WECC Composite Load Model Parameter Identification Using Evolutionary Deep Reinforcement Learning

Fankun Bu, *Graduate Student Member, IEEE*, Zixiao Ma, *Graduate Student Member, IEEE*,
AQ1 Yuxuan Yuan, *Graduate Student Member, IEEE*, and Zhaoyu Wang, *Member, IEEE*

***Abstract*—Due to the increasing penetration of distributed energy resources (DERs), the load composition in distribution grids has significantly changed. This inverter-based device has notably different behavior from traditional static and induction motor loads. To accurately represent the combination of static load, induction motor and the emerging inverter-based devices, the composite load model with distributed generation (CMPLDWG) has been developed by Western Electricity Coordinating Council (WECC). Due to the large number of parameters and model complexity, the CMPLDWG model brings new challenges to parameter identification, which is critical to power system studies. To address these challenges, in this paper, a cutting-edge approach inspired by the evolutionary deep reinforcement learning (EDRL) *with an intelligent exploration mechanism* is innovatively proposed to perform parameter identification for CMPLDWG. First, to extract parameters' contributions to dynamic power, parameter sensitivity analysis is conducted using a data-driven feature-wise kernelized Lasso (FWKL). Then, the EDRL with intelligent exploration, which can handle the natural high nonlinearity and nonconvexity of CMPLDWG, is employed to perform parameter identification. In the parameter identification process, the extracted parameter sensitivity weights are innovatively integrated into the EDRL with intelligent exploration to improve discovery efficiency. Finally, the proposed approach is validated using numerical experiments.**

***Index Terms*—WECC composite load model, parameter identification, evolutionary strategy, intelligent exploration.**

I. Introduction

PARAMETER identification of load models is essential to power systems studies, such as planning, operation and control [1]–[4]. Due to the increasing diversity of load types and the integration of distributed energy resources (DERs) [5], [6], parameter identification still remains a challenging problem to academic researchers and industrial practitioners. Measurement-based approaches are widely employed to perform parameter identification, where voltage and power measurements in fault-induced delayed-voltage-recovery (FIDVR) events are used to determine the parameters of given dynamic load models.

Previous works have mainly focused on identifying parameters of a composite load model which consists of a ZIP and an induction motor, where ZIP model is a combination of a constant-impedance load, a constant-current load and a constant-power load. In [2], based on trajectory sensitivities, the induction motor parameter number is reduced and only critical parameters are identified. The proposed approach is validated using real field measurements, and it is demonstrated that the approach can decrease identification time without losing the composite load model's dynamic characteristics. In [7], a robust time-varying parameter identification approach is proposed for synthesis load modeling. The synthetic load model includes time-varying ZIP, induction motor, and equivalent line impedance model. To achieve the goal of robustness enhancement, dynamic programming is used to detect voltage disturbances, and then a time-varying parameter identifier with a smaller iteration threshold is designed. In [8], a multi-modal long short-term memory deep learning method is employed to identify the time-varying parameters of the composite load model. In [9], a computationally efficient technique is utilized for identifying the composite load model parameters, by performing a similarity analysis of parameter sensitivity. The partial derivative of each parameter is employed to identify parameters with similar sensitivities, and Levenberg-Marquardt algorithm is used to solve the optimization problem. To improve computational efficiency, in [10], model parameter sensitivities are analyzed using eigenvalues of Hessian matrix, and the linear dependence between two parameters are then identified by examining the condition number of the Jacobian matrix. In [11], a robust time-varying parameter identification approach is developed for the composite load model. A batch-mode regression form is constructed to guarantee data redundancy, and the down-weighting coefficient for each measurement is calculated to reduce the impacts of outliers. To sum up, in previous works, both traditional optimization methods and modern learning-based approaches are employed to perform parameter identification of the composite load model which consists of a ZIP model and an induction motor model.

In recent years, as a large number of DERs are integrated into distribution systems, the composition of loads has changed significantly [12]–[14]. In order to accurately capture

AQ2 Manuscript received December 25, 2019; revised March 28, 2020 and June 1, 2020; accepted July 8, 2020. This work was supported by Advanced Grid Modeling Program at the U.S. Department of Energy Office of Electricity under Grant DE-OE0000875. Paper no. TSG-01923-2019. *(Corresponding author: Zhaoyu Wang.)*

The authors are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011 USA (e-mail: fbu@iastate.edu; wzy@iastate.edu).

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

Digital Object Identifier 10.1109/TSG.2020.3008730

the characteristics of this new type of load in modern power grids, the Western Electricity Coordinating Council (WECC) has developed a composite load model with distributed generation (CMPLDWG) [15]. Also, researchers have dedicated great efforts into studying this newly-proposed advanced load model. In [16], an easy-to-use tool is developed to generate dynamic load data to enhance utilities' planning studies. This tool can be adjusted to accommodate different customer types, various load components and characteristics. In [17], a generic modeling and open-source implementation of the WECC composite load model are presented, which reduces the gap between the WECC model and its further implementation. In [1], an approach is proposed for dynamic composite load modeling, where parameter dependency of the complex dynamic load model is analyzed and visualized using matrix decomposition and data clustering techniques. Meanwhile, the parameter identification performance is improved by adding a regularization term to include *a priori* parameter information into the objective function. However, the *a priori* parameter information is not generally available. In addition, the newly-approved aggregated distributed energy resources (DER_A) model in CMPLDWG has not been considered in [1]. In [18], the parameter identification process is divided into two steps: determining load composition and selecting a best-fit parameter vector candidate from Monte-Carlo simulations. To sum up, the primary disadvantages of previous WECC model parameter identification approaches are that they rely on prior knowledge of parameters or a comprehensive library of parameter candidates.

The CMPLDWG model contains 183 parameters, and the order of differential equations reaches 25. Therefore, the traditional optimization methods might not be able to handle the high-dimensional parameter vector and the severe nonconvexity of model structure. Considering this, we seek to perform parameter identification for CMPLDWG using an advanced learning-based approach with an embedded intelligent exploration (IE) mechanism, which is inspired by the evolutionary deep reinforcement learning (EDRL) technique. The proposed approach can efficiently avoid deceptive local optima and can handle the high-dimensional parameter vector [19], [20]. Specifically, first, the parameter sensitivity analysis (PSA) is conducted to obtain sensitivity weights reflecting contributions of parameters to dynamics, using feature-wise kernelized Lasso (FWKL), where Lasso denotes the least absolute shrinkage and selection operator. Then, the extracted parameter sensitivity weights are integrated into EDRL with IE to perform intelligent CMPLDWG parameter exploration by avoiding purely randomized or ineffective search. Parallelly, the EDRL with IE performs parameter exploitation using evolutionary strategy. Finally, the EDRL with IE guides the identifier to balance exploitation and exploration by designing time-varying dynamic weights assigned to the approximated performance gradient and novelty gradient.

The main innovations and contributions of our paper are summarized as follows: (1) To address the challenges of parameter identification caused by the nonlinearity of CMPLDWG model, we have designed a mechanism of intelligent exploration for encouraging the parameter identifier to escape from deceptive local optima. The exploration mechanism is achieved through time-varying dynamic weights which intelligently balance the exploitation and exploration. Most importantly, once the parameter identifier is stuck in a local optimum, it is stimulated to aggressively explore undiscovered parameter space. (2) The extracted CMPLDWG parameter sensitivity weights are innovatively integrated into the intelligent exploration to achieve directed and efficient parameter space discovery. By doing this, the parameter identifier can avoid purely randomized or inefficient exploration.

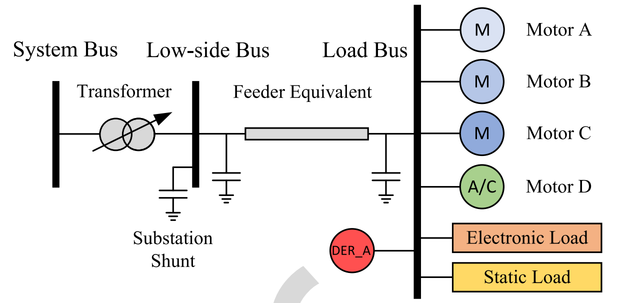


Fig. 1. The structure of the WECC composite load model with the distributed generation model of DER_A.

The rest of the paper is organized as follows: Section II introduces the CMPLDWG model and the overall framework of the proposed parameter identification approach. Section III proposes the method for parameter sensitivity analysis. Section IV describes the process of identifying CMPLDWG parameters using EDRL which is hybridized with IE. In Section V, case studies are conducted to validate the proposed approach and Section VI concludes the paper.

II. CMPLDWG Model and Overall Parameter Identification Framework

A. CMPLDWG Model

This paper focuses on the comprehensive WECC composite load model, which consists of three sections: substation, feeder and load, as illustrated in Fig. 1. The substation section is composed of a transformer model and a shunt capacitor model. The feeder section is denoted using an equivalent feeder model. The load section includes three three-phase induction motor models with different dynamic characteristics, one single-phase A/C performance-based motor model, an electronic load model, a static load model and a distributed generator model. In this paper, the distributed generator model is specified as the newly-approved DER_A model presented in [21]. Table I shows a list of WECC CMPLDWG model parameters of which detailed definitions can be found in [15], [21]. In addition, the mathematical state-space representations of CMPLDWG model are presented in [22].

B. Overall Framework of the Proposed Approach

The process of identifying unknown CMPLDWG parameters comes down to finding optimal parameters by reducing

TABLE I
PARAMETER LIST OF CMPLDWG MODEL

Component	Parameters
Substation, & Feeder	Load MVA Base, Bss, Rfdr, Xfdr, Fb, Xxf, TfixHS, TfixLS
Transformer	LTC, Tmin, Tmax, Step, Vmin, Vmax, Tdel, Ttap, Rcomp, Xcomp
Load Fraction	Fma, Fmb, Fmc, Fmd, Fel, Fzip, Fdg
Motor Type	MtpA, MtpB, MtpC, MtpD
Motor A	LfmA, RsA, LsA, LpA, LppA, TpoA, TppoA, HA, etrqA, DA, Vtr1A, Ttr1A, Ftr1A, Vrc1A, Trc1A, Vtr2A, Ttr2A, Ftr2A, Vrc2A, Trc2A
Motor B	LfmB, RsB, LsB, LpB, LppB, TpoB, TppoB, HB, etrqB, DB, Vtr1B, Ttr1B, Ftr1B, Vrc1B, Trc1B, Vtr2B, Ttr2B, Ftr2B, Vrc2B, Trc2B
Motor C	LfmC, RsC, LsC, LpC, LppC, TpoC, TppoC, HC, etrqC, DC, Vtr1C, Ttr1C, Ftr1C, Vrc1C, Trc1C, Vtr2C, Ttr2A, Ftr2C, Vrc2C, Trc2C
Motor D	LmfD, CompPF, Vstall, Rstall, Xstall, Tstall, Frst, Vrst, Trst, fuvr, vtr1, ttr1, vtr2, ttr2, Vc1off, Vc2off, Vc1on, Vc2on, Tth, Th1t, Th2t, tv, LFadj, Kp1, Np1, Kq1, Nq1, Kp2, Np2, Kq2, Nq2, Vbrk, CmpKpf, CmpKqf
Electronic Load	Pfel, Vd1, Vd2, Frcel
Static Load	Pfs, P1e, P1c, P2e, P2c, Pfreq, Q1e, Q1c, Q2e, Q2c, Qfreq
DER_A	Trv, dbd1, dbd2, Kqv, Vref0, Tp, Tiq, Ddn, Dup, fdbd1, fdbd2, femax, femin, Pmax, Pmin, dPmax, dPmin, Tpord, Imax, Vl0, Vl1, Vh0, Vh1, tvl0, tvl1, tvh0, tvh1, Vrfrac, fltrp, fhtrp, tfl, tfh, Tg, rrpwr, Tv, Kpg, Kig, Xe, Vpr, Iqh1, Iql1, Pflag, Frqflag, PQflag, typeflag

the following estimation residual [1]:

$$\min_{\boldsymbol{\theta}} l(\boldsymbol{Y}, \boldsymbol{\theta}, \boldsymbol{V}) = \min_{\boldsymbol{\theta}} \frac{1}{2}\Big(||\boldsymbol{Y} - f(\boldsymbol{\theta}, \boldsymbol{V})||_2^2\Big) \tag{1}$$

where, $\boldsymbol{Y}$ denotes active/reactive power measurement vector, $\boldsymbol{\theta}$ represents the vector of parameters to be identified, $\boldsymbol{V}$ denotes voltage measurement vector, l represents calculating the estimation residual, $||\cdot||_2$ is the l_2-norm, and $f(\cdot)$ denotes the mathematical representation of CMPLDWG model developed in [22]. More detailed variable definitions will be elaborated in Section III. To determine the optimal parameters for CMPLDWG, the EDRL approach with IE is developed in this paper. The components of parameter identification framework are illustrated in Fig. 2: *Component I - Sensitivity Analysis:* Sensitivity analysis evaluates the contributions of parameters to dynamic power measurements, and is based on the observation that the change of some parameters has an insignificant impact on power measurements. The high-order characteristic of induction motors and DER_A in CMPLDWG can significantly complicate PSA when using traditional methods. To address this challenge, an alternative data-driven PSA approach, FWKL, is proposed. The FWKL utilizes a set of randomly-generated CMPLDWG parameter vectors and corresponding calculated residuals to extract weights indicating parameter sensitivities. The PSA is formulated as a Lasso optimization problem given as

$$\min_{\boldsymbol{W}\in\mathbb{R}^d} \frac{1}{2}||\boldsymbol{e} - \boldsymbol{\Theta}^\top \boldsymbol{W}||_2^2 + \lambda||\boldsymbol{W}||_1, \tag{2}$$

where, $\boldsymbol{e}$ is the estimation residual vector, $\boldsymbol{\Theta}$ denotes the randomly-generated parameter vectors in a matrix form, $\boldsymbol{W} = [W_1, \ldots, W_d]^\top$ represents the parameter sensitivity weight vector, $||\cdot||_1$ is the l_1-norm and λ is the regularization parameter which is determined using grid search with cross-validation. Note that sensitivity analysis is a *one-off* work for each fault event. The extracted parameter sensitivity weight vector, $\boldsymbol{W}$, is passed to the novelty gradient estimator in each iteration whose number is denoted by t. *Component II - Parameter Vector Perturbator:* In each iteration, to perform evolution, a perturbator is designed to generate multiple mutated parameter vectors, $\boldsymbol{\theta}_t'$'s, using the identified parameter vector in the last iteration, $\boldsymbol{\theta}_t$, and random variance vector, $\boldsymbol{\epsilon}_t$. $\boldsymbol{\theta}_t'$'s and $\boldsymbol{\epsilon}_t$'s are then sent to a performance gradient estimator and a novelty gradient estimator to approximate performance and novelty gradients, respectively. *Component III - Performance Gradient Estimator:* This estimator achieves the function of exploitation of EDRL. Specifically, using $\boldsymbol{\theta}_t'$'s and $\boldsymbol{\epsilon}_t$'s generated by the parameter vector perturbator, the performance gradient estimator determines the direction in which $\boldsymbol{\theta}_t$ should move to improve expected reward. The performance gradient, $\Delta\boldsymbol{\theta}_t^{et}$, is then passed to a parameter updater. *Component IV - Novelty Gradient Estimator:* This component performs exploration by estimating the novelty gradient, $\Delta\boldsymbol{\theta}_t^{er}$, using the generated $\boldsymbol{\theta}_t'$'s and $\boldsymbol{\epsilon}_t$'s, and it also intelligently encourages the parameter identifier to explore unvisited parameter space. $\Delta\boldsymbol{\theta}_t^{er}$ is then sent to the parameter updater. *Component V - Parameter Updater:* To balance exploitation and exploration, the parameter updater assigns time-varying dynamic weights to the approximated performance and novelty gradients:

$$\Delta\boldsymbol{\theta}_t = \omega_t \Delta\boldsymbol{\theta}_t^{et} + (1 - \omega_t)\Delta\boldsymbol{\theta}_t^{er}, \tag{3}$$

where, ω_t denotes a dynamic weight. Then, $\boldsymbol{\theta}_{t+1}$ is calculated and added into the parameter vector archive to update the explored parameter space. *Component VI - Archive:* The archive collects the previously generated parameter vectors which are passed to the novelty gradient estimator for novelty evaluation. Component II to V compose the EDRL algorithm with IE. Since the construction of the parameter vector archive is straightforward, we will focus on elaborating the modules of sensitivity analysis and EDRL with IE in the next two sections.

III. PARAMETER SENSITIVITY ANALYSIS

PSA examines the sensitivity of dynamic power measurements with respect to load model parameters. In previous works, partial derivative of dynamic power to each parameter is calculated to conduct sensitivity analysis of induction motor parameters [9]. However, it becomes challenging to directly apply analytical approaches to calculate partial derivatives because of the high order and the complicated structure

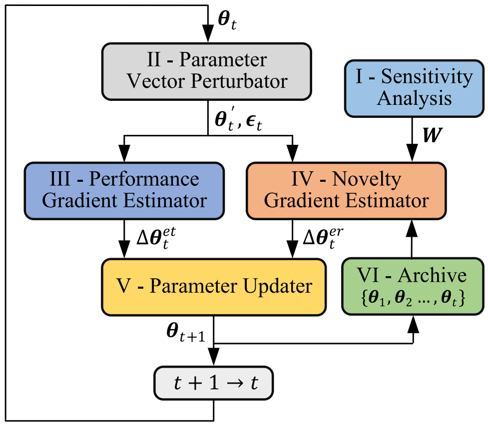


Fig. 2. The overall structure of the proposed parameter identification approach for CMPLDWG model.

of mathematical differential equations of the WECC composite load model. For example, the three-phase induction motor model in CMPLDWG is of 5th order and the DER_A model has ten state variables. Such a complex high-order nonlinear system can significantly complicate the calculation of partial derivatives. To address this challenge, we seek to employ a high-dimensional feature selection technique to evaluate the dependence of dynamic power on the CMPLDWG parameters [23]. Specifically, we use a data-driven FWKL instead of employing analytical derivatives [9].

Let $\boldsymbol{\theta}_i \in \mathbb{R}^d$ be a *randomly-generated* parameter vector and d be the number of parameters, therefore, the power residual corresponding to $\boldsymbol{\theta}_i$ can be calculated as

$$e_i = ||f(\boldsymbol{\theta}_i, \boldsymbol{V}) - \boldsymbol{Y}||_2, \tag{4}$$

where, $\boldsymbol{V} \in \mathbb{R}^K$ is a vector of voltage measurements, K denotes the total number of measurement points, $\boldsymbol{Y} = [\boldsymbol{P}^\mathsf{T}, \boldsymbol{Q}^\mathsf{T}]^\mathsf{T}$, $\boldsymbol{P} \in \mathbb{R}^K$ and $\boldsymbol{Q} \in \mathbb{R}^K$ represent the vector of recorded active power and reactive measurements, respectively. Also, T denotes the transpose. With a large number of generated $\boldsymbol{\theta}_i$'s, we can obtain n independent and identically distributed (i.i.d.) sample and residual pairs:

$$\{(\boldsymbol{\theta}_i, e_i), i = 1, \dots, n\}. \tag{5}$$

To perform supervised feature selection, first, we represent the original parameter vectors and corresponding residuals in a matrix format as

$$\boldsymbol{\Theta} = [\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_n] \in \mathbb{R}^{d \times n}, \tag{6a}$$

$$\boldsymbol{e} = [e_1, \dots, e_n]^\mathsf{T} \in \mathbb{R}^n. \tag{6b}$$

Then, PSA is formulated as a Lasso optimization problem formulated in (2) which works well for linear regression. However, the nonlinear dependency in our specific problem hinders its application. Therefore, we employ the feature-wise nonlinear Lasso to solve our problem and the key idea is to apply a nonlinear transformation in a feature-wise manner. Specifically, the generated parameter matrix, $\boldsymbol{\Theta}$, is represented in a feature-wise manner:

$$\boldsymbol{\Theta} = \left[\boldsymbol{\beta}_1, \dots, \boldsymbol{\beta}_d\right]^\mathsf{T} \in \mathbb{R}^{d \times n}, \tag{7}$$

where, $\boldsymbol{\beta}_k = [\theta_{k,1}, \dots, \theta_{k,n}]^\mathsf{T} \in \mathbb{R}^n$ is a vector denoting the k-th feature for all samples. To capture the nonlinear dependency of e on $\boldsymbol{\theta}$, dynamic power residual and parameter vector are transformed by a nonlinear function $\varphi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^p$. Then, the Lasso optimization problem given in the objective function (2) in the transformed space is reformulated as

$$\min_{\boldsymbol{W} \in \mathbb{R}^d} \frac{1}{2}||\varphi(\boldsymbol{e}) - \sum_{k=1}^{d} W_k \varphi\big(\boldsymbol{\beta}_k\big)||_2^2 + \lambda||\boldsymbol{W}||_1. \tag{8}$$

Although the objective function (8) can capture nonlinear dependency, there is no constraint for $W_k, k = 1, \dots, d$, and the same transformation function $\varphi(\cdot)$ for $\boldsymbol{e}$ and $\boldsymbol{\beta}_k$ limits the flexibility of capturing nonlinearity. To solve this, we seek to employ a revised FWKL to perform feature selection [23], and the revised objective function is formulated as

$$\min_{\boldsymbol{W} \in \mathbb{R}^d} \frac{1}{2}||\overline{\boldsymbol{U}} - \sum_{k=1}^{d} W_k \overline{\boldsymbol{V}}^{(k)}||_{Frob}^2 + \lambda||\boldsymbol{W}||_1, \tag{9a}$$

$$s.t. \quad W_1, \dots, W_d \geq 0. \tag{9b}$$

where, $|| \cdot ||_{Frob}$ denotes the Frobenious norm, $\overline{\boldsymbol{U}} = \boldsymbol{\Gamma} \boldsymbol{U} \boldsymbol{\Gamma}$ and $\overline{\boldsymbol{V}}^{(k)} = \boldsymbol{\Gamma} \boldsymbol{V}^{(k)} \boldsymbol{\Gamma}$ are centered Gram matrices, $\boldsymbol{U}_{i,j} = \boldsymbol{U}(e_i, e_j)$ and $\boldsymbol{V}_{i,j}^{(k)} = \boldsymbol{V}(\boldsymbol{\theta}_{k,i}, \boldsymbol{\theta}_{k,j})$ are Gram matrices, $\boldsymbol{U}(e, e')$ and $\boldsymbol{V}(\boldsymbol{\theta}, \boldsymbol{\theta}')$ are kernel functions, $\boldsymbol{\Gamma} = \boldsymbol{I}_n - \frac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n^\mathsf{T}$ denotes the centering matrix, $\boldsymbol{I}_n$ represents the n-dimensional identity matrix, and $\boldsymbol{1}_n$ denotes the n-dimensional vector with all ones. For the two kernel functions $\boldsymbol{U}(\cdot)$ and $\boldsymbol{V}(\cdot)$, we employ the Gaussian kernel which is formulated as

$$K\big(\boldsymbol{x}, \boldsymbol{x}'\big) = exp\left(-\frac{\big(\boldsymbol{x} - \boldsymbol{x}'\big)^2}{2\sigma_{\boldsymbol{x}}^2}\right), \tag{10}$$

where, $\sigma_{\boldsymbol{x}}$ is the Gaussian kernel width.

In the objective function (9a), the decoupling between $\boldsymbol{U}(\cdot)$ and $\boldsymbol{V}(\cdot)$ provides more flexibility compared with the objective function (8). In addition, the non-negativity constraint in (9b) fits the specific application in our problem, since negative sensitivity parameter weights do not have practical interpretability. Intuitively, problem (9) tends to find non-redundant parameters with significant contributions to power residual, and equivalently, to dynamic power. Also, for two strongly dependent features, either of their sensitivity weights tends to be eliminated. The parameter sensitivity weight vector, $\boldsymbol{W}$, is then integrated into the parameter identification algorithm to accelerates the learning process, which will be presented in Section IV.

IV. Parameter Identification Using the EDRL With IE

As stated in previous sections, the severe nonlinearity, high nonconvexity and the large number of parameters bring significant challenges to perform parameter identification for the CMPLDWG model when using existing approaches. This motivates us to tackle this challenge utilizing the EDRL with IE, which is recently demonstrated to be able to perform well on high-dimensional optimization tasks [19], [24]. The basic idea of performing optimization tasks using evolution

TABLE II
NUMERICAL INTERVAL OF LOAD PARAMETERS

Parameter	LB	UB	Parameter	LB	UB	Parameter	LB	UB	Parameter	LB	UB
Motor A			etrqB	1	4	Np2	1.6	4.8	Tp	0.01	0.04
TpoA	0.046	0.184	DB	0.5	2.0	Nq1	1	4	Tiq	0.01	0.04
TppoA	0.001	0.004	Motor C			Nq2	1.25	5	Tpord	2.5	10
LpA	0.05	0.20	TpoC	0.05	0.20	CmpKpf	0	2	Kpg	50	200
LppA	0.042	0.168	TppoC	0.0013	0.0052	CmpKqf	-6.6	-1.65	Kig	5	20
LsA	0.9	3.6	LpC	0.08	0.32	Electronic Load			Tg	0.01	0.04
RsA	0.02	0.08	LppC	0.06	0.24	Frcel	0	0.375	Tv	0.01	0.04
HA	0.05	0.20	LsC	0.9	3.6	Static Load			Xe	0.125	0.500
etrqA	0.5	2.0	RsC	0.015	0.06	P1c	0	0.4	Load Fraction		
DA	0.5	2.0	HC	0.1	0.4	P2c	0	0.6	Fma	0	0.5
Motor B			etrqC	1	4	Q1c	0	0.4	Fmb	0	0.5
TpoB	0.05	0.20	DC	0.5	2.0	Q2c	0	0.6	Fmc	0	0.5
TppoB	0.0013	0.0052	Motor D			Pfreq	-0.2	0.2	Fmd	0	0.5
LpB	0.08	0.32	Kp1	-1	1	Qfreq	-2	-0.5	Fel	0	0.5
LppB	0.06	0.24	Kp2	6	24	DER_A			Fzip	0	0.5
LsB	0.9	3.6	Kq1	3	12	Trv	0.01	0.04	Fdg	-0.5	0
RsB	0.015	0.06	Kq2	5.5	22	Trf	0.015	0.06			
HB	0.5	2.0	Np1	0.5	2	Kqv	0.5	2.0			

strategy is: During each iteration, a population of parameter vectors is perturbed based on one selected parameter vector among a meta-population, and then, these mutated vectors are recombined to update the selected ancestor vector. In this paper, the EDRL is also hybridized with IE to improve exploration. Compared with traditional random and blind search strategy, the IE module achieves efficient and directed exploration, which can efficiently assist EDRL to escape from local optima. The detailed steps are described as follows:

Step I - Initialization: The first step is to initialize M random parameter vectors which will be updated in each iteration. Note that only one vector is probabilistically selected to update in each iteration. The initialized M vectors are denoted as $S = \{\boldsymbol{\theta}_1^1, \ldots, \boldsymbol{\theta}_1^M\}$, where t denotes the number of iteration. The objective of constructing a meta-population is to enhance additional diversity. M and the tuning parameters in the remaining sections are determined using grid search with cross-validation which is a general hyperparameter optimization technique.

Step II - Sampling: In each iteration t, we probabilistically determine which parameter vector among the M meta-population to be updated based on parameter vectors' novelties. The novelty is evaluated in terms of Euclidean distances from a vector to the vectors in the newest archive. Specifically, first, the originality of each parameter vector in S, $\boldsymbol{\theta}_t^k$, conditioned on current parameter vector archive, $\boldsymbol{A}$, is evaluated as

$$O_t^k = o\left(\boldsymbol{\theta}_t^k, \boldsymbol{W}, \boldsymbol{A}\right) = \frac{1}{|C|} \sum_{j \in C} ||\boldsymbol{W}.* \left(\boldsymbol{\theta}_t^k - \boldsymbol{\theta}_j\right)||_2, \quad (11)$$

where, $1 \leq k \leq M$, $C = kNN(\boldsymbol{\theta}_t^k, \boldsymbol{A}) = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{N'}\}$, *kNN* denotes *k*-nearest neighbors algorithm, and .* denotes the element-wise multiplication operation. The purpose of *kNN* is to select *representative* parameter vectors in $\boldsymbol{A}$ for evaluating the novelty of $\boldsymbol{\theta}_t^k$. Intuitively, a small k can introduce higher distance variance, while a large k means higher computational cost. In our paper, we have conducted numerical experiments to determine the optimal k value which is sufficient for evaluating the novelty of a newly explored parameter vector while avoiding high computational time. The introduction of $\boldsymbol{W}$, which is obtained from PSA, aims to revise Euclidean distances between vectors. This revision is based on the consideration that parameters with different sensitivity weights have different contributions to vector novelty. Then, for each parameter vector in S, the novelty score which determines the probability of being selected to be updated is calculated as

$$P_t^k = \frac{O_t^k}{\sum_{j=1}^{M} O_t^j}. \quad (12)$$

P_t^k tells us that selecting the parameter vectors with high novelty scores can achieve directed or guided exploration.

Step III - Variation: In this step, variation is performed on the selected parameter vector in Step II, $\boldsymbol{\theta}_t^k$, to generate multiple workers. The function of these workers is explained as follows: First, EDRL produces parameter vectors in the neighborhood of $\boldsymbol{\theta}_t^k$, and then $\boldsymbol{\theta}_t^k$ is updated by following the direction determined by the population of the produced parameter vector workers. To obtain N workers, Gaussian noise is applied to $\boldsymbol{\theta}_t^k$ as follows

$$\boldsymbol{\theta}_t^{i,k} = \boldsymbol{\theta}_t^k + \sigma \boldsymbol{\epsilon}_t^i \quad i = 1, \ldots, N, \quad (13)$$

where, σ is a fixed noise standard deviation, $\boldsymbol{\epsilon}_t^i \sim \mathcal{N}(0, \boldsymbol{I})$ and $\boldsymbol{I}$ is an N-dimensional identity matrix.

Step IV - Gradient Estimation: In this step, the performance and novelty gradients determined by the meta-population of generated vectors in Step III are approximated. For each mutated parameter vector, $\boldsymbol{\theta}_t^{i,k}$, its fitness can be evaluated via calculating the difference between the estimated dynamic power and the real dynamic power. First, the power residual caused by the mismatch between estimated parameters and real parameters, $e_t^{i,k}$, is calculated by substituting $\boldsymbol{\theta}_t^{i,k}$ into (4). Then, the reward is obtained by inversing $e_t^{i,k}$:

$$R_t^{i,k} = r\left(\boldsymbol{\theta}_t^{i,k}, \boldsymbol{V}, \boldsymbol{Y}\right) = \frac{1}{e_t^{i,k}} \quad i = 1, \ldots, N. \tag{14}$$

Equation (14) indicates that as the residual decreases the reward increases. Thus, the performance gradient of $\boldsymbol{\theta}_t^k$ is *approximated* via taking a sum of the sampled parameter vector perturbations weighted by the reward:

$$\Delta\boldsymbol{\theta}_t^{et,k} \approx \alpha \frac{1}{N\sigma} \sum_{i=1}^{N} R_t^{i,k} \boldsymbol{\epsilon}_t^i, \tag{15}$$

where, α is a learning rate. In (15), $\Delta\boldsymbol{\theta}_t^{et,k}$ indicates a stochastic reward experienced over a full iteration of multiple worker interactions, which means the performance gradient relies on multiple workers and this can effectively avoid the high variance brought by a certain single mutated vector. Note that the calculated reward, $R_t^{i,k}$, is normalized through 1 to N before performing the gradient approximation in (15).

For the novelty gradient, first, the novelty with respect to each perturbed vector, $O_t^{i,k}$, is calculated using (11). Then, the novelty gradient of $\boldsymbol{\theta}_t^k$ is approximated as

$$\Delta\boldsymbol{\theta}_t^{er,k} \approx \alpha \frac{1}{N\sigma} \sum_{i=1}^{N} O_t^{i,k} \boldsymbol{\epsilon}_t^i. \tag{16}$$

Similar with $R_t^{i,k}$, $O_t^{i,k}$ is normalized before computing the novelty gradient. Intuitively, $\Delta\boldsymbol{\theta}_t^{er,k}$ indicates the direction which the parameter identifier should follow to increase the average originality of parameter vector distribution.

Step V - Gradient Combination: Using the computed performance and novelty gradients with respect to $\boldsymbol{\theta}_t^k$, we can balance exploitation and exploration by introducing a time-varying dynamic weight, ω_t. Thus, the overall gradient based on which $\boldsymbol{\theta}_t^k$ should be updated is computed as follows:

$$\Delta\boldsymbol{\theta}_t^k = \omega_t \Delta\boldsymbol{\theta}_t^{et,k} + (1 - \omega_t)\Delta\boldsymbol{\theta}_t^{er,k}. \tag{17}$$

Intuitively, the algorithm follows the approximated gradient in parameter-space towards directions that both exhibit novel behaviors and achieve high rewards. A large ω_t tends to encourage $\boldsymbol{\theta}_t^k$ to follow the performance gradient and restrain it to follow the novelty gradient. In comparison, a small ω_t tends to aggressively guide $\boldsymbol{\theta}_t^k$ to mutate to unseen parameter space and hold back exploitation.

Step VI - Updating: After obtaining $\Delta\boldsymbol{\theta}_t^k$, the updating of $\boldsymbol{\theta}_t^k$ is expressed as follows:

$$\boldsymbol{\theta}_{t+1}^k = \boldsymbol{\theta}_t^k + \Delta\boldsymbol{\theta}_t^k. \tag{18}$$

$\boldsymbol{\theta}_{t+1}^k$ is then added into the archive $\boldsymbol{A}$ for updating the pre-existing vector landscape. As more learned parameter vectors are saved into $\boldsymbol{A}$, the base for evaluating future parameter vectors' novelty changes and stimulates the algorithm to discover unexplored parameter space.

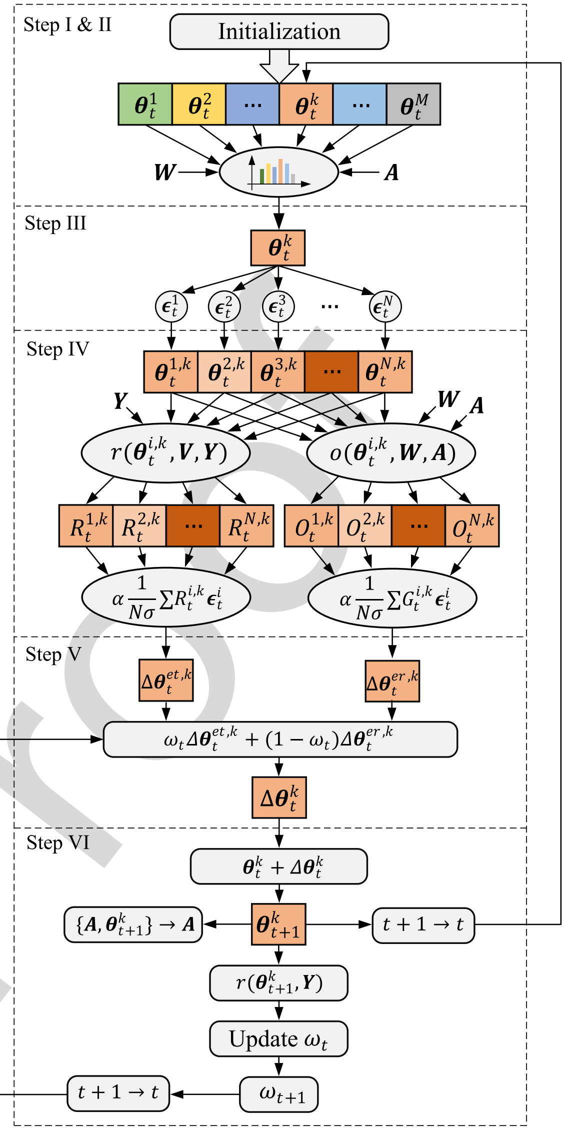


Fig. 3. Detailed structure of the EDRL with an intelligent exploration mechanism.

Algorithm 1 Updating ω_t

if $R_{t+1}^k > R_b^t$ **then**
  if $\omega_t \neq 0$ **then**
    $\omega_{t+1} \leftarrow min(1, \omega_t + \Delta\omega)$; $C_b^{t+1} \leftarrow 0$;
    $R_b^{t+1} \leftarrow R_{t+1}^k$;
  else
    $\omega_{t+1} \leftarrow 1$; $C_b^{t+1} \leftarrow 0$; $R_b^{t+1} \leftarrow R_{t+1}^k$;
  end if
else
  $C_b^{t+1} \leftarrow C_b^t + 1$;
end if
if $C_b^t > C_{set}$ **then**
  $\omega_{t+1} \leftarrow max(0, \omega_t - \Delta\omega)$; $C_b^{t+1} \leftarrow 0$;
end if

In addition to updating $\boldsymbol{\theta}_t^k$ and $\boldsymbol{A}$ in each iteration, the dynamic weight, ω_t, should also be updated for avoiding local optima. To do this, first, the latest reward, R_{t+1}^k, which is brought by $\boldsymbol{\theta}_{t+1}^k$, is calculated. We also define a "drag hand", R_b^t, to record the best reward among historical rewards. Then, the dynamic weight in (17), ω_t, is updated using Algorithm 1, where, $\Delta\omega$ denotes the weight updating rate, and C_b^t counts the number of rewards that are less than R_b^t in succession. C_{set} is a threshold which determines the frequency of updating ω_t when the parameter vector is stuck in a local optimum. Also, C_b^t and R_b^t are updated in each iteration, as presented in Algorithm 1. Note that Step II to VI constitute the entire operation in each iteration t.

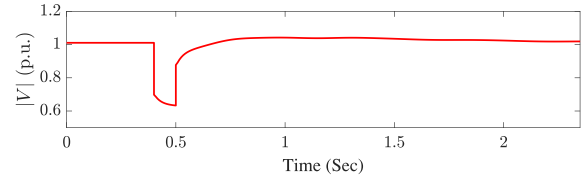
V. Case Study

In this section, the proposed parameter sensitivity analysis and parameter identification algorithms are validated using numerical experiments. Before performing verification, we firstly screen out the CMPLDWG parameters that are necessary to be identified. This screening is based on the consideration that CMPLDWG contains multiple types of parameters, of which some parameters can be determined by field measurements and engineering judgement. Specifically, the transformer impedance, substation shunt capacitive susceptance, feeder impedance and capacitive susceptance can be accurately calculated using transformer, capacitor and feeder parameters [25], [26]. For the stalling and restarting of induction motors, engineering judgement can be leveraged to estimate the settings [15], [27]. This is based on the observation that the stalling or restarting of a large number of induction motors can cause abrupt current, voltage and power changes [28], [29], which can be further corroborated in [1]. Also, the tripping of a large number of induction motors can cause sudden current decrease, power decrease and voltage increase. Excluding the parameters which can be accurately calculated using the electric power grid modeling technique can significantly reduce the complexity of parameter identification process. On the other hand, indistinguishably identifying all CMPLDWG parameters can pose an unnecessary computational burden on the proposed parameter identification algorithm. In our problem, 61 CMPLDWG parameters are screened out for parameter identification, as shown in Table III, and the remaining parameters are set with default values.

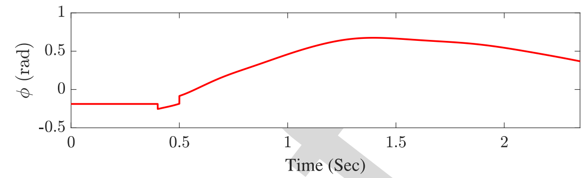
In this case study, the Power System Simulator for Engineering (PSS/E) and the ACTIVSg500 test case are employed to generate voltage and power measurements for parameter identification [30]. The fault-induced voltage-recovery curves are shown in Fig. 4. MATLAB is used to execute the processes of parameter sensitivity analysis and parameter identification. The case study is conducted on a standard PC with an Intel Xeon CPU running at 3.70 GHz and with 32.0 GB of RAM.

A. Parameter Sensitivity Identification

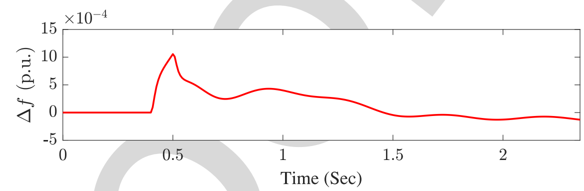
To fully extract the sensitivity weights hidden in the randomly-generated parameter samples and corresponding power residuals, first, we have created a comprehensive library containing 40,000 parameter vector and residual pairs which are divided into two sections, training dataset and test dataset, for cross-validation. Note that the dataset size is determined based on our numerical experiment result that once the dataset size exceeds 16,000, the FWKL gives us stable extracted parameter weights for different sets of the randomly selected parameter vector and residual pairs. Generating each pair of the parameter vector and the corresponding residual takes about 0.3 seconds. Then, the tuning parameters of FWKL are determined using grid search with cross-validation based on the training and test datasets [31]. Finally, the FWKL algorithm is applied to the entire dataset to conduct parameter sensitivity analysis. Based on our sensitivity analysis result, the load fraction parameters, the synchronous and subtransient reactances of three-phase induction motors, and the exponential load torque coefficients of three-phase induction motors have a significant effect on the load dynamics in the fault event specified in Fig. 4, as shown in Fig. 5. The remaining parameters have small or no effect on the dynamic procedure. It should be noted that the values of parameter sensitivity weights *change* according to specific dynamic events since the weight vector in (9) partially depends on the voltage and power measurements, which are determined by specific fault cases. Therefore, PSA should be conducted on a case-by-case basis to obtain more accurate parameter sensitivity weights for specific fault events.



(a) Load bus voltage magnitude.



(b) Load bus voltage angle.



(c) Load bus frequency.

Fig. 4. Fault-induced voltage-recovery curves at the load bus.

B. Parameter Identification

The extracted parameter sensitivity weights are integrated into EDRL algorithm with IE to perform parameter identification using given voltage and power measurements. There are only a couple of published technical reports involved with

TABLE III
REAL AND IDENTIFIED CMPLDWG PARAMETERS

Parameter	Real	Identified	Parameter	Real	Identified	Parameter	Real	Identified	Parameter	Real	Identified
Motor A			etrqB	2	2.4816	Np2	3.2	4.4470	Tp	0.02	0.0207
TpoA	0.092	0.0906	DB	1	1.2146	Nq1	2	1.6632	Tiq	0.02	0.0153
TppoA	0.002	0.0024	Motor C			Nq2	2.5	2.5239	Tpord	5	4.0030
LpA	0.1	0.1037	TpoC	0.1	0.0941	CmpKpf	1	0.5000	Kpg	100	68.3279
LppA	0.083	0.0495	TppoC	0.0026	0.0034	CmpKqf	-3.3	-4.2400	Kig	10	9.9675
LsA	1.8	1.8637	LpC	0.16	0.1268	Electronic Load			Tg	0.02	0.0156
RsA	0.04	0.0275	LppC	0.12	0.1064	Frcel	0.25	0.1551	Tv	0.02	0.0163
HA	0.1	0.1188	LsC	1.8	1.7535	Static Load			Xe	0.25	0.2239
etrqA	1	0.8368	RsC	0.03	0.0286	P1c	0.2	0.1953	Load Fraction		
DA	1	0.9661	HC	0.2	0.2839	P2c	0.3	0.2094	Fma	0.2	0.1969
Motor B			etrqC	2	2.3741	Q1c	0.2	0.1588	Fmb	0.3	0.4393
TpoB	0.1	0.0883	DC	1	1.0687	Q2c	0.3	0.1727	Fmc	0.3	0.3113
TppoB	0.0026	0.0034	Motor D			Pfreq	0	-0.0942	Fmd	0.1	0.1300
LpB	0.16	0.1094	Kp1	0	0.8636	Qfreq	-1	-0.8593	Fel	0.2	0.1804
LppB	0.12	0.1797	Kp2	12	11.6751	DER_A			Fzip	0.1	0.1774
LsB	1.8	2.0663	Kq1	6	8.0773	Trv	0.02	0.0262	Fdg	-0.2	-0.2053
RsB	0.03	0.0302	Kq2	11	10.9500	Trf	0.03	0.0221			
HB	1	1.4290	Np1	1	1.3602	Kqv	1	1.4408			

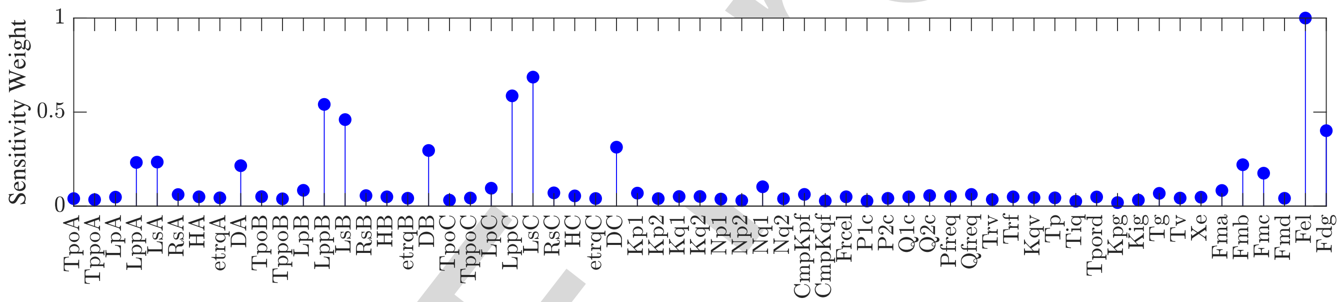
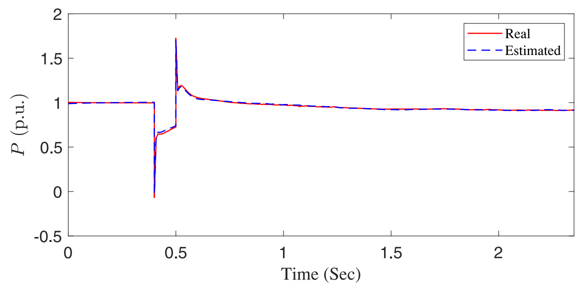


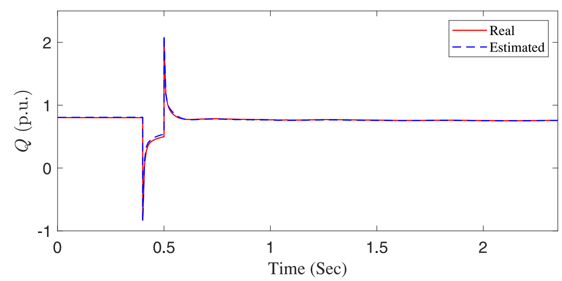
Fig. 5. Sensitivity weights of WECC composite load model parameters.

WECC model parameter settings. In this paper, the numerical intervals of parameters for randomly selecting initial values are determined based on [32], [33], along with our experience on deriving detailed mathematical representation of WECC composite load model [22]. The numerical intervals are presented in Table II, where, LB denotes lower bound and UB denotes upper bound. Table III shows the real and corresponding identified parameter values of CMPLDWG. As can be observed, the EDRL with IE can give us satisfying identified parameters. The identification accuracy is further corroborated by Fig. 6, in which, the estimated active and reactive power curves can closely fit the actual curves. While our approach is not designed for online parameter identification, it is of importance to examine the computational time. In our case studies, each iteration takes about 2 seconds.

It is also of significance to examine the collected best reward R_b^t and dynamic weight ω_t in each iteration, which are shown in Fig. 7 and 8, respectively. In Fig. 7, the loss corresponding to the collected best reward, e_b^t, is also shown for examining parameter identification performance. It can be seen that during Iteration 1 to 1226, the proposed parameter identification approach simultaneously performs exploitation and exploration, and the best reward increases continuously, as shown in Fig. 7. The corresponding learning process in this iteration range can be confirmed in Fig. 8, in which ω_t is firstly initialized as 0, once it stays invariant for 10 continuous iterations (C_{set}), it is decreased in a step size of 0.05 ($\Delta\omega$) to force the parameter identifier to follow more closely with novelty gradient. Once an unseen better reward occurs, ω_t gradually increases to 1 to encourage the identifier to act following the approximated performance gradient. During Iteration 1 to 1226, although ω_t alternatively decreases and increases, it does not reach 0. From Iteration 1227 to 1717, the parameter identifier is stuck in a local optimum and the best reward stays invariant, as shown in Fig. 7. During this iteration range, first, ω_t is designed to gradually decrease to 0, which means the identifier is stimulated to explore more aggressively in the unseen parameter space, as presented in Section IV. This is verified by the variation of dynamic weight ω_t, as shown in Fig. 8, where, from Iteration 1227 to 1717, ω_t decreases to 0 and keep unchanged, which means the identifier completely inhibits the performance gradient. At Iteration



(a) Active power



(b) Reactive power

Fig. 6. The real power curves and the estimated power curves using the identified parameters.

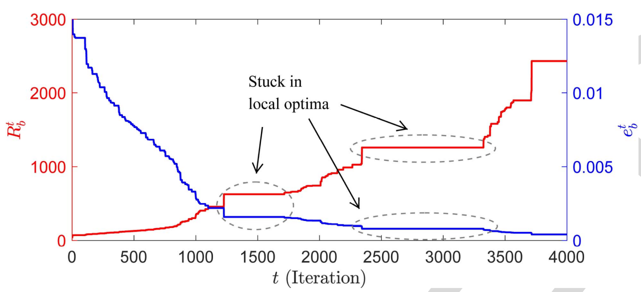


Fig. 7. The best reward and corresponding loss.

1718, the identifier discovers a parameter vector which can give higher reward than any of the previous best rewards. As expected, ω_t immediately jumps to 1 to avoid possible sliding out from the newly explored optimum with higher reward, due to novelty exploration inertia. From Iteration 1718 to 2342, the identifier simultaneously performs exploitation and exploration as shown in Fig. 7, accordingly, ω_t varies in the range of a non-zero value to 1, as shown in Fig. 8. This is similar to the process which occurs in the range of Iteration 1 to 1226. Similar with the range of Iteration 1227 to 1717, in the range of Iteration 2343 to 3324, ω_t decreases to 0 and R_b^t stays invariant, as shown in Fig. 8 and 7, respectively. At Iteration 3325, ω_t jumps to 1 to force the identifier immediately perform exploitation, which is similar at Iteration 1718, as shown in Fig. 8. Also, the best reward starts to increase at Iteration 3325, as shown in Fig. 7. The aforementioned cyclic process continues to pursue better rewards as the number of iterations increases, as shown in Fig. 7 and 8.

It is interesting to examine the efficaciousness of integrating sensitivity weights into the IE module. To do this, we perform additional CMPLDWG parameter identification using EDRL with IE without revising parameter vector novelty scores.

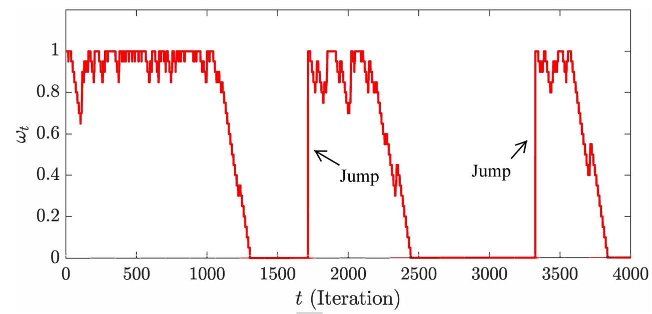


Fig. 8. Variation of the time-varying dynamic weight.

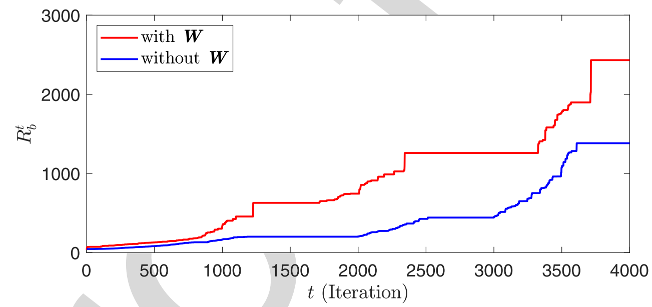


Fig. 9. The introduction of parameter sensitivity weights into EDRL with IE improves learning performance.

Fig. 9 shows two best reward collection curves corresponding to EDRL with IE by integrating $\boldsymbol{W}$ and without integrating $\boldsymbol{W}$, respectively. As can be seen, the introduction of $\boldsymbol{W}$ accelerates the exploitation and exploration in reaching the same best reward.

It is also significant to compare the proposed parameter identification approach with the presented algorithms in previous works. First, we focus on comparing our algorithm with the proposed parameter identification approach in [1], which also aims to identify a large number of parameters. The comparison shows that our approach can achieve better parameter identification accuracy and does not rely on *a priori* knowledge. And also, our method is easier to implement due to the utilization of mathematical representation of CMPLDWG model. In addition, the parameter identification accuracy using the proposed approach in [1] significantly relies on *a priori* knowledge about parameter setting. We have also compared the performance of our proposed approach with that of two other state-of-the-art optimization algorithms, Salp Swarm algorithm (SSA) and deep Q-networks (DQN). SSA is a newly proposed metaheuristic optimizer inspired by the process of looking for a food source by salps. SSA has demonstrated satisfying performance compared with other metaheuristic algorithms [34]. DQN is a cutting-edge reinforcement learning technique designed for sequential decision-making tasks [35]. The performance of the three algorithms (EDRL, SSA and DQN) is shown in Fig. 10. It can be seen that our proposed approach outperforms the other two methods in terms of the average fitness error, e_b^t. In comparison, SSA shows the fastest convergence rate. DQN takes the longest time to converge and shows the largest average fitness error. It is also important to

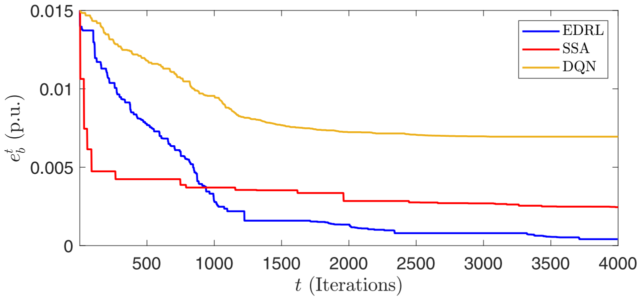


Fig. 10. Performance comparison of EDRL, SSA and DQN.

point out that DQN needs a significantly longer time to train a stable actor with satisfying identification performance.

VI. Conclusion

This paper presents a parameter identification approach for WECC composite load model. The proposed method employs a data-driven nonlinear feature selection technique to perform parameter sensitivity analysis, which avoids solving highly complex analytical derivatives caused by the high order and nonlinearity of differential equations of WECC composite load model. After that, the proposed method utilizes a cutting-edge approach inspired by evolutionary reinforcement learning technique, which is hybridized with an intelligent exploration mechanism to perform parameter identification. The parameter sensitivity weights are innovatively embedded in the reinforcement learning process to achieve efficient exploration. The numerical experiments demonstrate that the proposed approach can achieve promising accuracy. It is also shown that the proposed identifier can escape from local optima through the assistance of the intelligent exploration mechanism when stuck in local optima. Finally, it is verified that the integration of sensitivity weights into the reinforcement learning process accelerates the learning rate.

While our proposed approach can perform parameter identification of WECC composite load model with satisfying accuracy, the computational cost hinders its online application. Also, the model complexity stands in the way of widely applying WECC composite load model in the electric power industry. Considering this, one prospect for research on CMPLDWG is to simplify the model or develop a surrogate model to significantly reduce the computation cost and/or model complexity, while keeping the primary characteristics of WECC model.

References

[1] K. Zhang, H. Zhu, and S. Guo, "Dependency analysis and improved parameter estimation for dynamic composite load modeling," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3287–3297, Jul. 2017.

[2] J. Ma, D. Han, R.-M. He, Z.-Y. Dong, and D. J. Hill, "Reducing identified parameters of measurement-based composite load model," *IEEE Trans. Power Syst.*, vol. 23, no. 1, pp. 76–83, Feb. 2008.

[3] Q. Huang and V. Vittal, "Application of electromagnetic transient–transient stability hybrid simulation to FIDVR study," *IEEE Trans. Power Syst.*, vol. 31, no. 4, pp. 2634–2646, May 2016.

[4] R. Huang *et al.*, "Calibrating parameters of power system stability models using advanced ensemble Kalman filter," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 2895–2905, Oct. 2017.

[5] Y. Zhang, J. Wang, and Z. Li, "Uncertainty modeling of distributed energy resources: Techniques and challenges," *Curr. Sustain. Energy Rep.*, vol. 6, no. 2, pp. 42–51, Jun. 2019.

[6] M. Cui, J. Wang, and B. Chen, "Flexible machine learning-based cyberattack detection using spatiotemporal patterns for distribution systems," *IEEE Trans. Smart Grid*, vol. 11, no. 2, pp. 1805–1808, Mar. 2020.

[7] M. Cui, J. Wang, Y. Wang, R. Diao, and D. Shi, "Robust time-varying synthesis load modeling in distribution networks considering voltage disturbances," *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 4438–4450, Nov. 2019.

[8] M. Cui, M. Khodayar, C. Chen, X. Wang, Y. Zhang, and M. E. Khodayar, "Deep learning based time varying parameter identification for system-wide load modeling," *IEEE Trans. Smart Grid*, vol. 10, no. 6, pp. 6102–6114, Nov. 2018.

[9] J.-K. Kim, K. An, and J. Ma, "Fast and reliable estimation of composite load model parameters using analytical similarity of parameter sensitivity," *IEEE Trans. Power Syst.*, vol. 31, no. 1, pp. 663–671, Jan. 2016.

[10] S. Son *et al.*, "Improvement of composite load modeling based on parameter sensitivity and dependency analyses," *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 242–250, Jan. 2014.

[11] C. Wang, Z. Wang, J. Wang, and D. Zhao, "Robust time-varying parameter identification for composite load modeling," *IEEE Trans. Smart Grid*, vol. 10, no. 1, pp. 967–979, Jan. 2019.

[12] U.S. Energy Information Administration. (2011). *Share of Energy Used by Appliances and Consumer Electronics Increases in U.S. Homes*. [Online]. Available: https://www.eia.gov/consumption/residential/reports/2009/electronics.php

[13] U.S. Energy Information Administration. (2018). *EIA Electricity Data Now Include Estimated Small-Scale Solar PV Capacity and Generation*. [Online]. Available: https://www.eia.gov/todayinenergy/detail.php?id=23972

[14] J. Zhang, M. Cui, and Y. He, "Robustness and adaptability analysis for equivalent model of doubly fed induction generator wind farm using measured data," *Appl. Energy*, vol. 261, pp. 1–12, Mar. 2020.

[15] Modeling and Validation Work Group, "WECC dynamic composite load model (CMPLDW) specifications," Western Elect. Coordinating Council, Salt Lake City, UT, USA, Rep., 2015. AQ3

[16] A. Gaikwad, P. Markham, and P. Pourbeik, "Implementation of the WECC composite load model for utilities using the component-based modeling approach," in *Proc. IEEE/PES Transm. Distrib. Conf. Exposit. (T&D)*, May 2016, pp. 1–5.

[17] Q. Huang *et al.*, "A generic modeling and development approach for WECC composite load model," *Elect. Power Syst. Res.*, vol. 172, pp. 1–10, Jul. 2019.

[18] X. Wang, Y. Wang, D. Shi, J. Wang, and Z. Wang, "Two-stage WECC composite load modeling: A double deep Q-learning networks approach," *IEEE Trans. Smart Grid*, early access, Apr. 15, 2020, doi: 10.1109/TSG.2020.2988171.

[19] E. Conti, V. Madhavan, F. P. Such, J. Lehman, K. Stanley, and J. Clune, "Improving exploration in evolution strategies for deep reinforcement learning via a population of novelty-seeking agents," in *Proc. Adv. Neural Inf. Process. Syst.*, Dec. 2018, pp. 5032–5043

[20] D. Wierstra, T. Schaul, T. Glasmachers, Y. Sun, J. Peters, and J. Schmidhuber, "Natural evolution strategies," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 949–980, Mar. 2014.

[21] "The new aggregated distributed energy resources (DER_A) model for transmission planning studies: 2019 update," Elect. Power Res. Inst., Palo Alto, CA, USA, Rep., 2019.

[22] Z. Ma, Z. Wang, Y. Wang, R. Diao, and D. Shi, "Mathematical representation of the WECC composite load model," *J. Mod. Power Syst. Clean Energy*, to be published.

[23] M. Yamada, W. Jitkrittum, L. Sigal, E. P. Xing, and M. Sugiyama. (2019). *High-Dimensional Feature Selection by Feature-Wise Kernelized Lasso*. [Online]. Available: https://arxiv.org/abs/1202.0515.

[24] S. Wang *et al.*, "A data-driven multi-agent autonomous voltage control framework using deep reinforcement learning," *IEEE Trans. Power Syst.*, early access, Apr. 23, 2020, doi: 10.1109/TPWRS.2020.2990179.

[25] W. Kersting, *Distribution System Modeling and Analysis*. New York, NY, USA: CRC Press, 2011.

[26] J. D. Glover, M. S. Sarma, and T. J. Overbye, *Power System Analysis and Design*, 5th ed. Stamford, CT, USA: Cengage Learn., 2011.

[27] A. Hughes and B. Drury, *Electric Motors and Drives*, 4th ed. Amsterdam, The Netherlands: Elsevier, 2013.

[28] H. Wu and I. Dobson, "Analysis of induction motor cascading stall in a simple system based on the cascade model," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3184–3193, Aug. 2013.

[29] D. C. Yu, H. Liu, H. Sun, S. Lu, and C. Mccarthy, "Protective device coordination enhancement for motor starting programs," *IEEE Trans Power Del.*, vol. 20, no. 1, pp. 535–537, Jan. 2005.

[30] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, "Grid structural characteristics as validation criteria for synthetic networks," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3258–3265, Jul. 2017.

[31] C. M. Bishop, *Pattern Recognition and Machine Learning*. New York, NY, USA: Springer, 2009.

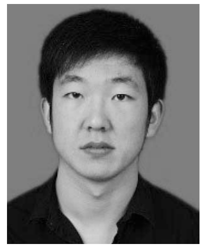
[32] R. D. Quint. (2015). *A Look Into Load Modeling: The Composite Load Model*. [Online]. Available: https://gig.lbl.gov/sites/all/files/6b-quint-composite-load-model-data.pdf

[33] "Reliability guide: Parameterization of the DER_A model," North Amer. Elect. Rel. Corporat., Atlanta, GA, USA, Rep., 2019.

[34] R. Abbassi, A. Abbassi, A. A. Heidari, and S. Mirjalili, "An efficient SALP swarm-inspired algorithm for parameters identification of photovoltaic cell models," *Energy Convers. Manag.*, vol. 179, pp. 362–372, Jan. 2019.

[35] V. François-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An introduction to deep reinforcement learning," *Found. Trends Mach. Learn.*, vol. 11, nos. 3–4, pp. 219–354, 2018.

Fankun Bu (Graduate Student Member, IEEE) received the B.S. and M.S. degrees from North China Electric Power University, Baoding, China, in 2008 and 2013, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. From 2008 to 2010, he worked as a Commissioning Engineer for NARI Technology Company Ltd., Nanjing, China. From 2013 to 2017, he worked as an Electrical Engineer for State Grid Corporation of China, Nanjing. His research interests include distribution system modeling, smart meter data analytics, renewable energy integration, and power system relaying.



Zixiao Ma (Graduate Student Member, IEEE) received the B.S. degree in automation and the M.S. degree in control theory and control engineering from Northeastern University in 2014 and 2017, respectively. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA, USA. His research interests are focused on the power system load modeling, microgrids, nonlinear control, and model order reduction.

Yuxuan Yuan (Graduate Student Member, IEEE) received the B.S. degree in electrical and computer engineering from Iowa State University, Ames, IA, USA, in 2017, where he is currently pursuing the Ph.D. degree. His research interests include distribution system state estimation, synthetic networks, data analytics, and machine learning.

Zhaoyu Wang (Member, IEEE) received the B.S. and M.S. degrees in electrical engineering from Shanghai Jiaotong University in 2009 and 2012, respectively, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology in 2012 and 2015, respectively. He is the Harpole-Pentair Assistant Professor with Iowa State University. He is the Principal Investigator for a multitude of projects focused on these topics and funded by the National Science Foundation, the Department of Energy, National Laboratories, PSERC, and Iowa Energy Center. His research interests include power distribution systems and microgrids, particularly on their data analytics and optimization. He is the Secretary of IEEE Power and Energy Society (PES) Award Subcommittee, the Co-Vice Chair of PES Distribution System Operation and Planning Subcommittee, and the Vice Chair of PES Task Force on Advances in Natural Disaster Mitigation Methods. He is an Editor of the IEEE TRANSACTIONS ON POWER SYSTEMS, the IEEE TRANSACTIONS ON SMART GRID, IEEE PES LETTERS, and the IEEE OPEN ACCESS JOURNAL OF POWER AND ENERGY, and an Associate Editor of *IET Smart Grid*.