an iterative process. With the significant progress of some new methods like experience replay and minibatch learning in training the NNs, the DQN is emerged as a powerful data-driven scheme to solve complex tasks, such as autonomous underwater vehicles [25], aerial robot [26], [27], and control of a quadrotor [28]. Despite the great successes of DQN, the training of NNs on the complex problems with large-scale datasets is computationally exhaustive. Moreover, applying the DQN in continuous action spaces is accompanied by the following issues: correlated data, poor diversification, and instability. In this respect, Lillicrap *et al.* [29] developed the deep deterministic policy gradient (DDPG) to extend the DQN techniques, by employing direct representation of a policy, to conquer the limitations of discrete actions.

Due to the highly nonlinear behavior and the stochastic feature associated with renewable sources, the deterministic control methodologies prove to be insufficient to stabilize the dc–dc converters. Experimental studies in the literature proved that the CPL in the dc–dc converters poses a destabilizing impact on the circuit and can result in a sever voltage distractions [30], [31]. Consequently, this work proposes a DDPG intelligent PI (DDPGiPI) based on a sliding mode (SM) observer to handle the destructive effects of the CPLs in the dc–dc buck converter. The ULM based on the I/O information of the buck converter is adopted, to eliminate the dependence of the controller on the accurate converter model.

The contributions of the article can be summarized as follows.

1) Often, the control design of the dc–dc buck converters is based on assumption that the CPLs are ideal [8], [12], but in practice, the CPLs have unknown and/or time-varying nature which are so-called nonideal CPLs [13], [32], [33]. Under such circumstances, the robust deterministic methodologies no longer ensure the stable performance of the power electronic systems. To address this issue, this article presents a novel model-independent controller against the power changes in the nonideal CPLs and the stability of the overall system is investigated within a wide range of power variations.
2) The ULM based on the I/O information of the buck converter is adopted to eliminate the dependence of the controller on the accurate converter model while the SM observer is established to estimate the poorly known dynamics of the ULM.
3) The voltage term of the dc–dc buck converter is regulated and stabilized by the DDPGiPI controller, to ameliorate the response voltage and transient tracking precision of the converter system. More specifically, the DDPG algorithm is developed in an actor-critic framework, where the weights of the deep NNs of the DDPG are updated in a way that compensates the estimation error of the SM observer adaptively [34].
4) Finally, dSPACE-based real-time examinations are conducted to verify the merits of the suggested controller from a systematic perspective. A comparative analysis is carried out subsequently to ascertain the supremacy of the model-independent DDPGiPI scheme based on the SM observer over that of the intelligent PI controller [35] and the model predictive control (MPC) approach [36].

The remainder of the article is structured as follows. The state-space model of the dc–dc buck converter with CPLs is presented in Section II. Then, in Section III, the detail of the suggested adaptive DDPGiPI controller is provided. The formulation of the SM observer based on the Lyapunov theorem is also presented in Section III. The dSPACE outcomes are given in Section IV to experimentally ascertain the applicability of the model-independent scheme. Finally, Section V concludes this article.

II. Model of a dc–dc Buck Converter With CPLs

Consider the integrated dc onboard MG depicted in Fig. 1 [12], [37], which consists of various storage devices (e.g., battery systems, fuel cell, and flywheel), and a dc source and a CPL. For small-signal analysis, a simplified circuit diagram of the interface dc–dc buck converter is depicted in Fig. 2 [15]. The concerned converter illustrated in Fig. 2, considers one pulsewidth modulation based dc–dc buck converter receiving its supply from the renewable energy sources (e.g. fuel cell) and supplying a CPL.

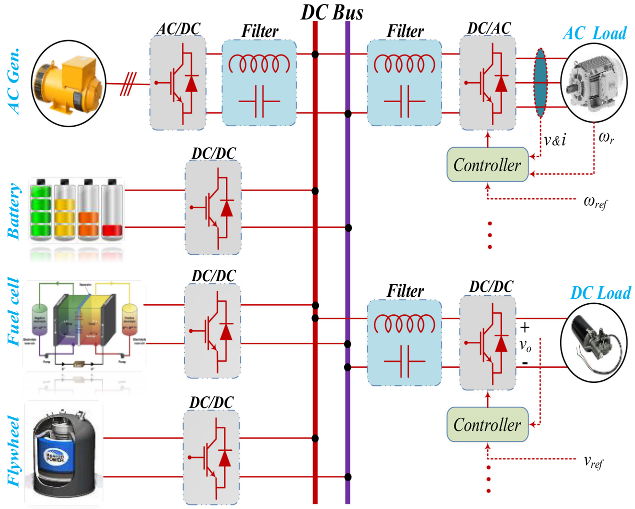


Fig. 1. Typical architecture of dc onboard MGs.

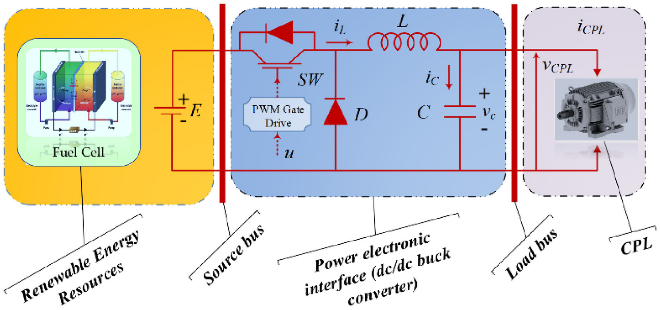


Fig. 2. Simplified circuit diagram of a dc–dc buck converter feeding CPL.

For the concerned power electronic system, the converter is regulated by the duty cycle u of the MOSFET in a way that the output voltage v_o remains stable. The source buck converter has input voltage *E*. Considering that the total load disturbances are defined by a CPL, i.e., which refers to a worst-case condition in terms of stability, the instantaneous current derived from the CPL is given by

$$i_{\mathrm{CPL}}\ (t) = \frac{P}{v_{\mathrm{CPL}}\ (t)}\ \forall\, v_{\mathrm{CPL}}\ (t)\ > \varepsilon \tag{1}$$

where P is the rated power the CPL; i_{CPL} is the current generated by CPL; v_{CPL} is the CPL voltage which has an equal amount to the capacitor voltage v_C; and ε has a small positive value.

Typically, the state-space model is adopted for the analysis and control design purposes of the buck type of dc–dc converters. Based on the average switching scheme [38], the simplified dc–dc buck converter feeding CPL depicted in Fig. 2 can be mathematically expressed as

$$\frac{dx_1}{dt} = \frac{1}{L}\ [E.u - x_2] \tag{2}$$

$$\frac{dx_2}{dt} = \frac{1}{C}\ \left[x_1 - \frac{P}{x_2}\right] \tag{3}$$

$$y\ = v_o\ . \tag{4}$$

In the earlier equations, x_1 and x_2 denote the average of the inductor current i_L and capacitor voltage v_C, respectively; C and L denote the capacitance and inductance of the buck converter, respectively; and the control input is represented by $u \in \{0,\ 1\}$. The readers can refer to [39], [40] to acquire more information about the modeling of dc–dc power Buck converters feeding CPLs.

III. Design of the Model-Independent DDPG Intelligent PI Controller-Based SM Observer

The general principle of the model-independent controllers is not new and it encompasses a wide range of control methodologies that show the lowest reliance on the mathematical models of the plant. Examples for these approaches adopted in the model-independent controllers are fuzzy logic and neural networks and so on. Recently, in the relevant literature, another variant of the model-independent approaches based on the concept of the ULM is developed, which has been successfully adopted in various applications [35]. It should be noted that the model-independent does not necessarily mean that there is no dynamic model, but due to the uncertainties and the unmodeled dynamics, obtaining a precise model describing the system dynamics is a complex and time-consuming task.

Inspired by the earlier discussion, a new model-independent DDPGiPI scheme based on the ULM is proposed in this section. The suggested structured controller consists of two subcomponents: First, an intelligent PI sub-controller based on SM observer for accommodating the unmodeled phenomena, perturbations and poorly known dynamics. Second, a DDPG subcontroller for compensating the estimation error of the SM observer by ameliorating the efficacy of overall controller performance.

A. Design of PI Type Intelligent Controller Based on the SM Observer

1) General Principles of ULM: It is assumed that only the dynamic behavior of the system is properly approximated within its operating range by employing the ordinary differential techniques, which might have nonlinear nature and time-varying characteristics. Based on the implicit function theorem [41], the ULM representation of a single-input/single-output plant can be described by an unknown ordinary differential, given as

$$\boldsymbol{F}\ \left(\boldsymbol{t},\ \boldsymbol{y}, \dot{\boldsymbol{y}}, \ldots, \boldsymbol{y}^{(\boldsymbol{l})}, \boldsymbol{u}, \dot{\boldsymbol{u}}, \ldots, \boldsymbol{u}^{(\boldsymbol{k})}\right) = \ \boldsymbol{0} \tag{5}$$

where l and k are the derivative order of the output variable (y) and the input variable u, respectively, and $\boldsymbol{F}$ denotes an adequately smooth function of its arguments. Define that also for an integer variable λ, $0 < \lambda < l$, $\partial \boldsymbol{F}/\partial \mathrm{y}^{\lambda}$. From implicit function theorem, the input/output feature can be approximately described by a completely unknown or partially known finite-dimensional differential equation

$$\mathrm{y}^{(\lambda)} = \ \Theta \left(t,\ y, \dot{y}, \ldots, y^{(\lambda-1)}, y^{(\lambda+1)}, \ldots y^{(l)}, u, \dot{u}, \ldots, u^{(k)}\right). \tag{6}$$

Note that, in the above equation, the unknown functional, $\Theta(\circ)$, does not necessarily have a form of linear or time-invariant. By setting $\Theta\ =\ \Phi + \Lambda u$, the instantaneous representation of the model-independent scheme is described as [22], [35]

$$y^{(\lambda)}\ (t)\ =\ \Phi\ (\mathrm{t}) + \Lambda u\ (t) \tag{7}$$

where Φ includes all the unmodeled phenomena and perturbations (e.g., nonlinearities, friction, uncertainties, etc.), which is estimated using $u(t)$ and the estimate of $y^{(\lambda)}(t)$ at every sampling instant. $\Lambda\ \epsilon\ \mathbb{R}$ is a nonphysical design coefficient, which is designed in such a way that $\Lambda u(t)$ and $y^{(\lambda)}(t)$ have the same order of magnitude.

Remark 1: Note that, the term λ is fully independent of the order of the unknown plant and may, in general, be chosen quite small, i.e., 1, 2. Referring to the literature related to the ULM schemes, almost all existing studies concrete the test-systems with $\lambda\ =\ 1$ while only a few the test-systems are provided by $\lambda\ =\ 2$ [42].

Assume that $\Phi(\mathrm{t})$ and Λ are well-known terms, based on [21], the control input of intelligent PI by considering $\lambda\ =\ 1$ is described as

$$u\ \ (t) = \frac{1}{\Lambda}\ \left(-\hat{\Phi}\ (t) +\ \dot{y}^{*}\ (t) + k_p e_1\ (t) + k_i \int e_1\ (t)\right) \tag{8}$$

where k_p and k_i are the coefficients of equivalent conventional proportional-integral (PI); $\dot{y}^{*}(t)$ denotes the desired reference; $e_1(t)\ \ = y^{*}\ (t) - y(t)$ denotes the output error; and $\hat{\Phi}(t)$ is the estimated value of $\Phi(t)$. Define the estimation error as $\zeta(t)\ =\ \Phi(t) - \hat{\Phi}(t)$, substituting (8) into (7) yields

$$e_1\ (t) + k_p e_1\ (t) + k_i \int e_1\ (t)\ \ =\ \zeta\ (t)\ . \tag{9}$$

Remark 2: Based on (7), the ULM of the dc–dc buck converter is built. For such systems, $\Phi(t)$ does not distinguish between the nonlinearities and unknown parts of the system. The scalar variable $\hat{\Phi}(t)$ must be updated at each sample time since (7) is valid just for a short period window. The SM observer is adopted to estimate the coefficient $\hat{\Phi}(t)$ and feedback to the DDPG intelligent PI structure.

Remark 3: If the uncertainties and perturbations are adequately small, the error will converge to zero, i.e., $\zeta(t) \rightarrow 0$. In the absence of the unknown parts, (9) converts into a linear equation which can ensure the closed-loop stability. But from the control engineering point of view, due to the unmodeled dynamics in real systems, $\zeta(t)$ should not necessarily be zero. Thus, by appropriate adjustment of the control parameters (k_p and k_i), $\zeta(t)$ is merely under bounded.

2) Design of SM observer: To estimate the unknown dynamics of a system, the SM observer is mostly adopted in the literature, like [43], [44]. Generally, there are three fundamental reasons for the popularity of this type of observers; these are

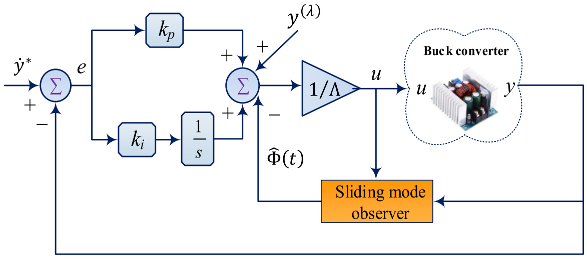


Fig. 3. Structure of the model-independent iPI controller with SM observer.

simplicity, flexibility in design, and robust control framework, which partially compensates the dependence of the observer on the plant model. The structure of the ULM controller-based SM observer is illustrated in Fig. 3. In this application, the iPI controller is established as the feedback controller to stabilize the voltage output of the converter. The SM observer is adopted to estimate the undesirable disturbances (or unmodeled dynamics) of the ULM and return it to the controller.

In (7), $\Phi(t)$ represents the unknown parts of the buck converter, the estimated value $\hat{\Phi}(t)$ will be calculated using an SM observer. The SM observer designed according to the ULM expressed by (7) is as [45]

$$\dot{\hat{x}}_2 = \sigma \mathrm{sgn}\left(x_2 - \hat{x}_2\right) + \Lambda u \tag{10}$$

where $\hat{x}_2$ denotes the observed value of x_2; $\sigma \mathrm{sgn}(x_2 - \hat{x}_2)$ denotes the sliding control term; σ is the designed coefficient; and $\mathrm{sgn}(\cdot)$ is a sign function.

The sliding surface of the observer is chosen as $e_2(t) = x_2 - \hat{x}_2$. By subtracting (7) from (10), one can obtain the error dynamics equation

$$\dot{e}_2\left(t\right) = \Phi - \sigma \mathrm{sgn}\left(x_2 - \hat{x}_2\right). \tag{11}$$

Theorem 1: For (11), the error value will be asymptotically converged to zero, if the SM manifold is selected as $s = e_2\left(t\right)$ and σ is designed appropriately.

Proof: Define the Lyapunov function

$$V = \frac{1}{2}\, s(t)^T s\left(t\right). \tag{12}$$

Differentiating (12) with respect to time, then

$$\dot{V} = s(t)^T \ \dot{s}\left(t\right) = e_2\ \left(t\right) \dot{e}_2\left(t\right). \tag{13}$$

Now, by substituting $\dot{e}_2(t)$ from (11)

$$\dot{V} = e_2\ \left(t\right)\left(\Phi\left(t\right) - \ \sigma \mathrm{sgn}\left(e_2\left(t\right)\right)\right) \leq \left|e_2\left(t\right)\right|\left(\left|\Phi\left(t\right)\right| - \sigma\right). \tag{14}$$

Suppose σ satisfies the condition of $|\Phi(t)| + \eta < \sigma$, where $\eta > 0$, according to (15), one can obtain

$$\dot{V} \leq -\eta \left|e_2\left(t\right)\right|. \tag{15}$$

This implies that the observer is stable asymptotically.

B. Deep Reinforcement Learning

1) Strategic Plan of Reinforcement Learning: Based on the reinforcement learning paradigm, an artificial agent, sequentially, interacts with an environment, to learn the optimal policies. When the environment is completely observable, the RL framework can be formally expressed by a Markov decision process (MDP) under the Markovian characteristic of the environment. An MDP is described by a five-tuple $\mathcal{S}, \mathcal{A}, P,\ R,\ \gamma$, where $\mathcal{S}$ is a state space, $\mathcal{A}$ is an action space, $P: \ \mathcal{S} \times \ \mathcal{A} \times \mathcal{S} \rightarrow [0,\ 1]$ is a state transition probability where $P = p(s_{t+1}|s_t, a_t)$, $R: \mathcal{S} \times \ \mathcal{A} \rightarrow \mathcal{R}$ is the reward function, and $\gamma \ \in [0,\ 1]$ is the discount factor. In each time step, an RL agent observes a state s_t and selects an action a_t, according to a policy $\pi(a_t|s_t)$ which maps from observations to actions. Then, the agent receives a reward feedback r_t for the undertaken action and devises the next state s_t. The definition of the discounted reward is given as

$$G_t = \sum_{k=0}^{\infty} \gamma^k r_{t+k}\,. \tag{16}$$

The task of RL is to maximize the expectation of a discounted return. Implementing a policy π, the objective function will be formalized as

$$J \ = \mathbb{E}_{r_i, s_i \sim En, a_i \sim\ \pi}\ \left[G_1\right] \tag{17}$$

where states were sampled from the environment En and actions a_i are derived from the policy π. In many RL problems, an action-value function is defined which expresses the expected return G_t after getting an action a_t in state s_t and then following policy π

$$\mathcal{Q}^{\ \pi}\ \left(s_t,\ a_t\right) = \mathbb{E}_{r_{i\geq t}, s_{i\geq t} \sim En, a_{i>t} \sim\ \pi}\ \left[G_t|s_t,\ a_t\right]. \tag{18}$$

Therefore, the key objective of RL is to compute the action-value function $\mathcal{Q}^{\ \pi}(s_t,\ a_t)$ and accordingly, find the policy π.

2) DQN and DDPG Algorithms: Similar to value iteration methodologies (e.g., adaptive dynamic programming, Q-learning, policy gradient, and so forth.), many methodologies in RL uses a recursive relationship, so-called Bellman equation, to recursively estimate the action-value function, as given in

$$\begin{aligned}\mathcal{Q}^{\ \pi}\ \left(s_t,\ a_t\right) = \mathbb{E}_{r_t, s_{t+1} \sim \mathrm{E}}\ [r_t\left(s_t,\ a_t\right) \\ + \gamma \mathbb{E}_{a_{t+1} \sim \pi}\left[\mathcal{Q}^{\ \pi}\left(s_{t+1},\ a_{t+1}\right)\right]].\end{aligned} \tag{19}$$

The property of a deterministic target policy can be characterized by a function $\mu: \ \mathcal{S} \leftarrow \mathcal{A}$ and afterward avoid the inner expectation.

Despite the iteration-based algorithms can find the optimal action-value function, the basic algorithms are quite impractical in real-world applications with high-dimensional problems. The reason is that without any generalization, the function is approximated independently for each sequence. A viable approach to addresses this problem is to estimate the action-value function via a deep neural network as an approximator, and accordingly, the DQN algorithm has emerged because of the need for accurate prediction.

In the DQN algorithm, especially, an experience replay memory R is intended in its training process to stores the recent transition tuples $(\mathrm{s_t},\ \mathrm{a_t},\ \mathrm{r_t},\ \mathrm{s_{t+1}})$. During the DQN learning, a mini-batch of tuples are chosen randomly from R.

Since the standard DQN works on a limit set of actions, it is not applicable to straightforwardly implement DQN to continuous action spaces, i.e., the discrete actions lead to violent system responses in some practical systems (e.g., power systems, robots,

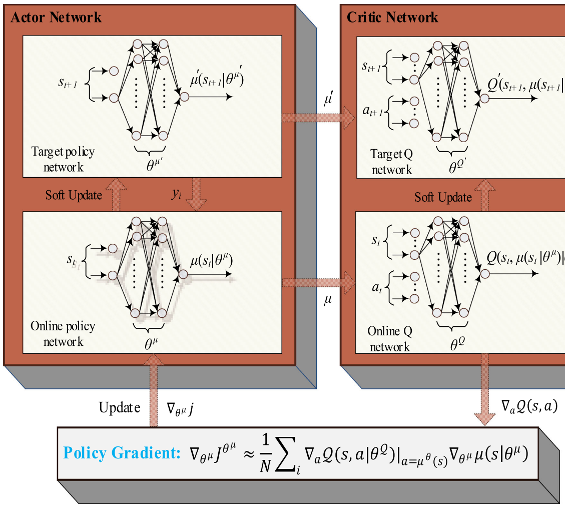


Fig. 4. Illustration of the DDPG algorithm with actor-critic architecture.

etc.). It is shown in many applications [25], [28], [46] which the DDPG provides superior performance than the DQN to solve control problems with continuous dynamics. Fig. 4 illustrates the overall procedure of the DDPG which is made up of two eponymous ingredients:

1) an actor-network that adjusts the weights θ^μ of policy $\mu(s|\theta^\mu)$ by fitting an observation (or state) to its corresponding action while and
2) a critic network adjusts the weights of the action-value function $Q(s,a|\theta^Q)$.

By minimizing the loss function $\mathcal{L}(\theta^Q)$ of (21), the DDPG updates the weight parameters of the critic-network θ^Q

$$\mathcal{L}\ \left(\theta^Q\right) = E_{(s,a)}\ \left[\left(Q\left(s_t,\ a_t|\theta^Q\right) - y_t\right)^2\right] \tag{20}$$

where

$$\begin{aligned} y_t &= r_t\ \left(s_t,\ a_t\right) + \gamma Q\left(s_{t+1}, \mu(s_t|\theta^\mu)\ |\theta^Q\right), \\ &= \mathbb{E}_{s_t\sim\rho^\beta}\ \left[\nabla_a Q(s,a|\theta^Q)|_{a\ =\mu^\theta\ (s)} \nabla_{\theta^\mu}\mu\left(s|\theta^\mu\right)\right]. \end{aligned} \tag{21}$$

The actor network coefficients θ^μ can be updated according to the following gradient:

$$\nabla_{\theta^\mu} J^{\theta^\mu} \approx \mathbb{E}_{s_t\sim\rho^\beta}\left[\nabla_{\theta^\mu} Q\left(s,a|\theta^Q\right)|_{a\ =\mu(s|\theta^\mu\ )} \nabla_{\theta^\mu}\mu\left(s|\theta^\mu\right)\right] \tag{22}$$

where ρ is the discounted distribution and β is a specific policy to the current policy π.

Remark 4: When updating the actor and critic networks, sharing the values of the weight coefficients for the components at time step t (current) and $t+1$ (target) leads to risks of divergence. The reason for the instabilities is that small variations in the network weights result in a large oscillation in the coefficients. To avoid the instability of the DDPG learning, two distinct networks consist of the critic target $Q'(s,a|\theta^{Q'})$ and the actor target $\mu'(s|\theta^{\mu'})$ are also implemented in the DDPG, in addition to the main networks (see Fig. 4). By employing the following soft updates, the target networks will follow the trained networks [46]

$$\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\,\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\,\theta^{\mu'} \tag{23}$$

where $\tau \ll 1$.

Besides, an exploration noise $\mathcal{N}$ based on the Ornstein–Uhlenbeck process [47] is added to the actor actions (i.e., $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}$) to ameliorate the training performance by the vicinity space of the suboptimal trajectory.

The pseudocode for the standard DDPG scheme is presented in Algorithm 1.

Algorithm 1: Pseudocode of the DDPG Algorithm With Actor-Critic Framework.

```
Randomly initialize critic $Q(s,a|\theta^Q)$ and actor $\mu(s|\theta^\mu)$ networks with weights $\theta^Q$ and $\theta^\mu$
Initialize target networks $Q'$ and $\mu'$ with weights $\theta^{Q'} \leftarrow \theta^Q, \theta^{\mu'} \leftarrow \theta^\mu$
Set up empty replay buffer $R$
for episode $= 1$ to $M$ do
    Begin with an Ornstein-Uhelnbeck (OU) noise $\mathcal{N}$ for exploration
    Receive initial observation state
    for t $= 1$ to $T$ do
    Apply action $a_t = \mu(s_t|\theta^\mu) + \mathcal{N}$ to environment
    Observe next state $s_{t+1}$ and reward $r_t$
    Store following transitions $(s_t,\ a_t,\ r_t,\ s_{t+1})$ into replay buffer $R$
    Sample random minibatch of $K$ transitions from $R$
    Set $y_i = r_i\ + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})\ |\theta^{Q'})$
    Update critic by the loss:
        $L\ = \frac{1}{N}\ \sum_{i\ =\ 1}^{n}\ (y_i - Q(s_i,\ a_i|\theta^Q))^2$
    Update the actor policy using the sampled policy gradient:
        $\nabla_{\theta^\mu} J^{\theta^\mu} \approx \frac{1}{N}\sum_i \nabla_a Q(s,a|\theta^Q)|_{a=\mu^\theta\ (s)} \nabla_{\theta^\mu}\mu(s|\theta^\mu)$
    Update the target networks:
        $\theta^{Q'} \leftarrow \tau\theta^Q + (1-\tau)\theta^{Q'}, \theta^{\mu'} \leftarrow \tau\theta^\mu + (1-\tau)\theta^{\mu'}$
    end for
end for
```

C. ULM Control Based on Deep Reinforcement Learning

Considering the negative impedance impact of the CPLs, a model-independent controller based on ULM is introduced into the dc–dc converters. In the context of ULM, an intelligent PI controller is developed to stabilize a dc–dc buck converter with CPLs, and the SM observer is designed to estimate the unmodeled dynamics and perturbations of the ULM. Also, an adaptive compensator based on the DDPG algorithm is added to the model-independent controller to improve the robustness and compensate for the SM observer estimation error [34]. The overall structure of the suggested adaptive control scheme is depicted in Fig. 5.

The extra DDPG input control to the converter is denoted by u_{DDPG}. Therefore, the final expression of the DDPG intelligent

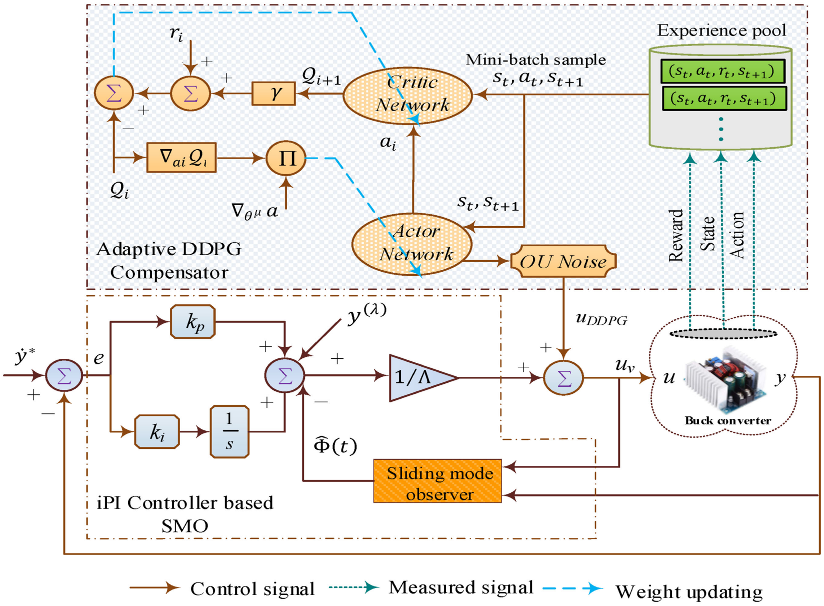


Fig. 5. Schematic diagram of the DDPGiPI control-based SM observer.

PI can be described as

$$u\ (t) = \frac{1}{\Lambda}\left(-\hat{\Phi}\,(t) + \ \dot{y}^*\,(t) + k_p e\,(t) + k_i \int e\,(t)\ \right) + u_{\text{DDPG}}\,(t)\,. \tag{24}$$

Remark 5: For the converter under investigation, the model-independent control offers stable and accurate tracking of the voltage reference, even with CPL and parametric variations. Notice that with adopting the auxiliary DDPG control actions (u_{DDPG}), the feedback controller (iPI controller) does not need to be suitably adjusted to meet the voltage output stability and robustness.

The optimal policy of the DDPG algorithm as illustrated in Fig. 5, has been developed based on solving the Bellman's equation in the actor-critic framework. In this algorithm, the critic evaluates the quality of the actor policy control, while the actor generates an auxiliary control action to remove the tracking error. The design of the actor and critic networks, which consist of one input layer, one output layer and three hidden layers containing 100, 100, and 20 neurons between the input and output layers, are demonstrated in Fig. 6. The input signals to the actor network are a vector state of the v_o, i_L, and e and their derivative in time step t, i.e., $s_{t,i} = \{v_o, i_L,\ e,\ (\frac{dv_o}{dt}),\ (\frac{di_L}{dt}),\ (\frac{de}{dt})\}$, and its output is the compensator signal of the SM observer $\mu(s_t|\theta^\mu)$. The state vector s_t and an action $\mu(s_t|\theta^\mu)$ are adopted as inputs of the critic network which outputs an approximate Q-value $Q(s_t, \mu(s_t|\theta^\mu)|\theta^Q)$.

For the stabilization of buck converter with CPL, the control goal is to minimize the output voltage error in the shortest time. To quantify the performance of the converter output, the reward function r is defined based on the sum of voltage output errors in each step time, as given in

$$r_t = \frac{1}{|y^*\,(t) - y\,(t)|}\,. \tag{25}$$

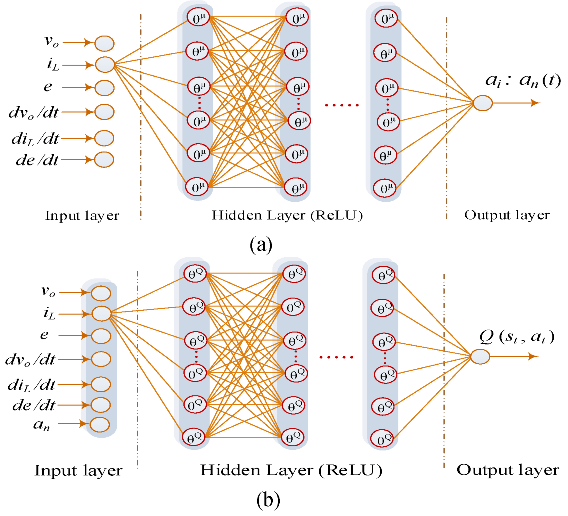


Fig. 6. Illustration of the (a) actor-network and (b) critic-network.

Based on the considered reward signal, the weight coefficients of DDPG networks (i.e., actor and critic) will be trained in such that minimizes the error between the reference voltage v_{ref} with its actual value v_o.

IV. Experimental Results

In this section, the DDPG intelligent PI based on the SM observer is developed to stabilize a buck converter feeding time-varying CPL in a typical dc MG scenario. In the comparative studies, the transient outcomes of the suggested method are compared with that of iPI and MPC controllers. The hyperparameters for the design of the DDPG controller are given in Table I.

To validate the model-independent feature and adaptation capability of the suggested controller, it is examined experimentally on a dc–dc buck converter given in Fig. 7. The dSPACE

TABLE I
HYPER PARAMETERS SETTINGS OF THE DDPG

Hyper Parameter	Values
Discount factor, γ	0.99
Learning rate, λ	0.0005
Mini-batch size	32
Replay buffer size	20000
Soft target update, τ	0.01

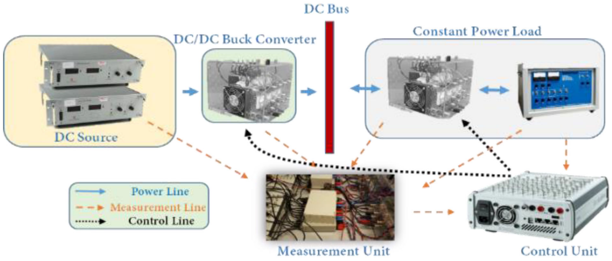


Fig. 7. Real-time simulation setup.

TABLE II
PARAMETERS OF THE BUCK CONVERTER

Parameter	Values
Inductance, L	1 mH
Capacitance, C	1000 μf
Converter input voltage, E	110 V
DC bus voltage reference, V_{ref}	48 V

MicroLabBox with DS1202 PowerPC DualCore 2 GHz processor board and DS1302 I/O board is adopted to investigate the applicability of the model-independent controller from the real-time perspective. In this application, the dSPACE testbed is adopted to provide a power hardware-in-the loop (PHiL) simulation by interfacing the control section to the electrical section. The electrical setup for a typical buck converter in the real-time testbed is given in Table II.

The PHiL responses of the system with time-varying CPL including the power tracking, inductor current, and dc bus voltage are depicted in Figs. 8–10. The transient outcomes of the system are obtained under the following operating condition

$$P = \begin{cases} 250\ W \text{ for } t \in [0,\ 0.3) \text{ sec} \\ 150\ W \text{ for } t \in [0.3,\ 0.7) \text{ sec} \\ 350\ W \text{ for } t \in [0.7,\ 1) \text{ sec} \end{cases} . \quad (26)$$

In response to the considered changes of the CPL power, one can observe from Fig. 8 that by employing the DDPGiPI based on the SM observer, the voltage term remains at a constant value, while the set-points of the inductor current term are tracked accurately. In this test case, a negligible steady-state error with values less than 2% is reached for the output voltage which is quite acceptable from a power engineering perspective. In comparison with the MPC and iPI approaches (Figs. 9 and 10), a higher level of reliable operation of the buck converter against the instantaneous step changes of the CPL power is yielded by the suggested controller. The reason is that despite the large CPL changes, the MPC scheme and intelligent PI controller can still stabilize the voltage output but are not optimal enough, i.e., the steady error in the system output is more than 2%.

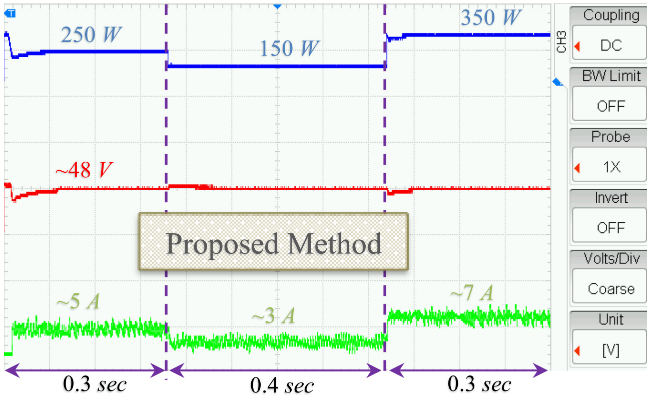


Fig. 8. Transient outcomes of the DDPGiPI based on the SM observer controller (the CPL power variations, bus voltage, and inductor current are depicted with blue, red, and green curves, respectively).

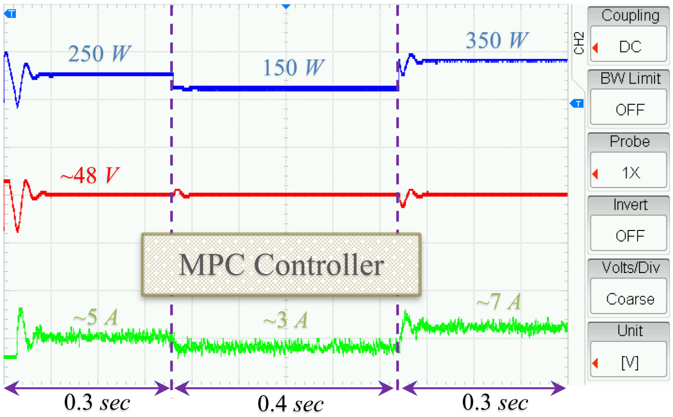


Fig. 9. Transient outcomes of the MPC controller (the CPL power variations, bus voltage, and inductor current are depicted with blue, red, and green curves, respectively).

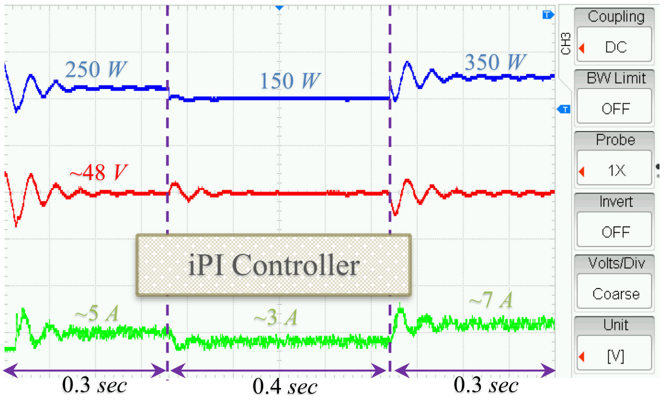
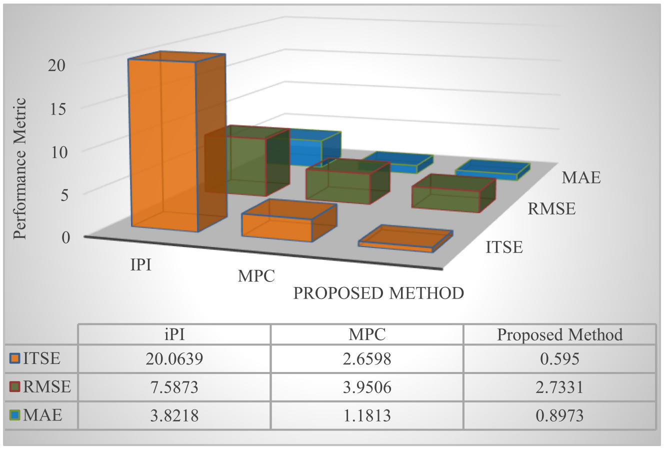
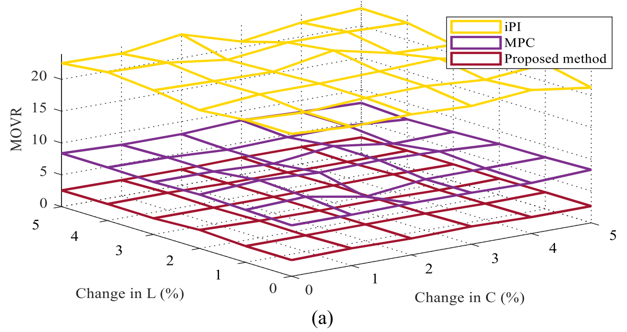


Fig. 10. Transient outcomes of the IPI controller (the CPL power variations, bus voltage, and inductor current are depicted with blue, red, and green curves, respectively).

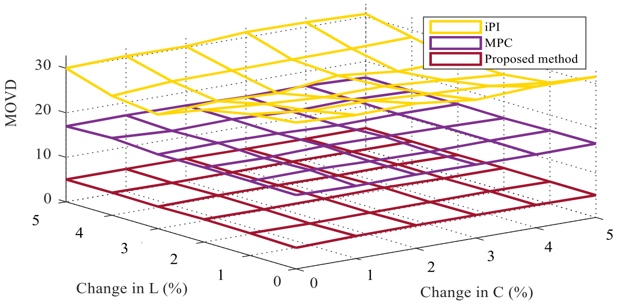


	iPI	MPC	Proposed Method
🟧 ITSE	20.0639	2.6598	0.595
🟫 RMSE	7.5873	3.9506	2.7331
🟦 MAE	3.8218	1.1813	0.8973

Fig. 11. Bar plot performance analysis of controllers.



(a)



(b)

Fig. 12. Robustness analysis against parameter changes. (a) MOVR. (b) MOVD.

Furthermore, some performance specifications of the buck converter are applied to evaluate the usefulness of the suggested method from various power engineering aspects. The sum of the integral-time-square error, root-mean-square error, and mean absolute error are presented in Fig. 11.

To ascertain the robustness of the suggested model-independent method against the plant parametric uncertainty, some critical parameters of the test converter (i.e., L and C) are varied from their nominal values to 5%, L is varied from 1 to 1.05 mH and C is varied from 1000 to 1050 μf. It is desired to provide the lower values of output voltage deviation against simultaneous variations of L and C. The values of maximum output voltage raise/drop (MOVR/MOVD) corresponding to the parametric changes are presented in Fig. 12(a) and (b). Since the DDPGiPI based on the SM observer obtained the lowest values of MOVR/MOVD, the suggested control strategy has less sensitivity to plant parametric variation than the other two considered controllers.

V. Conclusion

As it was evident from the experimental outcomes, one can infer that the transient and steady-state behavior of the dc–dc buck converter feeding nonideal CPL with time-varying nature was remarkably improved compared to the state-of-the-art methods. The objective was achieved by developing a novel adaptive model-independent controller, so-called DDPGiPI combined with an SM observer, for the concerned buck converter case study in an MG scenario. The suggested controller combines the DDPG into the iPI controller-based SM observer to reduce the observer estimation error and handle the destructive impedance instabilities. In this article, a real-time comparative study based on PHiL simulation was carried out to verify the supremacy of the suggested controller against the instant changes of CPL power over that of the other prevalent approaches. The experimental outcomes revealed that as compared to the iPI and MPC, the suggested method stabilizes the interface buck converter much quicker with less overshoot and the system outcomes were quickly returned to the steady-state.

For the future work, the proposed method can be implemented on the other types of dc–dc power converters (e.g., dc–dc boost power converter, and dc–dc buck-boost power converter) to enhance the performance and efficiency of the whole of the system.

References

[1] M. H. Amini, K. G. Boroojeni, T. Dragičević, A. Nejadpak, S. Iyengar, and F. Blaabjerg, “A comprehensive cloud-based real-time simulation framework for oblivious power routing in clusters of DC microgrids,” in *Proc. IEEE 2nd Int. Conf. DC Microgrids*, 2017, pp. 270–273.

[2] A. Kwasinski and C. N. Onwuchekwa, “Dynamic behavior and stabilization of DC microgrids with instantaneous constant-power loads,” *IEEE Trans. Power Electron.*, vol. 26, no. 3, pp. 822–834, Mar. 2011.

[3] R. Heydari, M. Gheisarnejad, M. H. Khooban, T. Dragicevic, and F. Blaabjerg, “Robust and fast voltage-source-converter (vsc) control for naval shipboard microgrids,” *IEEE Trans. Power Electron.*, vol. 34, no. 9, pp. 8299–8303, Sep. 2019.

[4] M. H. Khooban, M. Gheisarnejad, H. Farsizadeh, A. Masoudian, and J. Boudjadar, “A new intelligent hybrid control approach for dc–dc converters in zero-emission ferry ships,” *IEEE Trans. Power Electron.*, vol. 35, no. 6, pp. 5832–5841, Jun. 2020.

[5] G. Lin, Y. Li, J. Liu, and C. Li, “Resonance analysis and active damping strategy for shipboard DC zonal distribution network,” *Int. J. Elect. Power Energy Syst.*, vol. 105, pp. 612–621, 2019.

[6] L. He *et al.*, “A flexible power control strategy for hybrid AC/DC zones of shipboard power system with distributed energy storages,” *IEEE Trans. Ind. Informat.*, vol. 14, no. 12, pp. 5496–5508, Dec. 2018.

[7] P. Magne, B. Nahid-Mobarakeh, and S. Pierfederici, “Active stabilization of dc microgrids without remote sensors for more electric aircraft,” *IEEE Trans. Ind Appl.*, vol. 49, no. 5, pp. 2352–2360, Sep./Oct. 2013.

[8] L. Herrera, W. Zhang, and J. Wang, “Stability analysis and controller design of DC microgrids with constant power loads,” *IEEE Trans. Smart Grid*, vol. 8, no. 2, pp. 881–888, Mar. 2017.

[9] H. C. Foong, Y. Zheng, Y. K. Tan, and M. T. Tan, “Fast-transient integrated digital DC-DC converter with predictive and feedforward control,” *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 7, pp. 1567–1576, Jul. 2012.

[10] M. Cespedes, L. Xing, and J. Sun, “Constant-power load system stabilization by passive damping,” *IEEE Trans. Power Electron.*, vol. 26, no. 7, pp. 1832–1836, Jul. 2011.

[11] A. Emadi, A. Khaligh, C. H. Rivetta, and G. A. Williamson, "Constant power loads and negative impedance instability in automotive systems: definition, modeling, stability, and control of power electronic converters and motor drives," *IEEE Trans. Veh. Technol.*, vol. 55, no. 4, pp. 1112–1125, Jul. 2006.

[12] Q. Xu, C. Zhang, C. Wen, and P. Wang, "A novel composite nonlinear controller for stabilization of constant power load in DC microgrid," *IEEE Transt. Smart Grid*, vol. 10, no. 1, pp. 752–761, Jan. 2019.

[13] S. Singh, D. Fulwani, and V. Kumar, "Robust sliding-mode control of dc–dc boost converter feeding a constant power load," *IET Power Electron.*, vol. 8, pp. 1230–1237, 2015.

[14] M. H. Khooban, N. Vafamand, and J. Boudjadar, "Tracking control for hydrogen fuel cell systems in zero-emission ferry ships," *Complexity*, vol. 2019, 2019, Art. no. 5358316.

[15] J. Yang, H. Cui, S. Li, and A. Zolotas, "Optimized active disturbance rejection control for DC-DC buck converters with uncertainties using a reduced-order GPI observer," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 65, no. 2, pp. 832–841, Feb. 2018.

[16] M. Asghar, A. Khattak, and M. M. Rafiq, "Comparison of integer and fractional order robust controllers for dc–dc converter feeding constant power load in a DC microgrid," *Sustain. Energy, Grids Netw.*, vol. 12, pp. 1–9, 2017.

[17] N. Barabanov, R. Ortega, R. Griñó, and B. Polyak, "On existence and stability of equilibria of linear time-invariant systems with constant power loads," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 63, no. 1, pp. 114–121, Jan. 2016.

[18] N. Vafamand, M. H. Khooban, T. Dragicevic, F. Blaabjerg, and J. Boudjadar, "Robust non-fragile fuzzy control of uncertain DC microgrids feeding constant power loads," *IEEE Trans. Power Electron.*, vol. 34, no. 11, pp. 11300–11308, Nov. 2019.

[19] H. Abouaïssa and S. Chouraqui, "On the control of robot manipulator: A model-free approach," *J. Comput. Sci.*, vol. 31, pp. 6–16, 2019.

[20] M. Haddar, R. Chaari, S. C. Baslamisli, F. Chaari, and M. Haddar, "Intelligent PD controller design for active suspension system based on robust model-free control strategy," in *Proc. Inst. Mech. Eng., Part C, J. Mech. Eng. Sci.*, 2019, pp. 4863–4880.

[21] S. Han, H. Wang, and Y. Tian, "Model-free based adaptive nonsingular fast terminal sliding mode control with time-delay estimation for a 12 DOF multi-functional lower limb exoskeleton," *Adv. Eng. Softw.*, vol. 119, pp. 38–47, 2018.

[22] A. G. Haroun and Y. Y. Li, "A novel optimized hybrid fuzzy logic intelligent PID controller for an interconnected multi-area power system with physical constraints and boiler dynamics," *ISA Trans.*, vol. 71, pp. 364–379, 2017.

[23] X. Zhang, H. Wang, Y. Tian, L. Peyrodie, and X. Wang, "Model-free based neural network control with time-delay estimation for lower extremity exoskeleton," *Neurocomputing*, vol. 272, pp. 178–188, 2018.

[24] Z. Yan and Y. Xu, "Data-driven load frequency control for stochastic power systems: A deep reinforcement learning method with continuous action search," *IEEE Trans. Power Syst.*, vol. 34, no. 2, pp. 1653–1656, Mar. 2019.

[25] W. Shi, S. Song, C. Wu, and C. P. Chen, "Multi pseudo Q-learning-based deterministic policy gradient for tracking control of autonomous underwater vehicles," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 12, pp. 3534–3546, Dec. 2019.

[26] J. Zhu, Z. Wang, S. Guo, and C. Xu, "Hierarchical decision and control for continuous multitarget problem: Policy evaluation with action delay," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 2, pp. 464–473, Feb. 2019.

[27] C. Sampedro, H. Bavle, A. Rodriguez-Ramos, P. de la Puente, and P. Campoy, "Laser-based reactive navigation for multirotor aerial robots using deep reinforcement learning," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2018, pp. 1024–1031.

[28] Y. Wang, J. Sun, H. He, and C. Sun, "Deterministic policy gradient with integral compensator for robust quadrotor control," *IEEE Trans. Syst., Man, Cybern. Syst.*, 2019.

[29] T. P. Lillicrap *et al.*, "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.

[30] S. Singh, D. Fulwani, and V. Kumar, "Emulating DC constant power load: a robust sliding mode control approach," *Int. J. Electron.*, vol. 104, pp. 1447–1464, 2017.

[31] A. M. Rahimi and A. Emadi, "Active damping in dc–dc power electronic converters: A novel method to overcome the problems of constant power loads," *IEEE Trans. Ind. Electron.*, vol. 56, no. 5, pp. 1428–1439, May 2009.

[32] H. Farsizadeh, M. Gheisarnejad, M. Mosayebi, M. Rafiei, and M. H. Khooban, "An intelligent and fast controller for dc–dc converter feeding CPL in a DC microgrid," *IEEE Trans. Circuits Syst. I, Exp. Briefs*, vol. 67, no. 6, pp. 1104–1108, Jun. 2020.

[33] S. Yousefizadeh, N. Vafamand, J. D. Bendtsen, M. H. Khooban, and F. Blaabjerg, "Implementation of a cubature kalman filter for power estimation of non-ideal constant power loads in a DC Microgrid," *Wseas Trans. Power Syst.*, vol. 14, pp. 122–129, 2019.

[34] M. Gheisarnejad, H. Farsizadeh, M. -R. Tavana, and M. H. Khooban, "A novel deep learning controller for DC–DC buck-boost converters in wireless power transfer feeding CPLs," *IEEE Trans. Ind. Electron.*, 2020.

[35] T. MohammadRidha *et al.*, "Model free iPID control for glycemia regulation of type-1 diabetes," *IEEE Trans. Biomed. Eng.*, vol. 65, no. 1, pp. 199–206, Jan. 2018.

[36] M. M. Mardani, M. H. Khooban, A. Masoudian, and T. Dragičević, "Model predictive control of DC–DC converters to mitigate the effects of pulsed power loads in naval DC microgrids," *IEEE Trans. Ind. Electron.*, vol. 66, no. 7, pp. 5676–5685, Jul. 2019.

[37] M. Wu and D. D. C. Lu, "A novel stabilization method of LC input filter with constant power loads without load performance compromise in DC microgrids," *IEEE Trans. Ind. Electron.*, vol. 62, no. 7, pp. 4552–4562, Jul. 2015.

[38] J. Wu and Y. Lu, "Feedback linearization adaptive control for a buck converter with constant power loads," in *Proc. IEEE Int. Power Electron. Appl. Conf. Expo.*, 2018, pp. 1–6.

[39] C. N. Onwuchekwa and A. Kwasinski, "Analysis of boundary control for buck converters with instantaneous constant-power loads," *IEEE Trans. Power Electron.*, vol. 25, no. 8, pp. 2018–2032, Aug. 2010.

[40] S. Singh and D. Fulwani, "Voltage regulation and stabilization of dc–dc buck converter under constant power loading," in *Proc. IEEE Int. Conf. Power Electron., Drives Energy Syst.*, 2014, pp. 1–6.

[41] H. Ahmed, I. Salgado, and H. Ríos, "Robust synchronization of master-slave chaotic systems using approximate model: An experimental study," *ISA Trans.*, vol. 73, pp. 141–146, 2018.

[42] Y. Al Younes, A. Drak, H. Noura, A. Rabhi, and A. El Hajjaji, "Robust model-free control applied to a quadrotor UAV," *J. Intell. Robot. Syst.*, vol. 84, pp. 37–52, 2016.

[43] Z. Qiao, T. Shi, Y. Wang, Y. Yan, C. Xia, and X. He, "New sliding-mode observer for position sensorless control of permanent-magnet synchronous motor," *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 710–719, Feb. 2013.

[44] H. Zhang and J. Wang, "Adaptive sliding-mode observer design for a selective catalytic reduction system of ground-vehicle diesel engines," *IEEE/ASME Trans. Mechatronics*, vol. 21, no. 4, pp. 2027–2038, Aug. 2016.

[45] K. Zhao *et al.*, "Robust model-free nonsingular terminal sliding mode control for PMSM demagnetization fault," *IEEE Access*, vol. 7, pp. 15737–15748, 2019.

[46] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous navigation of UAVs in large-scale complex environments: A deep reinforcement learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 3, pp. 2124–2136, Mar. 2019.

[47] M. Zhu, X. Wang, and Y. Wang, "Human-like autonomous car-following model with deep reinforcement learning," *Transp. Res. Part C, Emerg. Technol.*, vol. 97, pp. 348–368, 2018.