Data-Driven Dynamic State Estimation in Power Systems via Sparse Regression Unscented Kalman Filter

Elham Jamalinia[a], Javad Khazaei[a], Rick S.Blum[a]

[a]*Department of Electrical and Computer Engineering at Lehigh University,*

Abstract

This paper proposes a novel data-driven modelling and dynamic state-estimation approach for nonlinear power and energy systems, highlighting the critical role of a known dynamic model for accurate state estimation in the face of uncertainty and complex models. The proposed framework consists of a two-phase approach: *data-driven model identification* and *state-estimation*. During the model identification phase, which spans a relatively short time interval, state feedback is collected to identify the dynamics of the nonlinear systems in the power grid using a novel density-guided sparse identification algorithm. Unlike conventional sparse regression, which relies on a large library of linear and nonlinear functions to fit data, our proposed algorithm iteratively updates a relatively small initial library by adding higher-order nonlinear functions if the coefficients of the current functions are dense. Following the identification of the model's dynamics, the estimation phase addresses the challenge of incomplete state measurements. By implementing an *unscented Kalman filter*, the state variables of the system are dynamically estimated by measuring the noisy output. Finally, simulation results on an IEEE 30-bus system are presented to illustrate the effectiveness of the density-guided sparse regression unscented Kalman filter compared to a physics-based unscented Kalman filter with model uncertainty. This study contributes to the fields of data-driven modelling techniques, machine learning for power systems, and computational intelligence in smart grids. It emphasizes the use of advanced sparse regression and unscented Kalman filter methods for state estimation, enhancing the robustness and accuracy of monitoring and control in electrical and energy systems.

Keywords: Data Driven Modelling, Dynamic State Estimation, Power and Energy Systems, Unscented Kalman filter.

1. Introduction

Data-driven modeling techniques play a pivotal role in enabling real-time monitoring, predictive maintenance, and optimized operation through Dynamic State Estimation (DSE). By continuously analyzing data from sensors and smart meters, these methods accurately assess system conditions, predict faults, and optimize resource allocation. This capability improves the reliability, efficiency and resilience of the grid, supports the integration of renewable energy sources and allows proactive grid management in response to changing demand and environmental conditions [1], [2]. These uncertainties might be the result of sensor error or other sources of interference. The dynamic of the system cannot be fully captured by the infrequent and non-synchronous data provided by the supervisory control and data acquisition (SCADA) system. On the other hand, wide area measurement and control (WAMAC) uses phasor measurement units (PMUs) to provide much more frequent measurements of the power system, enabling better understanding and control of the system dynamics [3]. As power systems expand substantially and renewable energy sources are rapidly integrated, the complexity of power systems continues to increase [4],[5],[6], making the development of an accurate power system model significantly more challenging. Considering the uncertainty in power system models and the uncertainty of measurements, there is a strong motivation to identify the model and estimate the states of the power system [7]. Addressing these challenges requires advanced data-driven modeling techniques and computational methods such as machine learning and big data analytics.

Kalman filtering has emerged as a powerful tool for accurately estimating the dynamics of state variables of power systems. [8, 9]. The Ensemble Kalman Filter (EnKF), which is a Monte Carlo approximation of a Kalman filter, was developed to increase the accuracy of estimation by a Kalman filter [10]. The original Kalman filter and the EnKF are applicable only for linear dynamics. On the other hand, the Extended Kalman Filter (EKF) was proposed for nonlinear dynamics. The EKF uses the Jacobian matrix to linearize the model in each time step and implements the Kalman filter for state estimation [8, 11]. Although EKF is successful in some low-dimensional models, the estimation process diverts from the true values in high-dimensional and highly nonlinear systems, including power systems. The Unscented Kalman Filter (UKF) was developed with unscented transformation to provide accurate estimation for nonlinear systems [12, 13]. The UKF employs a deter-

ministic sampling approach to propagate the state of the system through a set of carefully chosen sigma points. This enables more accurate estimation in higher order nonlinear systems[14], [15]. The established accuracy of the UKF for nonlinear systems motivated us to choose the UKF for our study of nonlinear power systems.

There are existing approaches in power systems to DSE that use physics-based models of power systems, disregarding the inherent uncertainties that exist in practical scenarios, which means obtaining a detailed model of power systems is not always possible or accurate enough to use. Further exploration is necessary to confront these uncertainties within the context of DSE for power systems. Data-driven model identification has the capability to unearth the inherent model uncertainties. Nevertheless, constructing accurate power system models from data is challenging, primarily due to the nonlinearity and uncertainty inherent to these systems [16]. In the context of power system identification, different studies have been devoted to exploring data-driven methods. These include Dynamic Mode Decomposition (DMD) [17], Sparse Identification of Nonlinear Dynamics (SINDy)[18, 19], neural networks [20], and Koopman theory [21], [22]. Among these methods, DMD heavily relies on a linear dynamics assumption, but can handle high-dimensional data. Neural-network-based approaches require a large amount of training data and are also infamous for not being interpretable [18, 19]. The Koopman operator connects DMD to nonlinear dynamics through an infinite dimensional linear operator. Under special circumstances and provided that a good measurement basis is selected, the Koopman operator may converge to a finite dimensional space, but this is not guaranteed for many systems[22]. On the other hand, sparse identification uses the sparse regression technique to identify dominant dynamics of candidate functions, and has shown promise in accurately modeling the unknown dynamics of nonlinear systems [18, 19]. Among the major advantages of sparse regression is its sparsity, which enables easy implementation, reduces the training time, results in an interpretable model, and provides an accurate formulation that outperforms other model identification techniques. While the existing research shows the significant potential of sparse regression in identifying nonlinear dynamics of synchronous generators, its application in dynamic state-estimation has not been reported yet.

1.1. Contributions

To be specific, our contribution can be outlined as:

- Unlike traditional DSE techniques that predominantly rely on physics-based models and often fail to account for model/parameter fluctuations and uncertainties, our research uniquely addresses these challenges, focusing on state estimation for nonlinear power systems. This emphasis ensures more robust and accurate system performance under real-world conditions, where uncertainties are inherent and unavoidable.

- One significant contribution of our work is the enhancement of the conventional sparse regression method. Traditional approaches involve constructing an extensive library of linear and nonlinear functions to represent system dynamics, which can be computationally intensive and slow, particularly for real-time applications. Our innovative approach reduces computational complexity and enhances computation speed by starting with a smaller, more focused set of functions. This strategic reduction in the initial library size not only minimizes computational demands but also decreases interdependencies among the functions, thereby streamlining the process of real-time model identification (see Fig. 1).

- Another critical advancement in our method is the introduction of an improved sparse regression technique that optimizes feature selection. By systematically evaluating the correlation of feature values with the ratio of the dominant element to the sparsity tuning parameter (γ), our technique ensures more precise and effective feature selection. This refinement addresses the limitations of the original sparse regression method, providing a more sophisticated and accurate approach to identifying relevant system dynamics. The effectiveness of our approach is demonstrated through comprehensive state-of-the-art comparisons, as shown in Table 1.

The key distinction between learning-based machine learning models and the sparse regression approach proposed in this paper lies in their methodology. While machine learning methods typically identify a nonlinear model that best fits the measured data, sparse regression pinpoints the exact underlying dynamics of the system. Furthermore, techniques such as reinforcement learning often require vast amounts of training data and are computationally intensive [23]. In contrast, our method is significantly simpler and requires

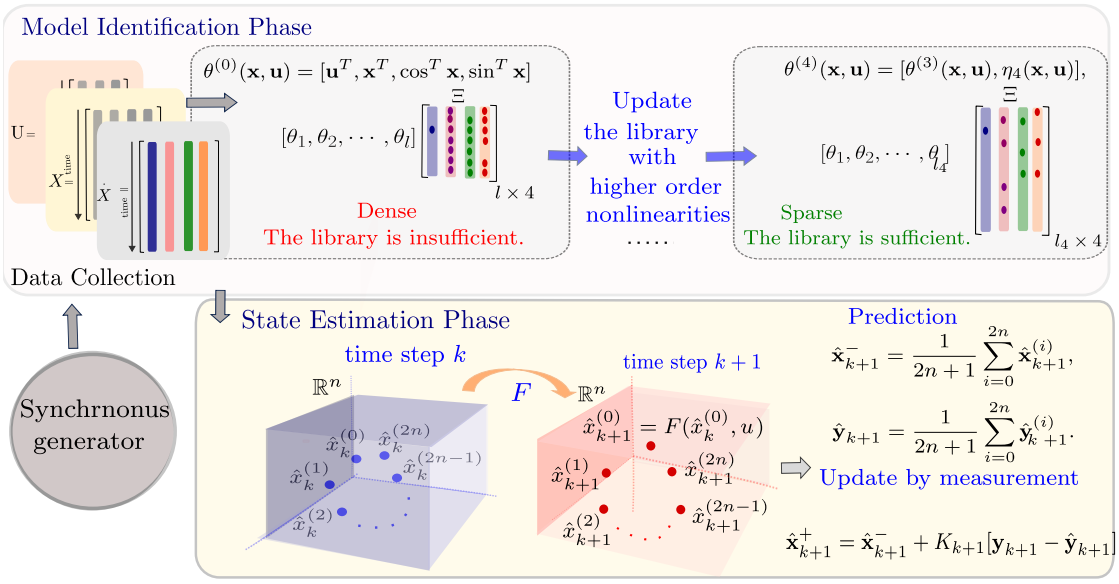

Figure 1: Proposed density-guided sparse regression of nonlinear generators: the library of function is updated when a column of Ξ is dense and higher order nonlinear functions should be added to the library till all the columns of Ξ are sparse.

Table 1: State-of-the-art study on model identification of power systems

Method	Prior knowledge	Linearization	Adaptive
DMD [17]	✓	✗	✗
Neural Network [20]	✓	✓	✗
Koopman Theory [21]	✗	✓	✗
Sparse Regression [19]	✗	✓	✓
Proposed Method	✓	✓	✓

far less data, making it highly suitable for applications where data availability is limited and low model complexity is essential.

The rest of the paper is organized as follows. Section 2 presents the formulation of the model free state estimation problem. In sections 3 and 3.2, sparse regression and UKF is introduced. Section 4 proposes the innovative density-guided sparse regression UKF and section 5 includes the case studies.

2. Problem Formulation

Without loss of generality, the model of a synchronous generator is considered as an illustrative example within nonlinear power systems. Here, the problem of model identification and state estimation for a synchronous generator is structured into two distinct segments. In the first part, the problem is model-free, and the generator's inherent model remains undisclosed. In the second part, the focus is shifted to the situation where the physics-based model is established, yet the precise parameters of the model remain undetermined.

2.1. Formulation of Synchronous Generator

Generally, the dynamic of a generator with additive process and measurement noise has the following formulation.

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) + \mathbf{w}, \tag{1}$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}, \mathbf{u}) + \mathbf{v}, \tag{2}$$

where the system has n state variables $\mathbf{x} \in \mathbb{R}^n$ and d inputs $\mathbf{u} \in \mathbb{R}^d$. Also, $\mathbf{f}(\mathbf{x}, \mathbf{u})$ is the transition function, $g(\mathbf{x}, \mathbf{u})$ is the measurement function, and $\mathbf{w} \in \mathbb{R}^n$ and $\mathbf{v} \in \mathbb{R}^m$ are the process and measurement noise, assumed to be Gaussian, i.e., $\mathbf{w}_k \sim N(\mathbf{0}, \mathbf{Q}_k)$, and $\mathbf{v}_k \sim N(\mathbf{0}, \mathbf{R}_k)$, where $Q_k \in \mathbb{R}^{n \times n}$ and $R_k \in \mathbb{R}^{m \times m}$ are covariances of the process noise and measurement noise respectively. The transition function, $\mathbf{f}$ in (1), is assumed to be **unknown** and the goal is first to identify the transition function and second to estimate the states of the system. Without loss of generality, the physical model of the fourth-order synchronous generator from [24] is considered. The behavior of the system is described by the following equations, which need to be

identified:

$$\dot{x}_1 = \omega_0\, x_2, \tag{3}$$

$$\dot{x}_2 = \frac{1}{J}\, u_1 - \frac{D}{J}\, x_2 - \frac{1}{J}\cdot\frac{1}{2}\cdot\left(\frac{1}{x_q} - \frac{1}{x'_d}\right)\cdot u_3^2 \sin(2x_1) - \frac{1}{J}\cdot\frac{1}{x'_q} u_3\, x_3\, \sin x_1, \tag{4}$$

$$\dot{x}_3 = \frac{1}{T'_{do}} u_2 - \frac{1}{T'_{do}}\left(1 + \frac{x_d - x'_d}{x'_d}\right) x_3 + \frac{1}{T'_{do}}\left(\frac{x_d - x'_d}{x'_d}\right) u_3 \cos x_1, \tag{5}$$

$$\dot{x}_4 = -\frac{1}{T'_{qo}}\, x_4 + \frac{1}{T'_{qo}}\left(\frac{x_q - x'_q}{x'_q}\right)\, u_3 \sin x_1, \tag{6}$$

The generator has four states , $n = 4$, and three inputs, $d = 3$, defined as $\mathbf{x} = [x_1 \quad x_2 \quad x_3 \quad x_4]^T = [\delta \quad \Delta w \quad e'_q \quad e'_d]^T$ and $\mathbf{u} = [u_1 \quad u_2 \quad u_3] = [T_m \quad E_{fd} \quad V_t]^T$, where δ (rad) is the rotor angle, $\Delta\omega$ (rad/sec) is the deviation of rotor speed from the synchronous speed, e'_q (volt) is the transient voltage along the q-axis, and e'_d (volt) is the transient voltage along the d-axis. The three inputs are T_m (Newton-meter), which represents the mechanical input torque, E_{fd} (volt), the field excitation voltage (steady-state internal voltage of the armature), and V_t (volt), the terminal bus voltage. The measurement function, representing the electrical output power, is defined as:

$$y = \frac{u_3}{x'_d} x_3 \sin(x_1) + \frac{u_3^2}{2}\left(\frac{1}{x_q} - \frac{1}{x'_d}\right)\sin(2x_1). \tag{7}$$

In these equations, J represents the inertia constant, and D is the damping factor. The parameters T'_{do} and T'_{qo} are the transient open circuit time constants on the direct and quadratic axes, respectively, while x_d and x_q denote the synchronous reactance of the direct and quadratic axes. The terms x'_d and x'_q represent the transient reactance along the direct and quadratic axes.

2.2. Model with uncertainty

In this subsection, the physics-based model of the nonlinear synchronous generator is introduced where the linear and nonlinear functions in the transition function are known, however the parameters have uncertainties. Later

Table 2: Values of model parameters

Parameter	Value
J	13
D	0.05
T'_{do}	0.131
T'_{qo}	0.0131
x_d	2.06
x_q	1.254
x'_d	0.375
x'_q	0.375

in section 5, it is illustrated that our proposed model-free approach can significantly overcome the physics-based state estimation. The physics-based model of a nonlinear fourth-order synchronous generator is written as (3)-(6) [25]. While the values of the model parameters are predetermined, it is assumed that there is uncertainty in the value of parameters as formulated in

$$x_d = \bar{x}_d + \Delta x_d, \quad |\Delta x_d| \leq D_{x_d} \tag{8}$$
$$x_q = \bar{x}_q + \Delta x_q, \quad |\Delta x_q| \leq D_{x_q} \tag{9}$$
$$x'_d = \bar{x}'_d + \Delta x'_d, \quad |\Delta x'_d| \leq D_{x'_d} \tag{10}$$
$$x'_q = \bar{x}'_q + \Delta x'_q, \quad |\Delta x'_q| \leq D_{x'_q}, \tag{11}$$

where $\bar{x}_d$ is the predetermined value, D_{x_d} is the bound of uncertainty (known for each parameter), and Δx_d is the deviation of the parameter (unknown).

3. Mathematical Background

3.1. Sparse Regression Theory

Assume a dynamical system is modeled as

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t)) \tag{12}$$

where $\mathbf{x}$ is the state vector, $\mathbf{u}$ is the input vector and $f(\mathbf{x}, \mathbf{u}) : \mathbb{R}^n \times \mathbb{R}^d \to \mathbb{R}^n$. The underlying principle of sparse identification is that the function f consists of only a few active terms. To identify the governing equations of the nonlinear system, it is essential to collect the values of states, the time derivatives of

$$\theta(\mathbf{X},\mathbf{U}) = \begin{bmatrix}\mathbf{1} & \mathbf{U} & \mathbf{X} & \mathbf{P_2}(\mathbf{X},\mathbf{U}) & \mathbf{P_3}(\mathbf{X},\mathbf{U}) & \cdots & sin(\mathbf{X},\mathbf{U}) & sin(\mathbf{X},\mathbf{U})\end{bmatrix} \tag{13}$$

states and the inputs over m time steps during the identification phase [26]. (13) defines the vector of candidate functions $\theta(\mathbf{X},\mathbf{U})$ used in the density-guided sparse regression algorithm. This vector is constructed to encompass a variety of linear and nonlinear terms that describe the relationships between the input variables represented by $\mathbf{X}$ (state variables) and $\mathbf{U}$ (control inputs). The first element, $\mathbf{1}$, represents a constant term, essential for fitting models that may have nonzero intercepts. The term $\mathbf{U}$ captures the direct influence of control input on system dynamics, allowing the model to account for external manipulations. Additionally, the term $\mathbf{X}$ includes the state variables of the system, which are crucial to understanding its current condition. Higher-order polynomial terms, represented by $\mathbf{P_2}(X,U)$ and $\mathbf{P_3}(X,U)$, encompass second-order and third-order polynomial combinations of $\mathbf{X}$ and $\mathbf{U}$, facilitating the modeling of more complex relationships between state and control variables by capturing nonlinear interactions. Furthermore, the inclusion of trigonometric functions, specifically $\sin(\mathbf{X},\mathbf{U})$, allows the model to account for periodic behaviors that may arise in certain power system applications.

3.1.1. Data Collection

Since all measurements contain noise, it is needed to smooth out the data to increase reliability before identification, and in this work, Savitzky-Golay filtering [27] is applied. Based on the filtered data, the following matrices are

defined

$$\mathbf{X} = \begin{bmatrix} \mathbf{x}^T(t_1) \\ \mathbf{x}^T(t_2) \\ \vdots \\ \mathbf{x}^T(t_m) \end{bmatrix} = \begin{bmatrix} x_1(t_1) & x_2(t_1) & \cdots & x_n(t_1) \\ x_1(t_2) & x_2(t_2) & \cdots & x_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ x_1(t_m) & x_2(t_m) & \cdots & x_n(t_m) \end{bmatrix},$$

$$\dot{\mathbf{X}} = \begin{bmatrix} \dot{\mathbf{x}}^T(t_1) \\ \dot{\mathbf{x}}^T(t_2) \\ \vdots \\ \dot{\mathbf{x}}^T(t_m) \end{bmatrix} = \begin{bmatrix} \dot{x}_1(t_1) & \dot{x}_2(t_1) & \cdots & \dot{x}_n(t_1) \\ \dot{x}_1(t_2) & \dot{x}_2(t_2) & \cdots & \dot{x}_n(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \dot{x}_1(t_m) & \dot{x}_2(t_m) & \cdots & \dot{x}_n(t_m) \end{bmatrix},$$

$$\mathbf{U} = \begin{bmatrix} \mathbf{u}^T(t_1) \\ \mathbf{u}^T(t_2) \\ \vdots \\ \mathbf{u}^T(t_m) \end{bmatrix} = \begin{bmatrix} u_1(t_1) & u_2(t_1) & \cdots & u_d(t_1) \\ u_1(t_2) & u_2(t_2) & \cdots & u_d(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ u_1(t_m) & u_2(t_m) & \cdots & u_d(t_m) \end{bmatrix}, \tag{14}$$

where $\boldsymbol{X}$, $\boldsymbol{U}$, and $\dot{\boldsymbol{X}}$ are the matrices of the filtered states, inputs, and time derivative of states, respectively. In the model identification phase, it is assumed that the state variables $\mathbf{X}$, their time derivatives $\dot{\mathbf{X}}$, and the input data $\mathbf{U}$ can all be measured. Given this assumption, the Savitzky-Golay filter is applied to the measured data to smooth out noise in $\mathbf{X}$, $\dot{\mathbf{X}}$, and $\mathbf{U}$, ensuring the accuracy of the model identification. Since this process occurs offline, smoothing the data prior to model identification is crucial. The time stamps $t_1, t_2, \ldots, t_m$ represent discrete points in the data collection phase, which occurs before the identified model is used in Dynamic State Estimation (DSE). This ensures that the model is constructed based on pre-processed and noise-reduced data.

3.1.2. Library of Candidate Functions

To identify the terms on the right-hand side of (12), a library of candidate functions is chosen. This library may encompass constants, polynomials, and trigonometric terms. The structure of this library is denoted as in (13) (which can be expanded by higher orders of nonlinearities), where $P_2(\mathbf{X}, \mathbf{U})$ given in (15) represents the second-order combinations of elements of $\mathbf{X}$ and $\mathbf{U}$.

$$P_2(\mathbf{X}, \mathbf{U}) = \begin{bmatrix} x_1(t)u_1(t) & x_1(t)u_2(t) & \cdots & x_n(t)u_d(t) \end{bmatrix} \tag{15}$$

To construct the design matrix for model identification, the matrix $P_2(\mathbf{X}, \mathbf{U})$, shown in (15), captures the products of the state variables and inputs at each time step. This matrix is then used to generate the design matrix $\Theta(\mathbf{X}, \mathbf{U})$, defined in (16). The matrix $\Theta(\mathbf{X}, \mathbf{U})$ is formed by stacking the regression terms $\theta(\mathbf{X}(t), \mathbf{U}(t))$ for each time instant $t_1, t_2, \ldots, t_m$, where L is the number of regression terms at each time point and m is the total number of time steps collected. This matrix is critical for identifying the system's dynamics through the proposed methodology. After m time steps, the matrix $\boldsymbol{\Theta}$, which augments the matrix $\boldsymbol{\theta}$ at each time step, is defined as (stacking rows):

$$\Theta(\mathbf{X}, \mathbf{U}) = \begin{bmatrix} \theta(\mathbf{X}(t_1), \mathbf{U}(t_1)) \\ \theta(\mathbf{X}(t_2), \mathbf{U}(t_2)) \\ \vdots \\ \theta(\mathbf{X}(t_m), \mathbf{U}(t_m)) \end{bmatrix}_{m \times L} \quad time \downarrow \tag{16}$$

The time derivative of states $\dot{\boldsymbol{X}}$ can now be written as a linear expansion of the functions in the library $\boldsymbol{\Theta}$

$$\dot{\mathbf{X}} = \boldsymbol{\Theta}(\mathbf{X}, \mathbf{U})\boldsymbol{\Xi}, \tag{17}$$

where $\boldsymbol{\Xi}$ is the matrix of coefficients for the candidate functions in Θ.

3.1.3. Identification by Sparse Regression

To continue with the regression problem, a sparsity-promoting hyper parameter is introduced. The goal of the optimization problem is then defined as

$$\xi_h = \arg\min \|\dot{X}_h - \Theta(\mathbf{X}, \mathbf{U})\hat{\xi}_h\|_2 + \gamma\|\hat{\xi}_h\|_0 \tag{18}$$

where ξ_h is the h-th column of $\boldsymbol{\Xi}$ represented by $\xi_h = [\xi_1\ \xi_2; \cdots\ \xi_p]^T$, moreover $\|.\|_2$ and $\|.\|$ denote l_2 and l_0 norm respectively. A sparse $\boldsymbol{\Xi}$ is derived by iteratively solving a sequentially thresholded least-square optimization [28]. The second term on the right-hand side of 18 introduces a sparsity constraint to the optimization problem. Specifically, the term $\gamma\|\hat{\xi}_h\|_0$ aims to promote a sparse solution for the vector $\hat{\xi}_h$, which represents the coefficients in the regression model. The l_0 norm counts the number of non-zero elements in $\hat{\xi}_h$, effectively enforcing sparsity by penalizing the inclusion of unnecessary parameters. This approach contrasts with the first term, which uses the l_2 norm to minimize the discrepancy between the predicted state dynamics $\Theta(\mathbf{X}, \mathbf{U})\hat{\xi}_h$ and the observed dynamics $\dot{X}_h$. The sparsity tuning parameter

γ plays a critical role in balancing the trade-off between fitting the data (as captured by the l_2 norm term) and maintaining a sparse representation of the model coefficients (as captured by the l_0 norm term). Adjusting γ influences the degree of sparsity in the solution: a larger γ encourages sparser solutions, while a smaller γ allows more parameters to be included in the model. This flexibility helps ensure that the identified model remains relevant and accurate in capturing the underlying dynamics of the system.
In (18), the matrix Ξ is $L \times n$, where L is the number of possible regression terms generated by the function $\Theta(\mathbf{X}, \mathbf{U})$, and n is the number of states. The vector ξ_h represents the h-th column of Ξ and is a sparse vector, meaning it has only a subset of non-zero elements. Specifically, the number of non-zero elements in ξ_h is denoted by p, which is much smaller than L.

3.2. Unscented Kalman Filter

Given the identified model in previous section, by measuring the noisy output $\mathbf{y}$, the states of the system are estimated via UKF with the assumption of knowing the statistics of the process and measurement noise. As UKF applies to discrete-time models of the system, it is necessary to write the discrete-time model first.

3.2.1. Discrete Time Model

Using the Euler method, the time evolution of states and outputs is represented by

$$\mathbf{x}_{k+1} = \mathbf{F}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{w}_k, \quad \mathbf{y}_k = \mathbf{G}(\mathbf{x}_k, \mathbf{u}_k) + \mathbf{v}_k, \tag{19}$$

where $\boldsymbol{F}(.,.)$ and $\boldsymbol{G}(.,.)$ are the discrete transition function and measurement function respectively and are defined as

$$\mathbf{F}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{x}_k + \hat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)\Delta t, \tag{20}$$

$$\mathbf{G}(\mathbf{x}_k, \mathbf{u}_k) = \mathbf{g}(\mathbf{x}_k, \mathbf{u}_k). \tag{21}$$

In (20), $\hat{\mathbf{f}}(\mathbf{x}_k, \mathbf{u}_k, \mathbf{w}_k)$ represents the identified system model, which is originally in continuous form. This model captures the dynamics of the system based on the identified model from the model identification process. Since the filtering process operates in discrete time, $\hat{\mathbf{f}}$ is discretized with a time step Δt, yielding the discrete state transition function $\mathbf{F}(\mathbf{x}_k, \mathbf{u}_k)$. This allows the model to be updated at each time step of the estimation process, making it

suitable for use in the Kalman filtering framework.
In UKF, by a set of deterministic sample points, known as sigma points, the mean and covariance of a Gaussian random variable are estimated. When sigma points are transformed by a nonlinear function, their true mean and covariance can be calculated accurately to the 3rd order of Taylor series expansion [29].

3.2.2. Prediction Step

In time step k, given the mean $\hat{\boldsymbol{x}}_k^+$ and covariance $\boldsymbol{P}_k^+$ of a random variable $\mathbf{x} \in \mathbb{R}^n$, $2n+1$ Sigma points of the variable with equal weights are calculated by the following procedure [30]

$$\hat{\mathbf{x}}_k^{(0)} = \hat{\mathbf{x}}_k^+, \tag{22}$$

$$\hat{\mathbf{x}}_k^{(i)} = \hat{\mathbf{x}}_k^+ + \tilde{\mathbf{x}}^{(i)}, \qquad i = 1, \cdots, 2n \tag{23}$$

$$\tilde{\mathbf{x}}^{(i)} = \left(\sqrt{n\boldsymbol{P}_k^+}\right)_i^T, \qquad i = 1, \cdots, n \tag{24}$$

$$\tilde{\mathbf{x}}^{(n+i)} = -\left(\sqrt{n\boldsymbol{P}_k^+}\right)_i^T, \qquad i = 1, \cdots, n \tag{25}$$

where $\hat{\mathbf{x}}_k^{(i)}, i = 0, \cdots, 2n$ is the $(i+1)$-th sigma point of state variable and $\boldsymbol{P}_k^+$ is the estimated covariance of the state variable at time step k. In equations (24) and (25), the notation $(\cdot)_i$ represents the i-th element of the corresponding vector. Specifically, it refers to the i-th row of the matrix $\sqrt{n\boldsymbol{P}_k^+}$, which is the Cholesky factorization of the estimated covariance matrix at time step k [31]. This factorization allows for efficient computation of the square root of the covariance matrix, facilitating the generation of sigma points. Each sigma point is transformed by the nonlinear function in (20) to the next time step

$$\hat{\mathbf{x}}_{k+1}^{(i)} = \boldsymbol{F}(\mathbf{x}_k^{(i)}, \mathbf{u}_k), \tag{26}$$

$$\hat{\mathbf{y}}_{k+1}^{(i)} = \boldsymbol{G}(\mathbf{x}_{k+1}^{(i)}, \mathbf{u}_{k+1}). \tag{27}$$

By these propagated sigma points, the predicted state and the predicted output at time step $k+1$ are,

$$\hat{\mathbf{x}}_{k+1}^- = \frac{1}{2n+1}\sum_{i=0}^{2n} \hat{\mathbf{x}}_{k+1}^{(i)}, \quad \hat{\mathbf{y}}_{k+1} = \frac{1}{2n+1}\sum_{i=0}^{2n} \hat{\mathbf{y}}_{k+1}^{(i)}. \tag{28}$$

Consequently, the covariance of the state is predicted by

$$\boldsymbol{P}_{k+1}^{-} = \frac{1}{2n}\sum_{i=1}^{2n}(\hat{\mathbf{x}}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1}^{-})(\hat{\mathbf{x}}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1}^{-})^T + \boldsymbol{Q}_k, \tag{29}$$

where $\boldsymbol{P}_{k+1}^{-}$ is the predicted covariance and $\boldsymbol{Q}_k$ is added to include the effect of process noise. Additionally, the covariance of the output and cross covariance of state and output at time step $k+1$ is predicted by

$$\boldsymbol{P}_{\mathbf{y}} = \frac{\sum_{i=0}^{2n}(\hat{\mathbf{y}}_{k+1}^{(i)} - \hat{\mathbf{y}}_{k+1})(\hat{\mathbf{y}}_{k+1}^{(i)} - \hat{\mathbf{y}}_{k+1})^T}{2n+1} + \boldsymbol{R}_k, \tag{30}$$

$$\boldsymbol{P}_{\mathbf{xy}} = \frac{1}{2n+1}\sum_{i=0}^{2n}(\hat{\mathbf{x}}_{k+1}^{(i)} - \hat{\mathbf{x}}_{k+1}^{-})(\hat{\mathbf{y}}_{k+1}^{(i)} - \hat{\mathbf{y}}_{k+1})^T. \tag{31}$$

where $\boldsymbol{P}_{\mathbf{y}}$ denotes the predicted covariance of the measurement and $\boldsymbol{R}_k$ is the measurement noise covariance. Additionally, $\boldsymbol{P}_{\mathbf{xy}}$ denotes the cross-covariance between the state and the measurement.

3.2.3. Update Step

With measurement $\mathbf{y}_{k+1}$ and Kalman filter equations [32], the estimation of the mean and covariance in previous step is updated by

$$\hat{\mathbf{x}}_{k+1}^{+} = \hat{\mathbf{x}}_{k+1}^{-} + K_{k+1}[\mathbf{y}_{k+1} - \hat{\mathbf{y}}_{k+1}], \tag{32}$$

$$\boldsymbol{P}_{k+1}^{+} = \boldsymbol{P}_{k+1}^{-} - K_{k+1}\boldsymbol{P}_{\mathbf{y}}K_{k+1}^{T}, \tag{33}$$

where K_{k+1} is the Kalman gain and is calculated by

$$K_{k+1} = \boldsymbol{P}_{\mathbf{xy}}\boldsymbol{P}_{\mathbf{y}}^{-1}. \tag{34}$$

Sigma points for propagating in the next time step is calculated by $\hat{\mathbf{x}}_{k+1}^{+}$ and $\boldsymbol{P}_{k+1}^{+}$ and the two steps of prediction and update will be repeated.

4. Density-Guided Sparse Regression UKF

In this section, Algorithm 1 is proposed to derive a sufficient library of functions to identify the dynamics of synchronous generators in power grid. The model of the system is identified and written as $\dot{\mathbf{x}}^T = \boldsymbol{\theta}(\mathbf{x}, \mathbf{u})\boldsymbol{\Xi}$, where $\boldsymbol{\Xi}$ is the matrix resulting from sparse regression in Algorithm 1. The matrix

$\boldsymbol{\Xi}$ depends on the library of functions in $\boldsymbol{\theta}(\mathbf{x}, \mathbf{u})$. This proposed algorithm begins with a relatively small library of nonlinear functions to find a model to fit data set and represent the system dynamics. If one or more columns of the matrix $\boldsymbol{\Xi}$ are not sparse, the algorithm will recognize that the nonlinear functions in the selected library were insufficient to represent the dynamics of the system. The proposed method is shown in Fig. 1.

4.1. Initial library of functions

Among the set of all possible nonlinear and linear functions, the set $\boldsymbol{H}$ is chosen based on the inherent properties of the system.

$$\boldsymbol{H} = \{\boldsymbol{\eta}_1(\mathbf{x}, \mathbf{u}), \cdots, \boldsymbol{\eta}_c(\mathbf{x}, \mathbf{u})\}, \tag{35}$$

where $\boldsymbol{\eta}_i(\mathbf{x}, \mathbf{u})$, $i = 1, \cdots, c$ are sets of functions raised to the i^{th} power. While $\boldsymbol{H}$ has many functions and is typically of substantial size, a subset of $\boldsymbol{H}$ is selected as an initial library of functions for our sparse regression method

$$\theta^0(\mathbf{x}, \mathbf{u}) \subset \boldsymbol{H}, \tag{36}$$

where $\boldsymbol{\theta}^0(.,.)$ is assumed to include l functions

$$\boldsymbol{\theta}^0(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \theta_1(\mathbf{x}, \mathbf{u}) & \theta_2(\mathbf{x}, \mathbf{u}) & \cdots & \theta_l(\mathbf{x}, \mathbf{u}) \end{bmatrix}. \tag{37}$$

The initial selection is based on the system's inherent properties, which allows us to focus on the most relevant terms. The choice of this initial library directly impacts the final model. A well-constructed library ensures that the model accurately captures the system dynamics while balancing complexity. If critical terms from $\boldsymbol{H}$ are excluded, the model may be unable to fit the data, failing to capture essential dynamics. Conversely, including too many irrelevant terms from $\boldsymbol{H}$ can lead to overfitting, increasing computational complexity, and reducing model generalizability. In practice, some knowledge of dynamic functions is known and the problem is not treated as a black-box model. This can limit the number of library functions in $\boldsymbol{H}$. By selecting a relevant subset of functions from $\boldsymbol{H}$, the proposed method ensures accurate, sparse, and interpretable models.

Assumption 4.1. *For each j, $\theta_j(\boldsymbol{x}, \boldsymbol{u})$ cannot be written as a linear combination of the other components $\theta_i(\boldsymbol{x}, \boldsymbol{u}), \forall i \neq j$.*

Taking into account all the snapshots of $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{u} \in \mathbb{R}^d$, the matrices $\boldsymbol{X} \in \mathbb{R}^{m \times n}$, $\boldsymbol{U} \in \mathbb{R}^{m \times d}$, $\dot{\boldsymbol{X}} \in \mathbb{R}^{m \times n}$ and Θ are constructed as follows:

$$\Theta^0(\boldsymbol{X}, \boldsymbol{U}) = [\theta^0(t_1); \theta^0(t_2); \cdots; \theta^0(t_m)], \tag{38}$$

where $\theta^0(t_i)$ is the initial library in i-th time step. An initial solution for Ξ in sparse regression algorithm is expressed as $\boldsymbol{\Xi}^0 = \boldsymbol{\Theta}^0(\boldsymbol{X}, \boldsymbol{U}) \backslash \dot{\boldsymbol{X}}$. In each column of $\boldsymbol{\Xi}^0$, there is a dominant element with maximum absolute value which is denoted as ξ^*. By considering the sparsity tuning parameter γ, the elements with absolute value less than ξ^*/γ will be replaced with zero and the expansion of $\boldsymbol{\Xi}^0 \in \mathbb{R}^{M \times n}$ can be written as

$$\boldsymbol{\Xi}^0 = \begin{bmatrix} \boldsymbol{\xi}_1^0 \\ \boldsymbol{\xi}_2^0 \\ \vdots \\ \boldsymbol{\xi}_n^0 \end{bmatrix}^T = \begin{bmatrix} \xi_{1,1}^0 & \xi_{1,2}^0 & \cdots & \xi_{1,n}^0 \\ \xi_{2,1}^0 & \xi_{2,2}^0 & \cdots & \xi_{2,n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \xi_{M,1}^0 & \xi_{M,2}^0 & \cdots & \xi_{M,n}^0 \end{bmatrix}, \tag{39}$$

where $\boldsymbol{\xi}_j^0$, $j = 1, \cdots, n$ is the j-th column and $\xi_{i,j}^0$ is the i-th element of this column. Thus, dynamics of the system are modeled as

$$\begin{aligned} &\left[\dot{x}_1, \dot{x}_2, \cdots, \dot{x}_n\right] = \\ &\left[\theta_1(\mathbf{x}, \mathbf{u}), \theta_2(\mathbf{x}, \mathbf{u}), \cdots, \theta_l(\mathbf{x}, \mathbf{u})\right] \begin{bmatrix} \xi_{1,1}^0 & \xi_{1,2}^0 & \cdots & \xi_{1,n}^0 \\ \xi_{2,1}^0 & \xi_{2,2}^0 & \cdots & \xi_{2,n}^0 \\ \vdots & \vdots & \ddots & \vdots \\ \xi_{l,1}^0 & \xi_{l,2}^0 & \cdots & \xi_{l,n}^0 \end{bmatrix}. \end{aligned} \tag{40}$$

The model is now a linear expansion of the nonlinear functions in $\theta^0(\mathbf{x}, \mathbf{u})$. A challenging question is whether the non-linear functions were sufficient enough to represent the model. In the following, this challenge is addressed by proposing a density-guided approach.

4.2. Updating the library: The density suggests a higher order of nonlinearities

The matrix Ξ^0 has n columns where the column $\xi_i^{\mathbf{0}}$ corresponds to the dynamic of i-th state

$$\dot{x}_j = \left[\theta_1(\mathbf{x}, \mathbf{u}), \theta_2(\mathbf{x}, \mathbf{u}), \cdots, \theta_l(\mathbf{x}, \mathbf{u})\right] \begin{bmatrix} \xi_{1,j}^0 \\ \xi_{2,j}^0 \\ \vdots \\ \xi_{M,j}^0 \end{bmatrix}, \tag{41}$$

where $j = 1, \cdots, n$ and (41) is a linear expansion of all the functions in the library $\theta^{\mathbf{0}}(\mathbf{x}, \mathbf{u})$ resulted by the original sparse regression algorithm.

Assumption 4.2. *Assuming an exact representation in* (41) *exists with finite l, the time derivative of each state variable $\dot{x}_j$ can be represented with a finite l (number of terms). The functions $\theta_j(\boldsymbol{x}, \boldsymbol{u})$ are members of the set $\boldsymbol{H}$ defined in* (35).

Theorem 4.1. *Under assumption 4.2, if the column $\boldsymbol{\xi}_j$ is dense, this density indicates that the library should be updated by adding more nonlinear functions to represent the dynamic of $\dot{x}_j$. If the column $\boldsymbol{\xi}_j$ is sparse, this sparsity indicates that the functions in the library are sufficiently large to represent the dynamics of the system.*

Proof. Suppose that the underlying dynamic of $\dot{x}_j$ is of the form

$$\dot{x}_j = \sum_{k=1}^{K} \alpha_{k,j} \varphi_k^j(\mathbf{x}, \mathbf{u}), \tag{42}$$

where $\varphi_k^j(\mathbf{x}, \mathbf{u})$, $k = 1, \cdots, K$ is a function of $\mathbf{x}$ and $\mathbf{u}$ and $\alpha_{k,j}^j \in \mathbb{R}$ is the corresponding coefficient. By the sparse regression approach and the matrix $\Xi^{\mathbf{0}}$, $\dot{x}_j$ is identified as

$$\tilde{\dot{x}}_j \approx \sum_{k=1}^{L} \xi_{k,j} \theta_k(\mathbf{x}, \mathbf{u}), \tag{43}$$

where $L \gg K$ so the column $\boldsymbol{\xi}_j$ can potentially be sparse. Thus, the error $(\tilde{\dot{x}}_j - \dot{x}_j)$ is

$$e_j = \sum_{k=1}^{K} \big(\xi_{k,j} \theta_k(\mathbf{x}, \mathbf{u}) - \alpha_{k,j} \varphi_k^j(\mathbf{x}, \mathbf{u})\big) + \sum_{k=K+1}^{L} \xi_{k,j} \theta_k(\mathbf{x}, \mathbf{u}). \tag{44}$$

If for all $1 \leq k \leq K, \varphi_k^j(\mathbf{x}, \mathbf{u})$, is equal to a $\theta_k(\mathbf{x}, \mathbf{u})$ in the library then (44), it can be written as

$$e_j = \sum_{r=1}^{K} \theta_r(\mathbf{x}, \mathbf{u}) \big(\xi_{r,j} - \alpha_{r,j}\big) + \sum_{r=K+1}^{L} \xi_{r,j} \theta_r(\mathbf{x}, \mathbf{u}). \tag{45}$$

For minimizing the error for arbitrary $(\mathbf{x}, \mathbf{u})$

$$\begin{cases} \xi_{r,j} = \alpha_{r,j}, & \text{for} \quad r = 1, \cdots, K \\ \xi_{r,j} = 0, & \text{for} \quad r = K+1, \cdots, L. \end{cases} \tag{46}$$

which results in a sparse column $\boldsymbol{\xi_j}$. This implies that if φ_k^j, $k = 1, \cdots, K$ are in the library of functions, the column $\boldsymbol{\xi_j}$ will be sparse. By the logic of contra-positive, if the column $\boldsymbol{\xi_j}$ is dense, then there exists at least one function φ_k^j that doesn't exist in the library of functions and the library has to be updated. □

This iterative process of updating the library as elucidated in Algorithm 1, is repeated until all the columns of matrix $\boldsymbol{\Xi}$ are sparse.

Remark. *The condition $L \gg K$ is imposed to guarantee that the library's size is sufficiently large for the coefficients to have the potential of sparsity. Without satisfying this condition, the library may encompass all required terms, yet the coefficient column could turn out to be dense. Even in that case, by updating the library, the size of the library will increase and the sparsity would appear given that all the terms include in the library.*

4.3. Data-Driven unscented Kalman filter implementation

Following identifying the model of the generator by Algorithm 1, the noisy output y of the generator is observed and the states are estimated by implementing the proposed data-driven UKF via the following steps

1. Start the identification phase,
2. Excite the inputs of the generator and collect data set for $\mathbf{x}$ and $\dot{\mathbf{x}}$ and $\mathbf{u}$ in each sampling time,
3. Improve the reliability of the data by a Savitzky-Golay filter,
4. Construct matrices $\boldsymbol{X}$, $\dot{\boldsymbol{X}}$ and $\boldsymbol{U}$ in the form of (14),
5. Start Algorithm 1 to find a sparse $\boldsymbol{\Xi}$ with a minimum number of nonlinear functions in the library $\boldsymbol{\theta}(\mathbf{x}, \mathbf{u})$,
6. Write the state space of model as $\dot{\mathbf{x}}^T = \boldsymbol{\theta}(\mathbf{x}, \mathbf{u})\boldsymbol{\Xi}$,
7. Construct sigma points as in (22) and propagate them to the next time step by the identified model,
8. Predict the mean and covariance of $\mathbf{x}$ and cross covariance between $\mathbf{x}$ and $\mathbf{y}$ by (28)-(31),
9. Update the estimation by Kalman filter equations in (32)-(34).

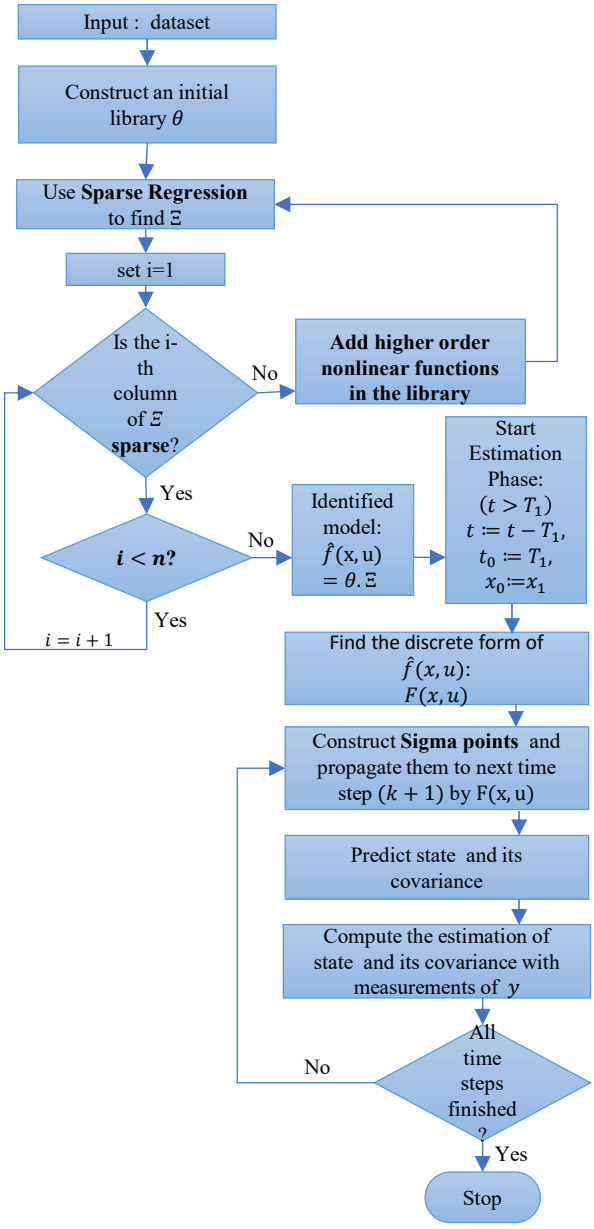

Figure 2: Schematic of the algorithm for density-guided sparse regression unscented Kalman filter.

The schematic of the proposed method is presented in Fig. 2. The excitation of the system plays a crucial role in gathering sufficient data to accurately capture the dynamics of the system and drive the underlying governing equations. Moreover, it is important that the identification phase occurs during the transient state of the system dynamics when the system is exhibiting dynamic behavior.

5. Case Studies

This section presents the application of the proposed method to a single generator in a model-free context. The simulations are executed on a host computer equipped with an Intel(R) Xeon(R) Silver 4214R CPU @ 2.40 GHz and 64 GB RAM. Subsequently, the implementation to a physics-based model is extended to effectively address parameter uncertainty. Next, the proposed approach is applied to a 30-bus system with and without fault scenarios. For the identification phase, data collection is performed from state variables and their corresponding time derivatives over a duration of $T_1 = 2s$ with $5ms$ sampling time. The sparsity tuning factor is established as $\gamma = 20$. The sinusoidal excitation is applied to all the inputs of the generator to enhance the observability of the system during the identification phase. By introducing small perturbations with a magnitude of 0.01 and a frequency of 70 Hz, the dynamic response of the system becomes more noticeable, allowing the algorithm to more accurately capture the system's behavior. The normalized estimation error is defined as $\frac{1}{n'}\sum_{i=1}^{n'}\frac{\|z-\hat{z}\|}{\|z\|}$, where z is the true value of the variable z, $\hat{z}$ is the estimated value and n' is the number of data points. This criteria will be used to evaluate the estimation accuracy in simulation results.

5.1. Single generator: Model-free

The initial condition of the state is set as $\mathbf{x}_0 = [1, 0, 0, 0]^T$ and the Gaussian process and measurement noise are implemented with time invariant covariance as $Q_k = 10^{-2} \cdot \mathbb{I}_{n\times n}$ and $R_k = 10^{-2} \cdot \mathbb{I}_{m\times m}$, where $n = 4$ and $m = 1$. The system's single output measurement corresponds to the output power, which has been shown to be sufficient for DSE in similar works [24]. Additionally, the system inputs u_1, u_2, u_3 are also directly measured, enabling a comprehensive estimation of all state variables. It is also noted that usually in power systems, direct measurements of other states of generators (such as δ or $\Delta\omega$) might not be available to the power system operator. Therefore, it

Algorithm 1 Proposed Method: Density-Guided Sparse Regression

Data: $\mathbf{U}, \mathbf{X}, \dot{\mathbf{X}}$
Result: $\boldsymbol{\Xi}$
Assume an exact representation in (41) with finite l.
Construct a set with different levels of nonlinear functions: $\boldsymbol{H} = \{\boldsymbol{\eta}_1(\mathbf{x}, \mathbf{u}), \cdots, \boldsymbol{\eta}_c(\mathbf{x}, \mathbf{u})\}$
Start with an initial library of function: $\boldsymbol{\theta}(\mathbf{x}, \mathbf{u}) \subset H$
for *q=1:c-1* **do**
  ▷ `comment: the iteration for various levels of nonlinear subsets within` H `commences.`
  Find $\boldsymbol{\Xi} = \boldsymbol{\Theta}(\boldsymbol{X}, \boldsymbol{U}) \backslash \dot{\boldsymbol{X}}$;
  for *j=1:n* **do**
    $\mu_j \leftarrow \max(\boldsymbol{\Xi}(:, j))$
      ▷ `comment:` μ_j `is the dominant element in` j`-th column.`
    Set γ
      ▷ `comment:` γ `is the sparsity tuning factor.`
    for *i=1:m* **do**
      if $|\boldsymbol{\Xi}(i, j)| < \mu_j / \gamma$ **then**
        $\boldsymbol{\Xi}(i, j) = 0$
          ▷ `Removes the non-dominant elements while retaining the features.`
      end
    end
  end
  for *j=1:n* **do**
    if Ξ *is sparse* **then**
      j=j+1
    else
      $\boldsymbol{\theta}(\mathbf{x}, \mathbf{u}) = [\boldsymbol{\theta}(\mathbf{x}, \mathbf{u}), \boldsymbol{\eta}_{q+1}(\mathbf{x}, \mathbf{u})]$;
        Find $\boldsymbol{\Xi} = \boldsymbol{\Theta}(\boldsymbol{X}, \boldsymbol{U}) \backslash \dot{\boldsymbol{X}}$;
        j=n+1; ▷ `comment: to exit the local loops and repeat the first loop to find new` $\boldsymbol{\Xi}$
    end
  end
end

Table 3: Sparse regression UKF notations

Notation	Description
$\mathbf{x}$	State variables of generator
n	Number of state variables
m	Number of time steps in identification phase
$\boldsymbol{f}$	Physics-based transition function
$\hat{\boldsymbol{f}}$	Identified transition function
$\boldsymbol{X}, \boldsymbol{Y}, \dot{\boldsymbol{X}}$	Data set of states, inputs and states time derivative
γ	Sparsity tuning
N	Sparsity threshold
$\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{u})$	a set of nonlinear functions of x and u
$\boldsymbol{\eta}(\boldsymbol{x}, \boldsymbol{u})$	a subset of nonlinear functions of x and u
$\boldsymbol{\theta}(\boldsymbol{x}, \boldsymbol{u})$	a library of nonlinear functions
Δt	Sampling time interval
$\boldsymbol{F}$	Discrete transition function
$\boldsymbol{g}$	Measurement function
$\boldsymbol{G}$	Discrete measurement function
$\boldsymbol{w}$	Process noise
$\boldsymbol{v}$	Measurement noise
$\boldsymbol{Q_k}$	Process noise covariance
$\boldsymbol{P_k}$	Measurement noise covariance
$\hat{\boldsymbol{x}}_k^+$	Estimated mean of the state at time step k
$\boldsymbol{P}_k^+$	Estimated covariance of the state at time step k
$\tilde{\boldsymbol{x}}_k^{(i)}$	the $(i+1)$-th sigma point at time step k
$\hat{\boldsymbol{x}}_k^-$	Predicted mean of the state for time step k
$\boldsymbol{P}_k^-$	Predicted covariance of the state for time step k
K	Kalman gain

is realistic to consider power measurement, which is widely available through existing smart metering systems.

5.1.1. Initial library

As described in Algorithm 1, the identification begins by selecting an initial library of functions as $\theta^0(\mathbf{x}, \mathbf{u}) = [\mathbf{u}^T, \mathbf{x}^T, \cos^T \mathbf{x}, \sin^T \mathbf{x}]$, which results in a dense Ξ with a corresponding function for each row in Table 4. The only sparse column in $\boldsymbol{\Xi}$ is the first column, indicating that the term x_2 is a sufficient representative for the time evolution of x_1. Having dense columns in $\boldsymbol{\Xi}$ indicates that the functions in the library of functions are not sufficient enough to represent the time evolution of remaining states. Therefore, a new library of functions is needed.

5.1.2. Updating the library

In the second iteration, a new library is selected that encompasses higher-order nonlinearities beyond those considered in the initial library $\theta^1(\mathbf{x}, \mathbf{u}) = [\theta^0(\mathbf{x}, \mathbf{u}), \eta_1(\mathbf{x}, \mathbf{u}), \eta_2(\mathbf{x}, \mathbf{u})]$, where $\eta_i(\mathbf{x}, \mathbf{u})$ is a vector encompassing all feasible combinations of the variables u_i and x_j, characterized by the forms

$$\begin{aligned} \eta_1(\mathbf{x}, \mathbf{u}) &= [u_i \sin(x_j), x_i \sin(x_j), u_i \cos(x_j), x_i \cos x_j], \\ \eta_2(\mathbf{x}, \mathbf{u}) &= [u_i \sin(2x_j), x_i \sin(2x_j), u_i \cos(2x_j), x_i \cos 2x_j]. \end{aligned} \tag{47}$$

where $u_i \in \{u_1, \cdots, u_d\}$ and $x_j, x_k \in \{x_1, \cdots, x_n\}$. The updated library in (47), will result in a new $\boldsymbol{\Xi}$, in which the first, third and the fourth column is sparse. However, the second column is still dense which indicates the need to update the library by more iterations. By third iteration, the library is updated as $\theta^2(\mathbf{x}, \mathbf{u}) = [\theta^1(\mathbf{x}, \mathbf{u}), \eta_3(\mathbf{x}, \mathbf{u}), \eta_4(\mathbf{x}, \mathbf{u})]$,

$$\begin{aligned} \eta_3(\mathbf{x}, \mathbf{u}) &= [u_i\, x_j \sin(x_k), u_i\, x_j\, \cos(x_k)] \\ \eta_4(\mathbf{x}, \mathbf{u}) &= [u_i^2 \sin(2x_j), x_i^2 \sin(2x_j), u_i^2 \cos(2x_j), x_i^2 \cos 2x_j]. \end{aligned} \tag{48}$$

This new library of functions results in the $\boldsymbol{\Xi}$ in Table 5, where all the columns are sparse and the dynamics match the actual dynamics.

Employing the identified model and observing noisy y from (7), the states of the system are estimated via unscented Kalman filter.

5.1.3. Selection of γ

The sparsity tuning parameter γ plays a crucial role in determining the trade-off between model complexity and accuracy in sparse regression. To select an optimal value for γ, experiments were conducted using different

Table 4: Identified dense Ξ with insufficient library.

Ξ	$\dot{x}_1$	$\dot{x}_2$	$\dot{x}_3$	$\dot{x}_4$
u_1	0	0	0	0
u_2	0	-0.94	-81.96	20.76
u_3	0	0	41.79	10.16
x_1	0	-0.54	8.70	-36.66
x_2	377	0	-24.33	115.18
x_3	0	0	-14.85	-26.71
x_4	0	3.67	-52.04	150.56
$\cos x_1$	0	0	34.69	0
$\cos x_3$	0	0	5.60	-26.53
$\cos x_4$	0	0.53	-5.12	20.92
$\sin x_1$	0	0.63	-3.99	66.16
$\sin x_3$	0	0	0	0
$\sin x_4$	0	-3.74	44.94	-193.54

values: $\gamma = 5, 15, 20, 25$, and 30. The selection was based on minimizing the normalized error of state variables within a time frame of 0.5 seconds. The results in Fig.3 indicate that a low value of $\gamma = 5$ leads to an under-identified model, where key elements from the library are excluded. This exclusion introduces a significant error between the true state variables and the estimated values, suggesting that $\gamma = 5$ is too small to capture the essential dynamics of the system. As γ increases to 15 and 20, the error decreases significantly, and the model accurately captures the underlying system behavior. These values are found to be optimal, striking a balance between model sparsity and accuracy, with minimal inclusion of unnecessary elements from the library. However, for higher values of $\gamma = 25$ and 30, the error begins to increase again. This indicates that unnecessary terms from the library are being incorporated into the model, leading to overfitting and reduced efficiency. Therefore, the results suggest that $\gamma = 15$ or 20 is optimal for this study. In summary, the selection of γ is a critical process that directly influences the sparsity and accuracy of the final model. The results demonstrate that while smaller values may omit important dynamics, larger values may introduce unnecessary complexity. Hence, an optimal range of $\gamma = 15$ to $\gamma = 20$ achieves the best performance in this case.

Table 5: Identified sparse Ξ with sufficient library of functions.

Ξ	$\dot{x}_1$	$\dot{x}_2$	$\dot{x}_3$	$\dot{x}_4$
u_1	0	**0.076**	0	0
u_2	0	0	**7.63**	0
u_3	0	0	0	0
x_1	0	0	0	0
x_2	**377**	**-0.0038**	0	0
x_3	0	0	**-20.49**	0
x_4	0	0	0	**-76.33**
$u_1 \sin x_4$	0	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$u_3 \sin x_1$	0	0	0	**52.75**
$u_1 \cos x_1$	0	0	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$u_3 \cos x_1$	0	0	**34.30**	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$u_3\, x_3\, \sin x_1$	0	**-0.20**	0	0
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
$u_3^2 \sin(2x_1)$	0	**0.07**	0	0

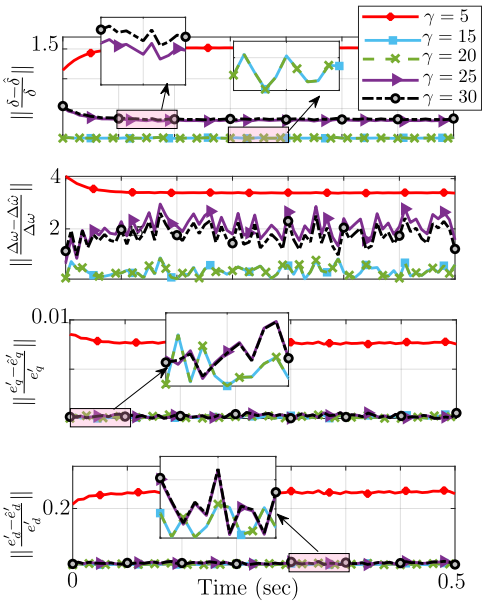

Figure 3: Impact of the sparsity tuning parameter γ on the normalized error of state variables over a time period of 0.5 seconds. The results show that lower values of γ lead to under-identification of the model, while higher values introduce unnecessary complexity. Optimal performance is achieved with γ in the range of 15 to 20, balancing model accuracy and sparsity.

5.1.4. Uncertainty in the model

In this case, it is assumed that the dynamic of the model is known as in (3). However, there exists uncertainty about the parameters. Based on the physics of the model, the library of function is selected as

$$\theta(\mathbf{x}, \mathbf{u}) = \\ [u_1, u_2, x_2, x_3, x_4, u_3 \sin x_1, u_3 \cos x_1, u_3^2 \sin(2x_1), u_3 x_3 \sin x_1] \tag{49}$$

Without loss of generality, it is assumed the uncertainty is in the parameter x_d, i.e.,

$$x_d = \bar{x}_d + \Delta x_d, \quad |\Delta x_d| \leq D_{x_d} \tag{50}$$

where $\bar{x}_d = 2$ and $D_{x_d} = 0.7$. Although the true value of x_d is 2.05, physics-based UKF estimates the state variables assuming $x_d = 2$. It is illustrated in Fig. 4 that, while the overall estimation error of all the state variables using the proposed method is 0.18%, the estimation error for the physics-based UKF is significantly higher at 10%. To further validate the effectiveness of the proposed sparse regression UKF approach, its performance has been compared with the model-based EKF (Extended Kalman Filter), which is another well-recognized method for dynamic state estimation in power systems and assumes prior knowledge of the exact system model. The results of this comparison indicate that both methods successfully estimate the system states; however, the sparse regression UKF demonstrates a notable advantage by not requiring precise information about the system's underlying model. Moreover, in terms of precision, the sparse regression UKF outperforms the EKF, achieving a mean square error (MSE) of 0.18%, compared to 0.27% for the EKF. This reduction in error underscores the enhanced precision of the sparse regression UKF in handling the nonlinear and time-varying dynamics of the system without the constraints of model dependence. These findings highlight the robustness and adaptability of the proposed approach, making it particularly suitable for real-world applications in dynamic state estimation.

5.2. Computational and memory efficiency

In this case study, the computational efficiency of the proposed density-guided sparse regression method is compared to the conventional sparse regression approache. In case 1, the nonlinear space H is defined in (35), which

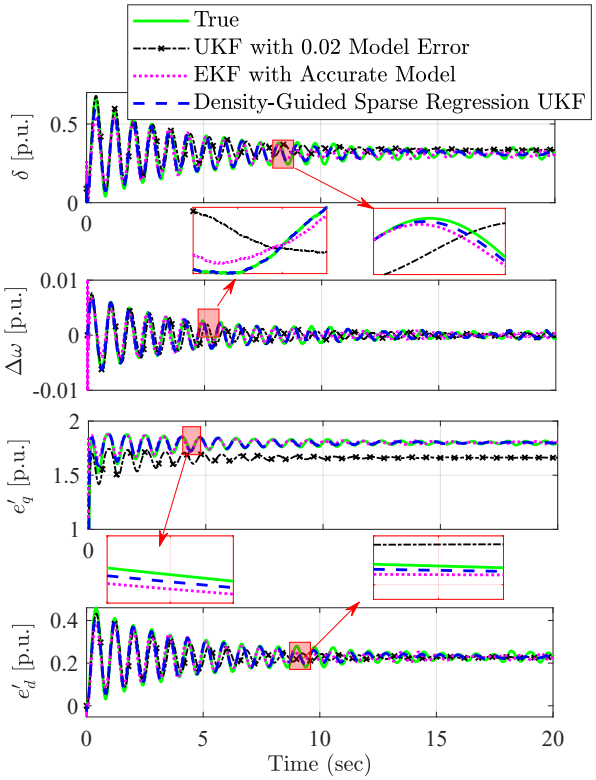

Figure 4: In contrast to physics-based UKF estimation which diverts from the true value, density-guided sparse regression UKF accurately estimates the true value of state variables.

includes functions characterized by fourth-degree nonlinearities. The traditional sparse regression method, as described in [26], typically utilizes all functions within this space H. In Case 2, the nonlinear space H extends to fifth-degree nonlinearities, while in Case 3, it encompasses sixth-degree nonlinearities. In contrast, our density-guided sparse regression method does not require the inclusion of all functions of the space H. Instead, the proposed methodology starts with a library consisting of lower-order functions and iteratively refines the selection over three iterations, only incorporating the necessary higher-order nonlinearities as needed. Table 6 presents a detailed comparison of computational efficiency between the proposed method and the various cases mentioned. Specifically, it evaluates the run time as a measure of computational efficiency and examines the sizes of the libraries θ in (13) and Θ in (16) to highlight the memory efficiency of the proposed approach. As shown in Table 6, this leads to a substantial reduction in run time (down to 3.31 seconds in the same case) and a much smaller library size compared to conventional sparse regression (Case 3), improving both computational and memory efficiency. Thus, this iterative selection process ensures that the method avoids the inefficiency of using the entire library, while still effectively capturing the dynamics of the system.

Table 6: Comparison of computational and memory efficiency of the proposed method.

	Proposed Method	Case 1	Case 2	Case 3
Run Time [sec]	3.31	6.83	11.11	18.14
Size of Library	225	$2,955$	$35,715$	$396,075$
Size of Θ	45×10^3	591×10^3	714×10^4	792×10^5

5.3. Effect of parameter variation

This case studies the effect of time-varying parameters of the model during the steady-state operation of the synchronous generator. At time 30 seconds, the generator's reactance x_d changes from 2.06 to 2.50. Such a change could normally happen during transients, causing a sudden shift in the state variables of the generator. The physics-based UKF and density-guided sparse regression UKF are compared in Fig. 5. As can be observed, the physics-based UKF cannot track the true value of the state variables because the parameter change was not included in the physics-based model that the UKF used. However, the proposed density-guided sparse regression UKF can accurately identify and update the model online at the onset

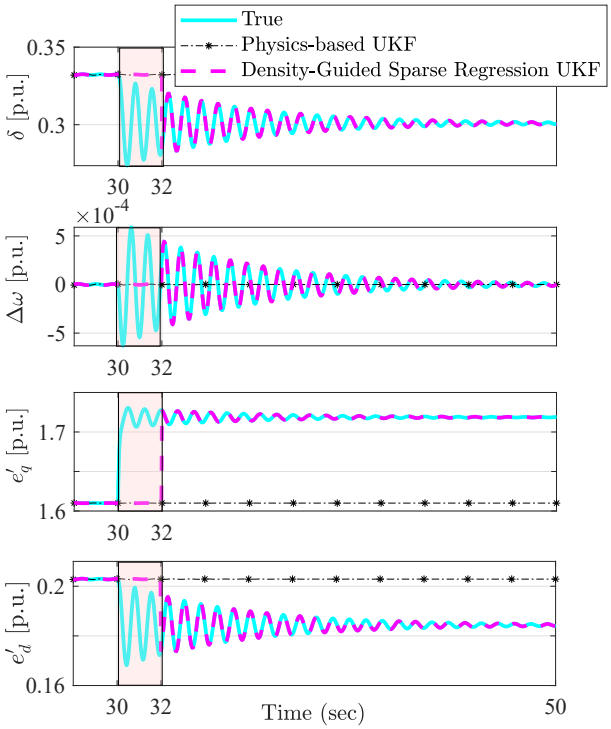

Figure 5: Impact of parameter change on accuracy of proposed density-guided sparse regression UKF that can update the model online and provide an accurate estimation of dynamics in real-time.

of the transient. The model identification process requires time to adapt to changes; therefore, while the initial identification can be considered offline for this duration, the methodology then transitions to online state estimation. It takes approximately 2 seconds from 30 seconds to 32 seconds to collect the data, retrain the proposed model identification, and update the model for the UKF. After this adaptation period, the UKF estimation continues with the updated model. The physics-based UKF has a normalized estimation error of 6.25%, rendering the estimation unreliable. In contrast, the true value of state variables can be successfully estimated by the density-guided sparse regression UKF with a normalized error of 0.24%.
It is important to note that during parameter changes, the library of functions remains fixed, and no additional iterations are needed for model identification. Instead, the method adapts by updating the coefficients in the model, allowing it to capture the system's altered dynamics. This explains the additional two-second delay in response after a parameter change.

5.4. IEEE 30*-bus system*

For a large scale implementation of the proposed data-driven DSE method, a 30-bus power system is modeled, as depicted in Fig. 6. The input parameters for high-lightened generators are assumed to be $\boldsymbol{U_1} = \boldsymbol{U_3} = [0.8, 0.5, 0.98]$, $\boldsymbol{U_2} = [0.8, 0.7, 0.98]$, $\boldsymbol{U_4} = [0.8, 0.7, 0.7]$. By integrating the density-guided sparse regression approach, the true underlying dynamics of generators are identified. In essence, this illustrative demonstration underscores the utility of the density-guided sparse regression UKF in effectively characterizing generator dynamics within a power system. The generator 1 and 3 is the same generator identified in previous case with identifying matrix $\boldsymbol{\Xi}$ in Table 5 which resulted in the identified state space in (47). After implementing density-guided sparse regression, the matrix $\boldsymbol{\Xi}$ for generator 2 and generator 4 is in Table 7 and 8 which result in their corresponding state spaces. State-estimation results for the line 2 current are shown in Fig.7. While a physics-based model UKF struggles to accurately track system uncertainties, state variables, and line currents (with overall 36% normalized estimation error), the proposed density-guided sparse regression UKF effectively overcomes this limitation (with overall 0.12% normalized estimation error), successfully monitoring and adapting to uncertainties in the system parameters.

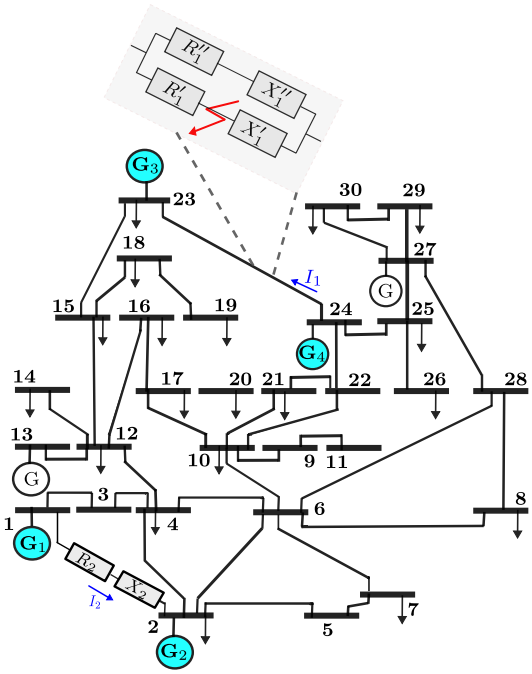

Figure 6: IEEE 30-bus system.

Table 7: Identified sparse Ξ for generator 2.

G2	$\dot{x}_1$	$\dot{x}_2$	$\dot{x}_3$	$\dot{x}_4$
u_1	0	**0.076**	0	0
u_2	0	0	**7.63**	0
x_2	**376.95**	**-0.0038**	0	0
x_3	0	0	**-35.76**	0
x_4	0	0	0	**-76.33**
$u_3 \sin x_1$	0	0	0	**52.75**
$u_3 \cos x_1$	0	0	**75.01**	0
$u_3^2 \sin(2x_1)$	0	**0.07**	0	0
$u_3\, x_3 \sin x_1$	0	**-0.20**	0	0

Table 8: Identified sparse Ξ for generator 4

G4	$\dot{x}_1$	$\dot{x}_2$	$\dot{x}_3$	$\dot{x}_4$
u_1	0	**0.066**	0	0
u_2	0	0	**7.63**	0
x_2	**376.93**	**-0.0033**	0	0
x_3	0	0	**-31.48**	0
x_4	0	0	0	**-76.33**
$u_3 \sin x_1$	0	0	0	**63.61**
$u_3 \cos x_1$	0	0	**34.30**	0
$u_3^2 \sin(2x_1)$	0	**0.06**	0	0
$u_3\, x_3 \sin x_1$	0	**-0.17**	0	0

Table 9: Parameters of generators in p.u.

Parameter	$G1, G3$	$G2$	$G4$
J	13	13	15
D	0.05	0.03	0.05
T'_{do}	0.131	0.131	0.131
T'_{qo}	0.0131	0.0131	0.0131
x_d	2.06	1.8	3.5
x_q	1.254	1.35	1.5
x'_d	0.375	0.375	0.375
x'_q	0.375	0.375	0.375

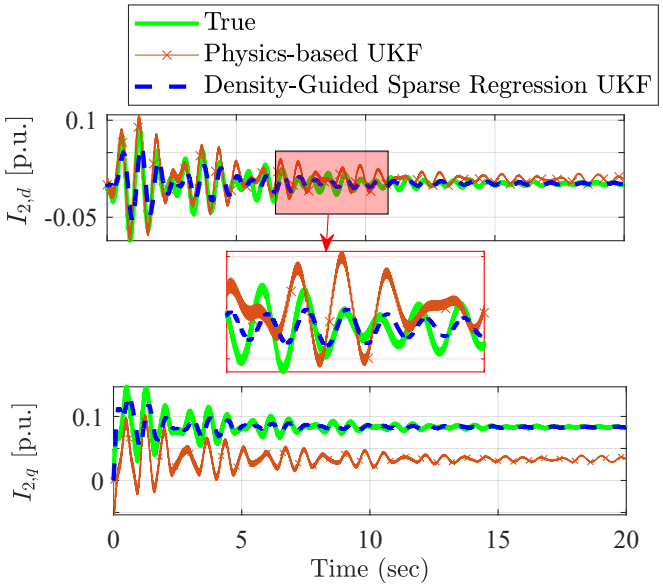

Figure 7: DSE results for line 2 current on after a transient.

5.5. Effect of fault

In this case, the performance of the proposed approach is evaluated in a scenario where a fault manifests at 55-seconds within the 30-bus system. Assume there are two parallel lines connecting node 23 to node 24 in the 30-bus system, the impedance of the corresponding line is the parallel combination of these two impedances. Once a fault occurs on one the lines, the line is isolated from the system and there will remain only one line connecting bus 23 to 24. Therefore, the fault is characterized by the change in the impedance from $5.5 + j22.85 \xrightarrow{t=55s} 20 + j50$. The results provided in Fig. 8 illustrate the accuracy of state estimation using our approach. Employing the density-guided sparse regression unscented Kalman filter, this methodology effectively gauges the system's response to the fault. As a result of the fault occurrence, the line current reduces (due to the increase in line impedance), followed by subsequent oscillations. Significantly, the proposed technique successfully captures the dynamics of line current after the faults. The proposed approach demonstrates its practical utility in enhancing system monitoring in real-world power system scenarios.

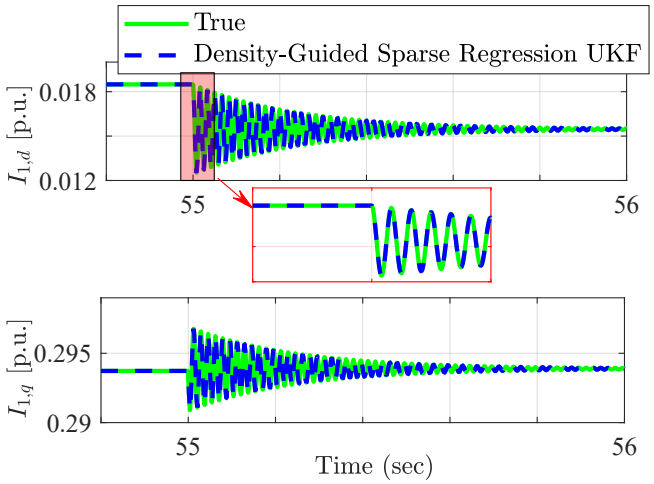

Figure 8: DSE results for Line 1 current after a fault.

5.5.1. Undervoltage fault

Undervoltage conditions frequently occur in power systems due to factors such as increased load, sudden generator disconnection, or transmission equipment failure. In this analysis, an undervoltage fault affects generators 1 and 3, with the terminal bus voltage V_t dropping from 0.9 [p.u.] to 0.85 [p.u.], as shown in Fig. 9. This drop in voltage causes notable changes in the system's line currents. Specifically, there is a reduction in the current of line 1, while the current in line 2 increases. The proposed density-guided sparse regression method effectively handles this fault. Despite the voltage disturbance, the method identifies the fault and adjusts the state estimations for the line currents with minimal error.

5.5.2. Node to ground fault

A node-to-ground short circuit is one of the most critical and common faults in power systems. In this scenario, a fault occurs when the nodes 2 and 23 are grounded, leading to a significant change in line currents. This fault can be triggered by various factors, such as insulation breakdowns or external disturbances, and it affects both system stability and power generation. As illustrated in Fig.10, the density-guided sparse regression method successfully detects the system dynamics' shift caused by the short circuit and adjusts the state estimation in real time. This allows the UKF to maintain high accuracy in its estimations, even in the presence of such a fault.

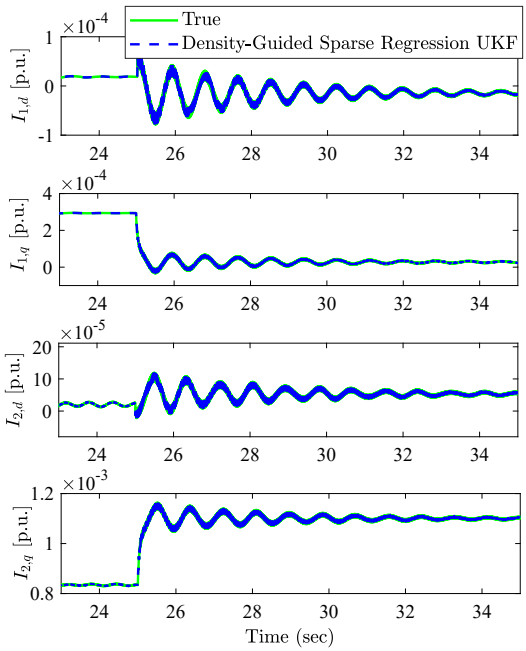

Figure 9: DSE results for line 1 and line 2 current after an undervoltage fault.

The adaptability of the method to sudden changes in system dynamics underscores its resilience in handling critical system disturbances.
Unlike parameter variations, fault scenarios like the short circuit in this example do not require re-identification of the model. The existing function library is sufficient to capture the dynamics, and fault detection is facilitated by changes in line impedance and system inputs, allowing the method to adapt seamlessly without an additional model identification step.

6. Conclusion

A novel data-driven modeling and dynamic state-estimation technique, the density-guided sparse regression unscented Kalman filter, was proposed in this paper. The technique employs a two-phase strategy that includes model identification and state estimation. By defining and utilizing the density of coefficients as a guiding criterion, the proposed sparse regression-based model identification approach iteratively updates the library of linear and nonlinear functions, achieving precise data representation and model identification. The identified data-driven model was then used for dynamic state

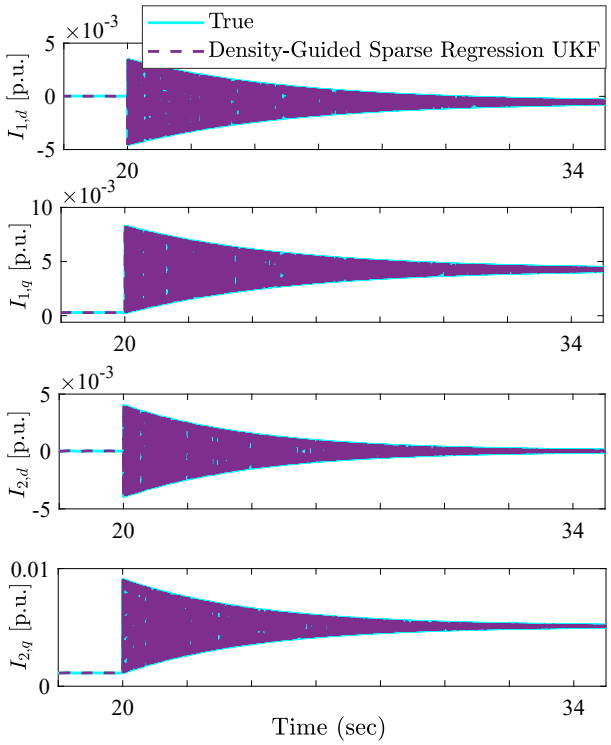

Figure 10: DSE result after node-to-ground fault: where nodes 2 and 23 are both connected to the ground. This short circuit results in a significant change in the line currents, which is efficiently handled by the density-guided sparse regression method.

estimation of a nonlinear power system via an unscented Kalman filtering technique. This data-driven and model-free method is specifically designed to handle the complexities and limited measurements typical of nonlinear generators. The results demonstrate that the method is proficient in accurately estimating states in nonlinear power systems while adapting to model and parameter changes in real-time, without relying on a predefined model. Moreover, the proposed approach has shown notable resilience in the presence of model uncertainties or system-level faults, making it a robust solution for real-world applications.

Acknowledgments

This research was in part under support from the National Science Foundation under Grant NSF-EPCN 2221784. E. Jamalinia, J. Khazaei, and R. S. Blum are with the Electrical and Computer Engineering department at Lehigh University, PA, USA. (E-mails: *elj*320@*lehigh.edu*, *jak*921@*lehigh.edu*, and *rblum*@*eecs.lehigh.edu*)

References

[1] M.-A. Sanchez-Hidalgo, M.-D. Cano, A survey on visual data representation for smart grids control and monitoring, Sustainable Energy, Grids and Networks 16 (2018) 351–369.

[2] N. Dkhili, J. Eynard, S. Thil, S. Grieu, A survey of modelling and smart management tools for power grids with prolific distributed generation, Sustainable Energy, Grids and Networks 21 (2020) 100284.

[3] K. Christakou, A unified control strategy for active distribution networks via demand response and distributed energy storage systems, Sustainable Energy, Grids and Networks 6 (2016) 1–6.

[4] J. Zhao, A. Gómez-Expósito, M. Netto, L. Mili, A. Abur, V. Terzija, I. Kamwa, B. Pal, A. K. Singh, J. Qi, et al., Power system dynamic state estimation: Motivations, definitions, methodologies, and future work, IEEE Transactions on Power Systems 34 (4) (2019) 3188–3198.

[5] I. Alotaibi, M. A. Abido, M. Khalid, A. V. Savkin, A comprehensive review of recent advances in smart grids: A sustainable future with renewable energy resources, Energies 13 (23) (2020) 6269.

[6] A. Alizadeh, I. Kamwa, A. Moeini, S. M. Mohseni-Bonab, Energy management in microgrids using transactive energy control concept under high penetration of renewables; a survey and case study, Renewable and Sustainable Energy Reviews 176 (2023) 113161.

[7] I. Kamwa, Grid behavior: Pmu-enabled dynamic state estimation [editors' voice], IEEE Power and Energy Magazine 21 (1) (2023) 4–9.

[8] E. Ghahremani, I. Kamwa, Dynamic state estimation in power system by applying the extended kalman filter with unknown inputs to phasor measurements, IEEE Transactions on Power Systems 26 (4) (2011) 2556–2566.

[9] G. Anagnostou, B. C. Pal, Derivative-free kalman filtering based approaches to dynamic state estimation for power systems with unknown inputs, IEEE Transactions on Power Systems 33 (1) (2017) 116–130.

[10] P. K. Ray, B. Subudhi, Ensemble-kalman-filter-based power system harmonic estimation, IEEE transactions on instrumentation and measurement 61 (12) (2012) 3216–3224.

[11] G. Pizarro, P. Poblete, G. Droguett, J. Pereda, F. Núñez, Extended kalman filtering for full-state estimation and sensor reduction in modular multilevel converters, IEEE Transactions on Industrial Electronics 70 (2) (2022) 1927–1938.

[12] J. Zhao, L. Mili, A theoretical framework of robust h-infinity unscented kalman filter and its application to power system dynamic state estimation, IEEE Transactions on Signal Processing 67 (10) (2019) 2734–2746.

[13] J. Zhao, L. Mili, A framework for robust hybrid state estimation with unknown measurement noise statistics, IEEE Transactions on Industrial Informatics 14 (5) (2017) 1866–1875.

[14] R. Kandepu, B. Foss, L. Imsland, Applying the unscented kalman filter for nonlinear state estimation, Journal of process control 18 (7-8) (2008) 753–768.

[15] S. Sarkka, On unscented kalman filtering for state estimation of continuous-time nonlinear systems, IEEE Transactions on automatic control 52 (9) (2007) 1631–1641.

[16] J. Schoukens, L. Ljung, Nonlinear system identification: A user-oriented road map, IEEE Control Systems Magazine 39 (6) (2019) 28–99.

[17] G. Kandaperumal, K. P. Schneider, A. K. Srivastava, A data-driven algorithm for enabling delay tolerance in resilient microgrid controls using dynamic mode decomposition, IEEE Transactions on Smart Grid 13 (4) (2022) 2500–2510.

[18] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proceedings of the national academy of sciences 113 (15) (2016) 3932–3937.

[19] Brunton, Steven L and Proctor, Joshua L and Kutz, J Nathan, Sparse identification of nonlinear dynamics with control (sindyc), IFAC-PapersOnLine 49 (18) (2016) 710–715.

[20] B. Huang, J. Wang, Applications of physics-informed neural networks in power systems-a review, IEEE Transactions on Power Systems (2022).

[21] M. Budišić, R. Mohr, I. Mezić, Applied koopmanism, Chaos: An Interdisciplinary Journal of Nonlinear Science 22 (4) (2012) 047510.

[22] A. Mauroy, J. Goncalves, Koopman-based lifting techniques for nonlinear systems identification, IEEE Transactions on Automatic Control 65 (6) (2019) 2550–2565.

[23] J. Hu, Q. Wang, Y. Ye, Y. Tang, Toward online power system model identification: A deep reinforcement learning approach, IEEE Transactions on Power Systems 38 (3) (2022) 2580–2593.

[24] E. Ghahremani, I. Kamwa, Online state estimation of a synchronous generator using unscented kalman filter from phasor measurements units, IEEE Transactions on Energy Conversion 26 (4) (2011) 1099–1108.

[25] P. W. Sauer, M. A. Pai, J. H. Chow, Power system dynamics and stability: with synchrophasor measurement and power system toolbox, John Wiley & Sons, 2017.

[26] S. L. Brunton, J. L. Proctor, J. N. Kutz, Discovering governing equations from data by sparse identification of nonlinear dynamical systems, Proceedings of the national academy of sciences 113 (15) (2016) 3932–3937.

[27] J. Khazaei, R. S. Blum, Model-free distributed control of dynamical systems, International Journal of Information and Communication Engineering 16 (10) (2022) 475–480.

[28] U. Fasel, E. Kaiser, J. N. Kutz, B. W. Brunton, S. L. Brunton, Sindy with control: A tutorial, in: 2021 60th IEEE Conference on Decision and Control (CDC), IEEE, 2021, pp. 16–21.

[29] E. A. Wan, R. Van Der Merwe, The unscented kalman filter for nonlinear estimation, in: Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No. 00EX373), Ieee, 2000, pp. 153–158.

[30] S. J. Julier, J. K. Uhlmann, Unscented filtering and nonlinear estimation, Proceedings of the IEEE 92 (3) (2004) 401–422.

[31] J. Qi, K. Sun, J. Wang, H. Liu, Dynamic state estimation for multi-machine power system by unscented kalman filter with enhanced numerical stability, IEEE Transactions on Smart Grid 9 (2) (2016) 1184–1196.

[32] G. Welch, G. Bishop, et al., An introduction to the kalman filter (1995).