Quantifying Uncertainty in State Estimation: The MoK-FoBS Method via Interval Analysis

Yuting Chen, *Student Member, IEEE*, Ning Zhou, *Senior Member, IEEE*, Ziang Zhang, *Senior Member, IEEE*

***Abstract*— The Bayesian framework is conventionally adopted in power system static state estimation (SSE) to quantify uncertainty via probability density functions (PDFs). However, the reliability of such PDFs is frequently undermined by the complex nature of noise in measurement systems, potentially leading to significant estimation inaccuracies. To address this issue, this paper proposes a modified Krawczyk-forward-backward synthesis (MoK-FoBS) method to quantify uncertainty in SSE through interval analysis. The proposed MoK-FoBS method combines the strengths of the modified forward-backward propagation (FBP) method with the modified Krawczyk method to mitigate the overestimation problem. Employing simulation data derived from IEEE testing systems, it is verified through the Monte Carlo method that the MoK-FoBS method can estimate hard boundaries that invariably contain the true state values. In contrast, the true state values may lie outside the uncertainty boundaries estimated by the weighted least squares approach. A comparative analysis reveals that the MoK-FoBS method can achieve narrower state boundaries than the FBP method, thereby improving estimation precision.**

***Index Terms*— Forward-backward propagation method, Interval analysis, Krawczyk method, Static state estimation.**

I. Introduction

Static state estimation (SSE) integrates measurements and power flow models to determine the operational conditions of a power system and has become an essential tool for guiding real-time operations in a control center [1]. Numerous SSE algorithms have been proposed to estimate bus voltage phasors based on different estimation criteria according to the nature of measurement noises. These include the weighted least squares (WLS) [2], the least median of squares [3], the least trimmed squares [4], the least absolute value [5], and the generalized maximum likelihood [6]. To mitigate the negative impacts of outliers at leverage points, the projection statistics-based algorithm [7], [8] has been designed. Moreover, innovative methods [9], [10] have surfaced to counteract false data injection attacks.

To ensure reliability in power grid operations, it is crucial to effectively quantify the uncertainty of SSE incurred by measurement noises and modeling inaccuracies. Most SSE methods rely on the Bayesian approach to quantify the uncertainty, which assumes that the probability density functions (PDFs) of measurement noises are well-defined and accurately known in advance. Yet, the assumption may not always hold in real-world applications. It is suggested in [11] and [12] that SCADA and PMU measurement noise can deviate from Gaussian distributions due to factors like nonlinear operations, instrument inaccuracy, external electromagnetic interference, communication channels, and cyberattacks. Due to the complex sources of measurement noise, it is often challenging to accurately quantify the noise using a well-defined PDF. When the quality of noise's PDFs is unclear and even questionable, the Bayesian-based SSE methods may become unreliable, potentially leading to significant estimation errors.

The uncertain nature of measurement noise distribution has motivated the application of interval analysis (IA) to quantify uncertainty in SSE. The IA approach quantifies noise uncertainty through boundaries instead of PDFs and is particularly effective in managing the nonlinear and bounded nature of power system measurements, enhancing robustness and reliability in SSE under varied operational conditions. Initiated by Schweppe [13] and further developed by Bertsekas and Rhodes [14], the IA SSE quantifies the uncertainty of the estimated states using boundary ranges and guarantees that the true values reside within those boundary ranges. In particular, Brdys and Chen introduced the set bounded state estimation [15], assuming that measurement errors are unknown but fall within a bounded range [16]. Qi et al. [17] proposed a power system set membership state estimator in a bounded-error context applying the forward-backward propagation (FBP) method. An interval state estimation algorithm based on WLS is proposed in [18] to estimate the states of distribution systems. However, this algorithm risks divergence when the interval gain matrix, which requires inversion, contains singular matrices. The guaranteed boundary ranges of the IA SSE are frequently preferred over a single 'optimal' point estimate, particularly in analyzing worst-case scenarios. Although boundary noise can also be described by truncated PDFs, the computational effort required to accurately adjust and maintain these distributions in high-dimensional SSE makes them less appealing for real-time applications. In contrast, interval analysis (IA) methods are computationally simpler and offer guaranteed inclusion of true states, thus making them the preferred choice to quantify uncertainty in SSE.

This paper is based on work supported by the National Science Foundation under grant #1845523 and by the U.S. Department of Energy's Office of Energy Efficiency and Renewable Energy (EERE) under the Solar Energy Technologies Office Award Number DE-EE0009341. The opinions, findings, conclusions, and recommendations expressed in this paper are those of the authors and do not necessarily reflect the views of the funding agencies.

Yuting Chen, Ning Zhou, and Ziang Zhang are with the Department of Electrical and Computer Engineering, Binghamton University, State University of New York, Binghamton, NY, 13902, USA (email: {ychen411, ningzhou, ziang.zhang}@binghamton.edu).

One major challenge in employing IA in SSE is the overestimation problem [19], where the estimated uncertainty range becomes much wider than the true uncertainty range. In the IA-based SSE, dependence problems and wrapping effects can lead to excessive overestimation, which can result in resource misallocation, increased operation costs, and reduced trust in the IA estimates. To mitigate the negative impact of the overestimation, the set inversion via IA (SIVIA) method has been applied in SE [20]. However, the low computational efficiency of the SIVIA method makes it impractical for real-time applications in power systems.

To address the challenge, a modified Krawczyk-forward-backward synthesis (MoK-FoBS) method is proposed for the power system SSE in this paper. The proposed MoK-FoBS method can accommodate the boundaries of SCADA and PMU measurement noise specified by the IEEE standard [21]. In addition, the outputs of the MoK-FoBS method are the upper and lower bounds of the estimated states, which are well-suited for various state monitoring and control strategies. The contributions of this paper can be summarized as follows:

1) Instead of estimating PDFs under the Bayesian framework, the proposed MoK-FoBS method calculates the hard boundaries that guarantee the inclusion of the true state values under the IA framework.
2) Compared to the conventional FBP method, the proposed MoK-FoBS method can substantially mitigate the overestimation problem in the SSE.
3) Compared to the WLS method [2], the boundaries constructed by the MoK-FoBS method are narrower and are guaranteed to include the true state values.

The rest of the paper is organized as follows: Section II formulates the SSE as an IA problem and discusses the overestimation problem. The modified forward-backward propagation (MFBP) method is proposed to deal with the overestimation problem in Section III. The Krawczyk method is modified for the SSE application in Section IV. The MoK-FoBS method is proposed to combine the MFBP and modified Krawczyk methods in Section V. Case studies are carried out in Section VI. Finally, Section VII draws the conclusions of the study.

II. Problem Formulation

This section begins with a review of the fundamental concepts of IA and conventional SSE problems. It then formulates the SSE as an IA problem and presents the FBP solution as an initial approach to resolving the IA-based SSE problem.

A. Review of Interval Analysis

In IA, an interval variable is defined by its upper and lower boundaries. A real-number interval variable, denoted as $[x]$, is a closed and continuous subset of $\mathbb{R}$ that can be defined by (1) [22]. Here, $\underline{x}$ and $\overline{x}$ are the lower and upper bounds of [x], respectively.

$$[x] = \left[\underline{x}, \overline{x}\right] = \left\{x \in \mathbb{R} \middle| \underline{x} \le x \le \overline{x}\right\}. \tag{1}$$

The width of $[x]$ is defined by (2). The intersection and union of two interval variables are defined by (3.a) and (3.b), respectively. Here, $[\cdot]$ represents the interval hull, which is the smallest interval containing $[x] \cup [y]$ [23].

$$\text{width}([x]) = \overline{x} - \underline{x}. \tag{2}$$

$$[x] \cap [y] = \{z \in \mathbb{R} | z \in [x] \text{ and } z \in [y]\}. \tag{3.a}$$

$$[x] \cup [y] = [\{z \in \mathbb{R} | z \in [x] \text{ or } z \in [y]\}]. \tag{3.b}$$

In IA, real functions are also extended to interval functions. Specifically, let f be a real function mapping from $\mathbb{R}^n$ to $\mathbb{R}^m$, i.e., $f: \mathbb{R}^n \to \mathbb{R}^m$. Its corresponding interval function, denoted as $[f]$, is defined in (4) [23]. Here, $[\mathbf{x}]$ is the real interval vector of $\mathbb{R}^n$, which is an ordered *n*-tuple of intervals, and the notation [.] represents the interval hull—the smallest interval containing the set.

$$[f]([\mathbf{x}]) = [\{f(x_1, \ldots, x_n) | x \in [\mathbf{x}]\}]. \tag{4}$$

The median of an interval is defined by (5).

$$\text{median}([x]) = \frac{1}{2}\left(\overline{x} + \underline{x}\right). \tag{5}$$

Hausdorff's distance is defined in (6) to measure the distance between two interval variables [22].

$$H_d([x], [y]) = max\left(\left|\underline{x} - \underline{y}\right|, |\overline{x} - \overline{y}|\right). \tag{6}$$

B. SSE Problem and the WLS Solution

To perform conventional SSE on a power system with *n* buses, *l* independent branches, and *m* measurements, measurements ($\boldsymbol{z} \in \mathbb{R}^m$) are written as the nonlinear algebraic function ($\boldsymbol{h}$: $\mathbb{R}^{2n-1} \to \mathbb{R}^m$) of state vector ($\boldsymbol{x} \in \mathbb{R}^{2n-1}$) shown in (7). Here, vector $r \in \mathbb{R}^m$ represents measurement noise. Vector $\boldsymbol{x}$ represents power system states, which typically include the bus voltage magnitudes (V_i) and angles (θ_i). P_i and Q_i in $\boldsymbol{z}$ are the measured real and reactive power injection at bus i, respectively. P_{ij} and Q_{ij} in $\boldsymbol{z}$ are the measured real and reactive power flows on the branch from bus i to bus j, respectively. V_i in $\boldsymbol{z}$ represents the measured bus voltage magnitudes.

$$\boldsymbol{z} = h(\boldsymbol{x}) + \boldsymbol{r}. \tag{7.a}$$

$$\boldsymbol{x} = [V_1, \cdots, V_n, \theta_2, \cdots, \theta_n]^T. \tag{7.b}$$

$$\boldsymbol{z} = \left[P_i, Q_i, P_{ij}, Q_{ij}, V_i\right]^T. \tag{7.c}$$

Under a Bayesian framework, the noise $\boldsymbol{r}$ is commonly assumed to follow a Gaussian distribution with zero mean and covariance of $\boldsymbol{R}$. The maximum likelihood estimate of *x* can be found through a WLS solution as in (8.a). To quantify the uncertainty associated with $\widehat{\boldsymbol{x}}_{wls}$, its covariance is estimated through (8.b), where $\boldsymbol{H}$ is the Jacobian matrix of $h(\boldsymbol{x})$ evaluated at $\widehat{\boldsymbol{x}}_{wls}$.

$$\widehat{\boldsymbol{x}}_{wls} = \arg\min_{x}\left(\boldsymbol{z} - h(\boldsymbol{x})\right)^T \boldsymbol{R}^{-1}\left(\boldsymbol{z} - h(\boldsymbol{x})\right). \tag{8.a}$$

$$\boldsymbol{\sigma}^2 = cov(\hat{\boldsymbol{x}}_{wls}) = (\boldsymbol{H}^T\boldsymbol{R}^{-1}\boldsymbol{H})^{-1}. \quad (8.b)$$

C. SSE Problem Formulated under IA framework

Under the IA framework, the SSE can be formulated as constraint satisfaction problems (CSPs), which are defined by (9) [22].

$$\mathcal{H}: (f(\boldsymbol{x}) = 0, \boldsymbol{x} \in [\boldsymbol{x}_0]). \quad (9)$$

$$\text{Here, } \begin{cases} f(\boldsymbol{x}) = \boldsymbol{z} - h(\boldsymbol{x}) - \boldsymbol{r} \\ \mathbf{z} \in [\boldsymbol{z}_0] \quad \boldsymbol{r} \in [\boldsymbol{r}_0] \end{cases}. \quad (10)$$

The symbol $[z_0]$ denotes the initial domain of the measurement vector, which can be set up based on measurement values and noise levels. The symbol $[r_0]$ denotes modeling error, which accounts for the discrepancies between modeled responses and actual behaviors of the real-world power system. The initial domains for state vector $[\boldsymbol{x}_0]$ and modeling errors $[\boldsymbol{r}_0]$ can be established based on engineering judgment.

The solution set of $\mathcal{H}$ is defined in (11). Note that *S* is not necessarily an interval vector. In IA, the interval hull of *S* is considered the optimal solution, as defined by (12).

$$\boldsymbol{S} = \{\boldsymbol{x} \in [\boldsymbol{x}_0] | \mathrm{f}(\boldsymbol{x}) = 0\}. \quad (11)$$

$$[\boldsymbol{x}]' = [\boldsymbol{S}]. \quad (12)$$

The primary objective of IA algorithms is to contract the domain of $\mathcal{H}$ (i.e., $[\boldsymbol{x}]$) to reduce its width. However, due to overestimation problems, these IA algorithms often find a suboptimal solution $[\boldsymbol{x}]$ which satisfies $[\boldsymbol{x}_0] \supseteq [\boldsymbol{x}] \supseteq [\boldsymbol{x}]' \supseteq \boldsymbol{S}$.

D. FBP Method for IA-based SSE

The FBP [24] method is one of the widely used IA methods to solve the CSPs defined in (9). The FBP can be applied to estimate the state vector through two stages iteratively: forward propagation and backward propagation. The implementation details of FBP in solving the SSE problem can be found in [17] and are briefly reviewed as follows so that the paper remains self-contained.

(i) **Initialization:** Set $\boldsymbol{x}_0 = [\underline{\boldsymbol{x}}_0, \overline{\boldsymbol{x}}_0]$ and $k = 0$. Here, $\overline{\boldsymbol{x}}_0$ and $\underline{\boldsymbol{x}}_0$ are the initial upper and lower bounds of the states, respectively. Here, the initial interval $\boldsymbol{x}_0$ should be large enough to cover its true values.

(ii) **Forward propagation stage:** Contract the intervals on the left side of the constraints (7) through $\boldsymbol{z} = \boldsymbol{z} \cap (h(\boldsymbol{x}_k) + \boldsymbol{r})$.

(iii) **Backward propagation stage:** Contract all intervals on the right side of the constraints (7) through $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k \cap h^+(\boldsymbol{z}, \boldsymbol{x}_k)$.

(iv) **Repeat:** Continue repeating (iii) and (iv) until the Hausdorff distance between $\boldsymbol{x}_k$ and $\boldsymbol{x}_{k+1}$ falls below a predetermined threshold. If so, $\boldsymbol{x}_k$ is the suboptimal solution found by FBP. If not, assign $k = k + 1$ and go to step(ii).

In step (i), *x* can be initialized with control and physical limitations. In step (ii), if an intersection on the right side of the equation is empty, the corresponding measurement is treated as bad data and excluded. In step (iii), function h^+ in backward propagation is the pseudo inverse of function h. They are constructed to derive $\boldsymbol{x}_{k+1}$ from $\boldsymbol{z}$ and $\boldsymbol{x}_k$.

E. Challenges of Overestimation Problems.

Although the FBP method provides rigorous uncertainty intervals for state estimation, its calculated interval bounds often turn out to be excessively wider than the actual bounds. This issue, known as the overestimation problem, poses a major challenge in applying IA in SSE. In SSE, the overestimation mainly stems from the dependency problem when the variables in the evaluation expression are interdependent. This is particularly evident when the system states such as V_i and θ_i in (27)-(30) appear multiple times within an evaluation expression. The repeated appearance of the interdependent variables can excessively widen the estimated interval widths. To make the IA-based SSE practical, the overestimation problems need to be addressed.

III. Modified FBP Method for SSE

This section proposes an MFBP method specifically designed to tackle the overestimation issue prevalent in SSE while simultaneously ensuring the computational complexity remains manageable. The proposed MFBP method incorporates four key enhancements, each of which is detailed below.

A. Subdivision Method

The subdivision method [25] is used in the forward propagation stage of the FBP in this paper to reduce the overestimation due to the dependency problem. The reduction is achieved by dividing a broad interval of dependent variables into several small intervals. More specifically, the uniform subdivision illustrated in (13) is used in the paper. Note that the subdivision method cannot be directly applied to the backward propagation stage of the FBP in SSE because it cannot handle the pseudo inverse function h^+.

$$x_{[i]} = [\underline{x} + (i-1)(\overline{x} - \underline{x})/N, \underline{x} + i(\overline{x} - \underline{x})/N]. \quad (13.a)$$

$$h_{(N)}(x) = \bigcup_{i=1}^{N} h(x_{[i]}). \quad (13.b)$$

$$z = z \cap \big(h_{(N)}(x) + r\big). \quad (13.c)$$

B. Subdivision Test

A subdivision test is proposed by the authors to suppress the overestimation in the backward propagation stage of the FBP. The pseudo-code of the subdivision test is summarized below.

Algorithm 1 the subdivision test

```
For i = 1 to N
    x_[i] = [x + (i − 1)(x̄ − x)/N, x + i(x̄ − x)/N]
    If [h(x_[i]) + r] ∩ z ≠ ∅
        x' = x + (i − 1)(x̄ − x)/N
        Break
    End
End
For i = 1 to N
    x_[i] = [x̄ − i(x̄ − x)/N, x̄ − (i − 1)(x̄ − x)/N]
    If [h(x_[i]) + r] ∩ z ≠ ∅
        x̄' = x̄ − (i − 1)(x̄ − x)/N
        Break
    End
End
x = [x', x̄']
```

The width of the test interval is determined by the width of the interval state *x* and the subdivision test number *N*. The value of *N* can be adjusted to strike a balance between convergence speed and precision. A larger value of *N* leads to better precision but slower convergence speed. To achieve a balance between these two factors, *N* is initialized to a small value and gradually increased in subsequent iterations.

C. State Expression and Pseudo Measurements

A major limitation of the subdivision method and subdivision test is that their computational complexity increases exponentially with the number of states to which subdivision is applied. This issue becomes particularly pronounced when calculating the intervals of power injection using bus voltage angles in conventional SSE because both θ_i and θ_j appear multiple times in the expression and cannot be effectively separated. To address the issue, a new set of state expressions and pseudo-measurement equations are introduced for SSE, as detailed below.

1) State expression

To reduce the computational complexity, this paper proposes an alternative state expression, which uses branch voltage angles ($\delta_{ij} \triangleq \theta_i - \theta_j$) as the states, instead of bus voltage angles as the states (θ_i). As a result, the proposed alternative state variables are denoted as an interval vector in (14).

$$\boldsymbol{x}' = [V_1 \quad \cdots \quad V_n \quad \delta_1 \quad \ldots \quad \delta_l]^{\mathrm{T}} \in \mathbb{R}^{n+l} \tag{14}$$

Here, l is the number of independent branches and n is the number of buses. The related measurement functions and Jacobian matrix using the proposed alternative states are detailed in Appendix A.

2) Pseudo-measurements

To ensure equivalence between the SE problem with the alternative state expression and the original state expression, additional pseudo-measurements need to be incorporated. These pseudo-measurements are formulated by adding equations to ensure that the sum of voltage angle differences around a loop is equal to zero. To identify the independent loops that accurately align with the original state expression, a loop identification method is proposed in this subsection based on the singular value decomposition (SVD), as detailed below.

Loops within a power grid can be found by identifying the linearly dependent rows in the branch-to-bus incidence matrix [26]. This matrix, denoted as $\boldsymbol{A} \in \mathbb{R}^{l\times n}$, is defined in (15). To locate the linearly dependent rows, SVD can be utilized, as described in the following pseudo code.

$$\boldsymbol{A}(i,j) = \begin{cases} 1 & \begin{array}{c} if\ bus\ j\ is\ the\ sending \\ terminal\ of\ branch\ i \end{array} \\ -1 & \begin{array}{c} if\ bus\ j\ is\ the\ receiving \\ terminal\ of\ branch\ i \end{array} \\ 0 & otherwise \end{cases}. \tag{15}$$

Algorithm 2 the SVD based loop identification method

1: $[\boldsymbol{U},\boldsymbol{S},\boldsymbol{V}] = svd(\boldsymbol{A})$
2: $r = rank(\boldsymbol{A})$
3: $\boldsymbol{U}_2 = \boldsymbol{U}(:, r+1{:}end)$
4: $\boldsymbol{\mathcal{L}} = rref(\boldsymbol{U}_2^T)$

Here, the SVD reveals that there are (*l*-*r*) linear dependent rows in the $\boldsymbol{A}$ matrix because $\boldsymbol{U}_2^T\boldsymbol{A} = 0$, which indicates that there are (*l*-*r*) independent loops in the systems. The resulting loop-to-branch incidence matrix $\boldsymbol{\mathcal{L}} \in \mathbb{R}^{(l-r)\times l}$ is the reduced row echelon form of $\boldsymbol{U}_2^T$, whose features can be summarized in (16.a). As such, the equation for pseudo-measurements can be summarized in (16.b).

$$\boldsymbol{\mathcal{L}}(i,j) = \begin{cases} 1 & \begin{array}{c} if\ branch\ j\ is\ in\ loop\ i\ and \\ has\ the\ same\ direction\ as\ the\ first\ branch \end{array} \\ -1 & \begin{array}{c} if\ branch\ j\ is\ in\ loop\ i\ and \\ has\ the\ opposite\ direction\ to\ the\ first\ branch \end{array} \\ 0 & otherwise \end{cases}. \tag{16.a}$$

$$\boldsymbol{\mathcal{L}\delta} = 0. \tag{16.b}$$

With the additional pseudo-measurements, measurement equation (7) can be rewritten into (17).

$$\begin{bmatrix} \boldsymbol{z} \\ \boldsymbol{0} \end{bmatrix} = \begin{bmatrix} \boldsymbol{I} \\ \boldsymbol{0} \end{bmatrix} h(\boldsymbol{x}') + \begin{bmatrix} \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{\mathcal{L}} \end{bmatrix} \boldsymbol{x}' + \begin{bmatrix} \boldsymbol{r} \\ \boldsymbol{0} \end{bmatrix}. \tag{17}$$

Here, the symbol $\boldsymbol{I}$ represents an identity matrix, and $\boldsymbol{0}$ denotes a matrix of zeros, appropriately sized.

D. Auxiliary Variables

Subdividing multiple variables within the same expression can result in an exponential increase in computational complexity. To address this issue, auxiliary variables are introduced in SSE. For example, auxiliary variables α_1 and α_2 defined in (18) can be introduced to calculate real power flow on transmission lines P_{ij} in (19).

$$\alpha_1 = g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}. \tag{18.a}$$

$$\alpha_2 = V_j\left(g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}\right) = V_j\alpha_1. \tag{18.b}$$

$$P_{ij} = (g_{si} + g_{ii})V_i^2 - V_i\alpha_2. \tag{19}$$

By introducing auxiliary variables α_1 and α_2, the variables that appear more than once during propagation can be subdivided to suppress overestimation. Additionally, only one subdivision is involved in a single node propagation, which reduces computational complexity.

E. Summary of the proposed MFBP method

The MFBP method for real power flow measurements is illustrated in Fig. 1, while its pseudo code is summarized in the table below. Here, the pseudo-code notation $sub(g(.), x)$ represents the calculation of function $g(.)$ while applying the subdivision method to variable *x*. Similarly, $sub\_tst\ (g(.) = 0, x)$ represents the application of the subdivision test to find a solution for state *x* through the constraint of $g(.) = 0$. The same process is applied to all other measurements. The related equations are detailed in Appendix B.

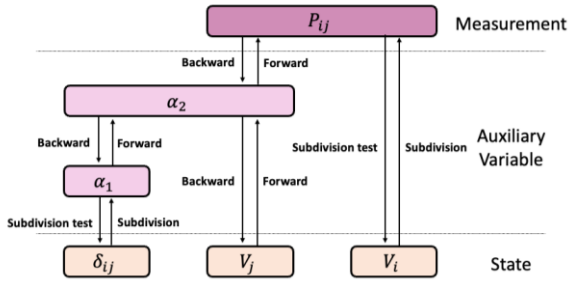

Fig. 1. The MFBP method for the real power flow measurement with auxiliary variables.

Algorithm 3 the MFBP algorithm

Initialization: initialize state with physical limit and engineering knowledge $\boldsymbol{x}_0 = [\underline{\boldsymbol{x}}_0, \overline{\boldsymbol{x}}_0]$, $k = 0$, $qdis = \infty$, $\delta_{ij} = [-\pi/4\ , \pi/4]$, $V_i = [0.9,1.1]$, $V_j = [0.9,1.1]$, $\alpha_1 = [-\infty, \infty]$, $\alpha_2 = [-\infty, \infty]$.

While $qdis > threhold$
  $\boldsymbol{x}_k = \left[\delta_{ij}, V_i, V_j\right]^T$;
  Forward propagation:
  $\alpha_1 = \alpha_1 \cap (\text{sub}(g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}, \delta_{ij}))$
  $\alpha_2 = \alpha_2 \cap (V_j\alpha_1)$
  $P_{ij} = P_{ij} \cap (\text{sub}((g_{si} + g_{ii})V_i^2 - V_i\alpha_2, V_i))$
  Backward propagation:
  $V_i = \text{sub\_tst}(P_{ij} = (g_{si} + g_{ii})V_i^2 - V_i\alpha_2, V_i)$
  $\alpha_2 = \alpha_2 \cap \text{sub}(-(P_{ij} - (g_{si} + g_{ii})V_i^2)/V_i, V_i)$
  $\alpha_1 = \alpha_1 \cap (\alpha_2/V_j)$
  $V_j = V_j \cap (\alpha_2/\alpha_1)$
  $\delta_{ij} = \text{sub\_tst}(\alpha_1 = g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}, \delta_{ij})$
  $qdis = H_d(\boldsymbol{x}_k, \left[\delta_{ij}, V_i, V_j\right]^T)$
  $k = k + 1$
End

IV. Modified Krawczyk Method

To further mitigate the overestimation problem, a grouping method is proposed in this section so that the Krawczyk method [27] can be applied to solve the SSE problems.

A. Review of the Krawczyk Method and its Limitation

Similar to the interval Newton method [27], the Krawczyk method is a nonlinear IA method for solving CSP problems. Compared to the interval Newton method, the Krawczyk method offers a distinct advantage by eliminating the need to calculate the inverse matrix of the interval Jacobian matrix. This is advantageous because the interval Jacobian matrix might contain singular matrices, potentially leading to failures during the matrix inversion. In addition, the Krawczyk method interactively refines the intervals and guarantees convergence to a tighter enclosure. The Krawczyk method is detailed in [27] and briefly summarized below.

Consider a CSP shown in (20). Here, the function $f: \mathbb{R}^{n_f} \to \mathbb{R}^{n_f}$. The principle of the Krawczyk method involves using an interval extension of the function and its derivative to compute a new interval that encloses the zeros of the function. The Krawczyk operator, denoted as $K(\cdot)$, is defined by (21) [27].

$$f(\boldsymbol{x}) = 0. \tag{20}$$

$$K(\boldsymbol{x}) = \boldsymbol{x}_m - \boldsymbol{C}f(\boldsymbol{x}_m) + \big(\boldsymbol{I} - \boldsymbol{C}f'(\boldsymbol{x})\big)(\boldsymbol{x} - \boldsymbol{x}_m). \tag{21}$$

Here, $\boldsymbol{x}_m$ denotes a fixed number and $\boldsymbol{x}_m \in \boldsymbol{x}$. f' represents the Jacobian matrix of the function f. Matrix $\boldsymbol{C}$ is a preconditioning matrix selected by users. The CSP in (20) can be solved by the Krawczyk method, which iteratively applies the Krawczyk operator as shown in (22).

$$\boldsymbol{x}^{k+1} = K(\boldsymbol{x}^k) \cap \boldsymbol{x}^k. \tag{22}$$

This iterative refinement ensures convergence towards a tighter enclosure of the true solution, provided that C is appropriately selected. The selection of C is crucial— improper selection of C can lead to issues such as non-convergence. Following the suggestion in [28], C in this study is chosen to be $f'^{-1}(\boldsymbol{x}_m)$, the pseudo-inverse of the measurement Jacobian matrix at $\boldsymbol{x}_m$.The method and its application are detailed further in [28]. Should non-convergence issues arise, users are encouraged to fine-tune their selection of C.

It is important to note that the Krawczyk method is only suited to solve well-determined systems. However, the SSE of power systems is typically an over-determined system, wherein the number of measurements exceeds the states to be estimated. To bridge the gap, this paper introduces a measurement data grouping algorithm based on SVD. This algorithm decomposes the over-determined SSE problem into multiple well-determined subproblems so that the Krawczyk method can be utilized to solve these well-determined subproblems of SSE. Then, the multiple SSE solutions are merged into a solution through intersection operations.

B. SVD-based Grouping Method

Assume that there are sufficient measurements in the SSE problem to render the system observable and make the SSE an overdetermined problem. The objective of the grouping method is to divide the measurements into several groups so that (a) the measurements in each group shall make the SSE a well-determined subproblem conducive to the application of the Krawczyk Method; (b) each measurement must belong to at least one group, ensuring that the information it contains is utilized to reduce the interval width of the estimated states; (c) the number of measurement groups shall be kept small to reduce computational complexity. Note that this approach permits the inclusion of the same measurement in multiple groups, thus allowing measurements to be shared across different groups. To address the grouping problem, this paper proposes an SVD-based grouping method, which is outlined below.

In Algorithm 4, objective (a) is achieved through steps 1-4 and 7-12. Here, the input $\boldsymbol{H}'$ is the Jacobian matrix of the measurement function, as described in (31)-(48) in Appendix A. The output *Group* contains indices of the measurements that make SSE a well-determined problem. In addition, the space spanned by $\boldsymbol{U}_2$ is orthogonal to the column space of $\boldsymbol{H}$, i.e., $\boldsymbol{U}_2^*\boldsymbol{H} = 0$. As such, any measurement corresponding to a non-zero element in $\boldsymbol{U}_C$ can be excluded while still maintaining system observability because the measurement is redundant and can be expressed as a linear combination of other measurements. To achieve objective (b), the *Group* is updated and stored for every iteration until each measurement is included in at least one group, as implemented in step 5. To achieve objective (c), the coefficient scalar CS in step 14 must be a real number greater than 1 ($CS > 1$). Scaling up the rows in $\boldsymbol{U}_2$ by CS tends to make the following group omit measurements that have been used in previous groups.

Algorithm 4 the SVD grouping method

1: $\boldsymbol{H} = \boldsymbol{H}'([V = 1, \theta = 0])$ % measurement Jacobian
2: $[\boldsymbol{U}, \boldsymbol{S}, \boldsymbol{V}] = svd(\boldsymbol{H})$
3: $[m, n] = size(\boldsymbol{H})$ % m: num of measurements; n: num of states
4: $\boldsymbol{U}_2 = \boldsymbol{U}(:, n+1:m)$ % note that $\boldsymbol{U}_2^*\boldsymbol{H} = 0$
5: **While** (any measurement is not grouped)
6: *Group = 1:m*;
7: **For** *i* = 1:*m-n*
8: $\boldsymbol{U}_c = abs(\boldsymbol{U}_2(:, i))$
9: $[maxU_c, D] = \max(\boldsymbol{U}_c)$
10: $Group = Group(Group \sim = D)$
11: % Omit measurement *D* from this group
12: **End**
13: % Store the *Group* as the output
14: $\boldsymbol{U}_2(Group, :) = \boldsymbol{U}_2(Group, :) * CS$
15: **End**

C. Modified Krawczyk Method for the SSE

To align with the framework presented in (20) and (21), measurement equation (7) is transformed into (23).

$$f(x) = \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix} h'(\boldsymbol{x}') + \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\mathcal{L}} \end{bmatrix} \boldsymbol{x}' + \begin{bmatrix} \boldsymbol{r} \\ \mathbf{0} \end{bmatrix} - \begin{bmatrix} \boldsymbol{z} \\ \mathbf{0} \end{bmatrix} = 0. \quad (23)$$

The measurement functions and their first-order derivatives in the Jacobian matrix are defined in (27)-(30) and (32)-(48) in Appendix A, respectively. The $\boldsymbol{x}_m$ and $\boldsymbol{C}$ needed in (21) are selected through (24) and (25), respectively.

$$\boldsymbol{x}_m = median(\boldsymbol{x}). \quad (24)$$

$$\boldsymbol{C} = \begin{cases} \boldsymbol{H'}^{-1}(\boldsymbol{x}_m) & if\ \boldsymbol{H'}(\boldsymbol{x}_m)\ is\ nonsingular \\ \boldsymbol{H'}^{-1}(\boldsymbol{x}_m + \frac{1}{4}\text{width}([\boldsymbol{x}])) & otherwise \end{cases}. \quad (25)$$

The modified Krawczyk algorithm can be outlined in Algorithm 5. Here, $\overline{\boldsymbol{x}}_0$ and $\underline{\boldsymbol{x}}_0$ are the upper and lower bounds of the initial states, respectively. Typically, the initial interval $\boldsymbol{x}_0$ should be sufficiently large to cover the true values of the states.

Algorithm 5 Modified Krawczyk method for SSE

Initialization: initialize states with physical limit and engineering knowledge $\boldsymbol{x}_0 = [\underline{\boldsymbol{x}}_0, \overline{\boldsymbol{x}}_0]$, $qdis = \infty$, $k = 0$.
Apply the SVD grouping method
While $qdis > threshold$
  $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$
  For $i = 1{:}\, number\ of\ group$
    $\boldsymbol{z}_{group} = \boldsymbol{z}(Group_i)$
    $K(\boldsymbol{x}_{k+1}) = \boldsymbol{x}_{m,k+1} - \boldsymbol{C}f(\boldsymbol{x}_{m,k+1}) + \left(\boldsymbol{I} - \boldsymbol{C}f'(\boldsymbol{x}_{k+1})\right)(\boldsymbol{x}_{k+1} - \boldsymbol{x}_{m,k+1})$
    $\boldsymbol{x}_{k+1} = K(\boldsymbol{x}_{k+1}) \cap \boldsymbol{x}_{k+1}$
  End
  $qdis = H_d(\boldsymbol{x}_k, \boldsymbol{x}_{k+1})$
  $k = k + 1$
End

V. MoK-FoBS Method

It is important to note that all algorithms within the Interval Analysis (IA) framework are rigorous, as highlighted in [22], ensuring that the resulting intervals invariably contain the true values. Capitalizing on this property, this section proposes the MoK-FoBS algorithm. This algorithm combines the estimated intervals from the MFBP and the modified Krawczyk method by executing them sequentially. Such a combination ensures that the MoK-FoBS algorithm achieves estimation intervals that are not only narrower or equal to those obtained from the individual algorithms but also come with guaranteed hard boundaries.

Algorithm 6 MoK-FoBS method

Initialization: initialize states with physical limit and engineering knowledge $\boldsymbol{x}_0 = [\underline{\boldsymbol{x}}_0, \overline{\boldsymbol{x}}_0]$, $k = 0$.
Apply SVD grouping method
While $tolerance > threshold$
  $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k$
  Apply the modified forward propagation method
  Apply the modified backward propagation method
  Apply the Krawczyk operator to all grouped measurements
  $qdis = H_d(\boldsymbol{x}_k, \boldsymbol{x}_{k+1})$
  If $qdis < tolerance$
    $tolerance = tolerance/10$
  End
  $k = k + 1$
End

The procedure for the MoK-FoBS method is detailed above. Initially, the tolerance level is set to a high value to accelerate convergence. This tolerance is gradually reduced over subsequent iterations until it meets a predefined threshold, ensuring precise results.

VI. Case Studies

In this section, the performance of the proposed MoK-FoBS method is evaluated and compared with the original FBP method [16], the MFBP method, the modified Krawczyk method, and the WLS method [2]. All case studies herein utilize the IEEE 14-bus system [29] and IEEE 33-bus system [30] for illustrative clarity. Moreover, the IEEE 39-bus [31] and IEEE 118-bus [32] systems are employed to examine the computational efficiency of the proposed method.

A. Comparison between the MoK-FoBS Method and Other IA-based Methods

In this case, the performance of the MoK-FoBS and FBP is evaluated using the IEEE 14-bus system, which models a transmission system, and the IEEE 33-bus system, which models a distribution system.

1) Comparison using the IEEE 14-bus system

As shown in Fig. 2, the IEEE 14-bus system consists of 20 branches and 14 buses. Assume that the system has 55 measurements, including 14 injection power measurements, 40 power flow measurements, and one bus voltage magnitude measurement on bus 1, which are marked in Fig. 2.

The measurement noise of power flow, power injection, and voltage magnitude are uniformly distributed within the interval of $[-1 \times 10^{-2}, 1 \times 10^{-2}]$. The initial interval states are [0.9, 1.1] for voltage magnitudes and [-π/4, π/4] for branch voltage angles. The modeling error [r_0] is set to [0, 0] because both the simulation and state estimation utilize the same model. To quantify the capability of the proposed MoK-FoBS method in mitigating the overestimation problem, a width improvement (WI) metric is defined in (26).

$$\text{WI} = \frac{\text{width}(\boldsymbol{x}_{FBP}) - \text{width}(\boldsymbol{x}_{MoK-FoBS})}{\text{width}(\boldsymbol{x}_{FBP})}. \quad (26)$$

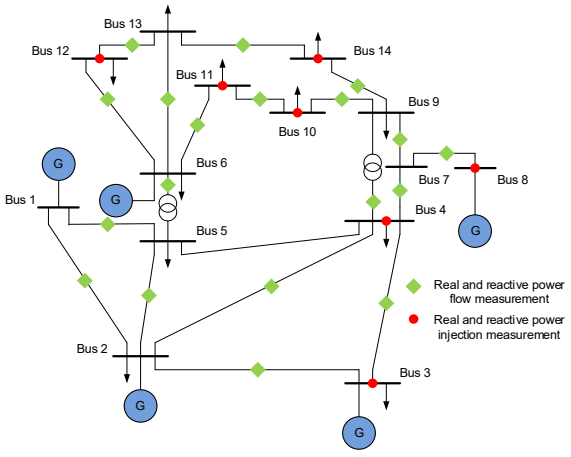

Fig. 2. Topology and measurement setup of the IEEE 14-bus system [29].

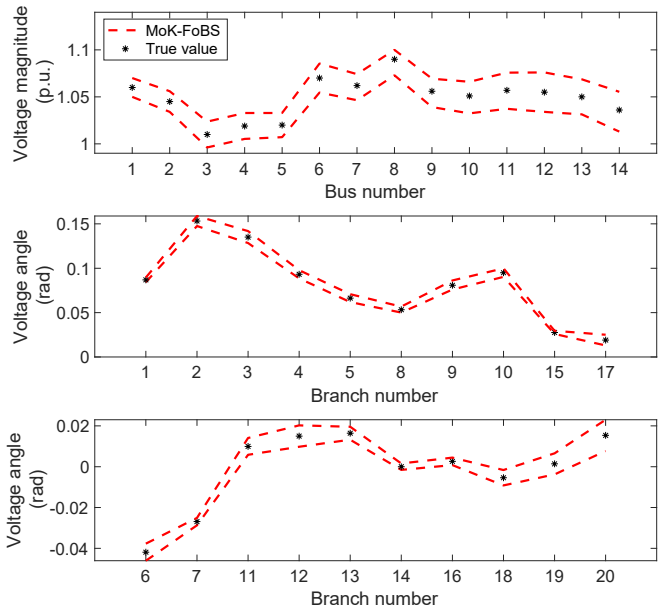

Fig. 3. Interval states estimated by the MoK-FoBS method.

The state estimation results obtained using the proposed MoK-FoBS method are plotted in Fig. 3. It can be observed from Fig. 3 that all true values of the states fall within the estimation intervals generated by the MoK-FoBS method, which verifies that the MoK-FoBS method is a rigorous IA method.

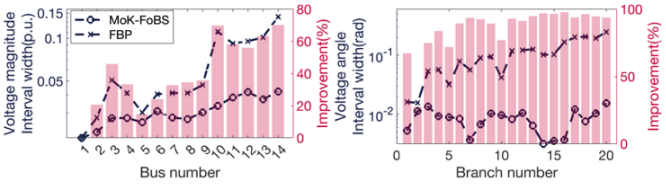

Fig. 4. Comparison of state interval widths estimated by the FBP and MoK-FoBS methods, along with their corresponding WIs.

Furthermore, the interval widths of the estimated results are plotted in Fig. 4. Here, the dashed lines with 'x' marks represent the interval widths from the FBP method, while the dashed lines with 'o' marks represent the interval widths from the proposed MoK-FoBS method. The WIs defined by (26) are also plotted as red bars in Fig. 4. It can be observed from Fig. 4 that the interval widths estimated by the MoK-FoBS method are consistently narrower than those estimated by the FBP method. The improvements are particularly noticeable in voltage magnitude on buses 10-14 and voltage angle differences across almost all the branches (except for branch 2), where improvements exceed 60% in branch voltage angles and 20% in bus voltage magnitudes. These observations indicate that the proposed MoK-FoBS has significantly reduced the negative impacts of the overestimation problem associated with the FBP method.

Moreover, Fig. 5 compares the interval widths of the estimation results obtained from three different methods: the MFBP method (magenta solid line), the modified Krawczyk method (blue dashed line), and the MoK-FoBS method (black dashed line with 'o' marks). It can be observed that the MoK-FoBS method offers the narrowest interval among these algorithms, particularly in the estimated voltage magnitudes for buses 10, 11, and 14, where its intervals are smaller than those of the other two methods. The observation indicates that the MoK-FoBS method effectively reduces the overestimation problem by combining the two methods.

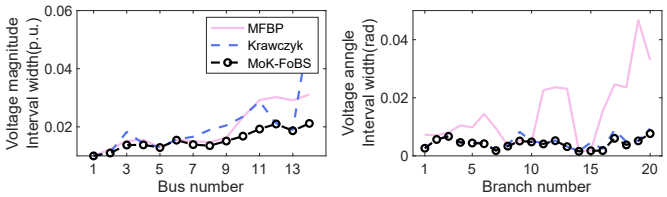

Fig. 5. Comparison of state interval widths estimated by the MFBP method, modified Krawczyk method, and MoK-FoBS method.

2) Comparison using the IEEE 33-bus system

To assess the applicability and effectiveness of the proposed MoK-FoBS method in a distribution system characterized by a radial configuration and heightened uncertainty levels, the IEEE 33-bus system, depicted in Fig. 6, is used to generate simulation data for comparison.

This system includes only six measurements, which measure the voltage magnitudes and power injections at corresponding buses. Additionally, pseudo-measurements of power injection are generated based on load predictions. Virtual measurements, indicating zero power injection, are introduced at nodes without load. The locations of all measurements, pseudo-measurements, and virtual measurements are marked in Fig. 6. The noise associated with measurements, pseudo-measurements, and virtual measurements is assumed to be uniformly distributed within the intervals of [-0.01,0.01], [-0.03,0.03], and $[-1*10^{-6}, 1*10^{-6}]$ p.u., respectively.

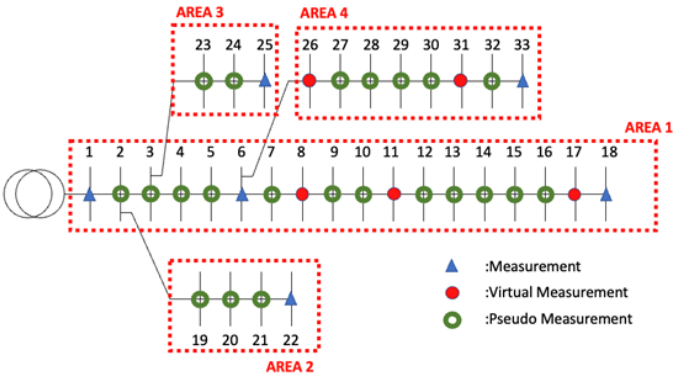

Fig. 6 Topology and measurement setup of the IEEE 33-bus system [30].

State estimation results obtained using the MFBP method, the modified Krawczyk method, and the proposed MoK-FoBS method are illustrated in Fig. 7. It can be observed from this figure that the true values of the states fall within the estimation intervals for all methods employed. Furthermore, the proposed MoK-FoBS method consistently provides the narrowest

interval among these algorithms, demonstrating its superior precision and reliability in handling the challenges of distribution systems.

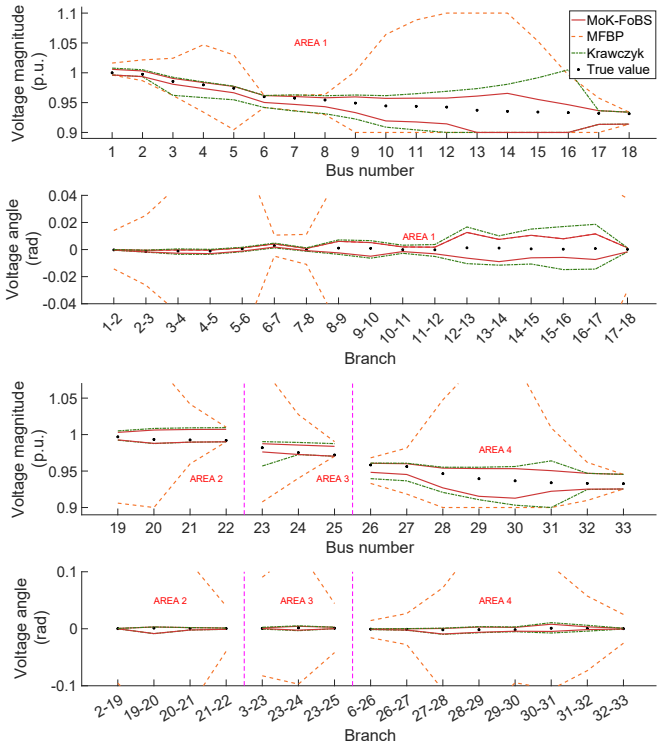

Fig. 7 Interval states estimated by the MFBP, modified Krawczyk, and MoK-FoBS methods.

B. *Comparison between the MoK-FoBS method and the WLS method*

In this subsection, the robustness of the confidence intervals generated by the WLS and MoK-FoBS methods is evaluated and compared in the presence of bounded measurement noise. For the IEEE 39-bus system, it is assumed that there are 20 power injection measurements, 46 power flow measurements, and one voltage magnitude measurement. For the IEEE 118-bus system, it is assumed that there are 118 power injection measurements, 179 power flow measurements, and 59 voltage magnitude measurements. For the IEEE 300-bus system, it is assumed that there are 1318 measurements. These comprise 300 voltage magnitude measurements, 100 pairs of power injection measurements, and 409 pairs of power flow measurements.

The measurement noises are constrained within the range of -1×10^{-2} to 1×10^{-2}. Three types of noise distributions are employed: uniform, truncated Gaussian, and bimodal. The truncated Gaussian distribution is obtained by setting the standard deviation of a normal Gaussian distribution to one-third of the boundary radii, truncating its PDF's long tails outside the boundaries, and normalizing the resulting PDF. The PDF of the bimodal distribution is uniformly distributed at the edges within the noise boundaries, i.e., $[-1\times10^{-2},-0.9\times10^{-2}]\cup[0.9\times10^{-2},1\times10^{-2}]$, as illustrated in Fig. 8.

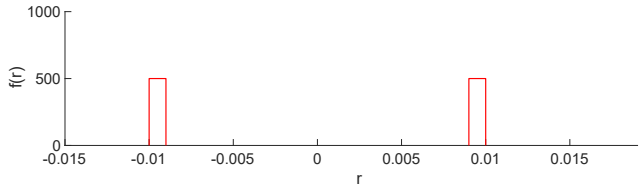

Fig. 8 PDF of the bimodal distribution used in the analysis.

To account for the randomness of the measurement noise, performance metrics are evaluated using the Monte Carlo method across 1,000 instances of measurement noise. Violations are defined as instances where any estimated interval fails to cover the true value of the states. The study results are summarized in Table I. Note that the WLS method constructs confidence intervals as $[\hat{\boldsymbol{x}}_{wls}-c\boldsymbol{\sigma},\hat{\boldsymbol{x}}_{wls}+c\boldsymbol{\sigma}]$, where $\hat{x}_{wls}$ and σ are determined using (8), and c is a user-selected constant.

Table I shows that the MoK-FoBS method consistently yields interval states that encompass the true state values. This underscores the rigor of the IA, suggesting that the algorithms developed under its framework are highly reliable. In contrast, the WLS method exhibits violations up to the 6σ confidence intervals for the truncated Gaussian distribution, up to the 10σ confidence interval for the uniform distribution, and up to the 12σ confidence interval for the bimodal distribution. These observations indicate that the accuracy of the confidence intervals generated by the WLS method is substantially influenced by the PDFs of the noise. Additionally, the confidence intervals generated by the WLS method cannot cover all the true values in the states with a similar interval width as the MoK-FoBS method.

TABLE I
NUMBER OF VIOLATION CASES IN THE 1,000 MONTE CARLO SIMULATIONS

Methods	Average interval width	Noise distribution	Number of violation cases
Proposed MoK-FoBS Method	0.0162	Gaussian	0
		Uniform	0
		Bimodal	0
WLS ($c=6$)	0.0102	Gaussian	26
		Uniform	573
		Bimodal	998
WLS ($c=8$)	0.0136	Gaussian	0
		Uniform	120
		Bimodal	938
WLS ($c=10$)	0.0169	Gaussian	0
		Uniform	7
		Bimodal	593
WLS ($c=12$)	0.0204	Gaussian	0
		Uniform	0
		Bimodal	244

C. Computational Efficiency

To assess its computational efficiency, the proposed MoK-FoBS method was implemented using MATLAB and tested on a PC equipped with an Intel Core i7 processor running at 3.20 GHz and 16 GB of RAM. The computation times are summarized in Tables II and III. Initially, the MoK-FoBS method was implemented using INTLAB [33], a toolbox designed for accurately implementing IA algorithms by overloading MATLAB operators. However, the implementation using INTLAB was found to have low computational efficiency, as shown in Table II.

TABLE II
COMPUTATIONAL TIME USING INTLAB IMPLEMENTATION

Method	Simulation system	Measurement number	Computational time (s)
MFBP	IEEE 14-bus	56	127.14
	IEEE 39-bus	134	419.75
	IEEE 118-bus	654	3604.43
	IEEE 300-bus	1318	11463.98
Modified Krawczyk method	IEEE 14-bus	56	10.37
	IEEE 39-bus	134	44.65
	IEEE 118-bus	654	324.09
	IEEE 300-bus	1318	377.13
MoK-FoBS method	IEEE 14-bus	56	103.21
	IEEE 39-bus	134	357.74
	IEEE 118-bus	654	3138.54
	IEEE 300-bus	1318	11986.10

TABLE III
COMPUTATIONAL TIME USING THE BOUND-FOCUSED CODE IMPLEMENTED BY THE AUTHORS

Method	Simulation system	Measurement number	Computational time (s)
MFBP	IEEE 14-bus	56	0.78
	IEEE 39-bus	134	2.05
	IEEE 118-bus	654	8.34
	IEEE 300-bus	1318	25.07
Modified Krawczyk method	IEEE 14-bus	56	0.64
	IEEE 39-bus	134	2.40
	IEEE 118-bus	654	40.78
	IEEE 300-bus	1318	56.07
MoK-FoBS method	IEEE 14-bus	56	1.04
	IEEE 39-bus	134	3.64
	IEEE 118-bus	654	45.97
	IEEE 300-bus	1318	101.45

To improve implementation efficiency, the authors modified the initial INTLAB implementation code by removing the calculation for rounding errors and focusing solely on bound calculation. Because the rounding errors are negligibly small compared to the measurement noises in the proposed application, this modified approach retains nearly the same accuracy level as the original INTLAB code while significantly boosting computational speed. As shown in Table III, the computational time has dramatically decreased, especially for the MFBP and MoK-FoBS methods.

Table II shows that with the INTLAB implementation, the modified Krawczyk method had the shortest computational time, suggesting that matrix-based methods are more computationally efficient, especially in larger systems. Conversely, the MFBP method exhibited the longest computational time. Table III shows that with the bound-focused implementation, the MFBP method and the Krawczyk method have similar computational time. On the other hand, the MoK-FoBS method has a longer computational time. The computation times for all three algorithms display a near-linear increase in correlation with the system size, suggesting their feasibility for deployment in larger systems.

VII. CONCLUSIONS AND FUTURE WORK

The MoK-FoBS method is proposed in this paper to quantify the uncertainty in power system SSE through IA. This proposed MoK-FoBS method mitigates the overestimation problem by synthesizing two modified IA methods and subsequently offers notably tighter interval estimates than the conventional FBP method. Furthermore, compared to the well-established WLS algorithm, the proposed MoK-FoBS method consistently generates hard boundaries that possess comparable interval widths while ensuring that all true values are encompassed within these intervals. Moreover, the paper highlighted a crucial enhancement in computational efficiency. By modifying the initial INTLAB implementation, the computational speed of the MoK-FoBS method was significantly improved without compromising the accuracy of the results. In summary, the MoK-FoBS method offers an accurate, reliable, and computationally efficient tool for quantifying uncertainty in power system SSE.

Future work may explore further optimization of the method and its application to more diverse and dynamic power system scenarios. In addition, to further narrow the interval width of estimated states, dynamic models will be introduced into state estimation to capture the temporal correlation of power system states. Approaches like the one detailed in [34] will be integrated into the proposed method within the forecast-aided state estimation framework [35].

APPENDIX A. MEASUREMENT FUNCTIONS AND THE JACOBIAN MATRIX FOR THE PROPOSED ALTERNATIVE STATES

To perform IA-based SSE on a power system with n buses, l independent branches, and m measurements, the measurements ($z \in \mathbb{R}^m$) in (7.a) are written as the nonlinear algebraic function ($h' : \mathbb{R}^{n+l} \to \mathbb{R}^m$) of the alternative states ($x' \in \mathbb{R}^{n+l}$) defined in (14). The measurement equations in h' are presented in (27)-(30). Here, g_{ij} and b_{ij} are the real and imaginary parts of the admittance of the series branch connecting buses i and j, respectively. g_{si} and b_{si} are the real and imaginary parts of the admittance of the shunt branch connected to bus i, respectively. G_{ij} and B_{ij} are the real and image parts of the element at the i^{th} row j^{th} column of the bus admittance matrix, respectively. N_i is

the set of buses that are directly connected to bus *i*.

$$P_i = V_i \sum_{j\in N_i} V_j(G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij})\,. \quad (27)$$

$$Q_i = V_i \sum_{j\in N_i} V_j(G_{ij}\sin\delta_{ij} - B_{ij}cos\delta_{ij})\,. \quad (28)$$

$$P_{ij} = V_i^2(g_{si}+g_{ij}) - V_iV_j(g_{ij}cos\delta_{ij} + b_{ij}sin\delta_{ij}). \quad (29)$$

$$Q_{ij} = -V_i^2(b_{si}+b_{ij}) - V_iV_j(g_{ij}sin\delta_{ij} - b_{ij}cos\delta_{ij}). \quad (30)$$

The structure of the measurement Jacobian, denoted as H', is described by (31).

$$H'(\boldsymbol{x}') = \begin{bmatrix} \frac{\partial \boldsymbol{V}}{\partial \boldsymbol{V}} & \boldsymbol{0} \\ \frac{\partial \boldsymbol{P_i}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{P_i}}{\partial \boldsymbol{\delta}} \\ \frac{\partial \boldsymbol{Q_i}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{Q_i}}{\partial \boldsymbol{\delta}} \\ \frac{\partial \boldsymbol{P_{ij}}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{P_{ij}}}{\partial \boldsymbol{\delta}} \\ \frac{\partial \boldsymbol{Q_{ij}}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{Q_{ij}}}{\partial \boldsymbol{\delta}} \end{bmatrix}. \quad (31)$$

The elements corresponding to the real power injection measurements are summarized in (32)-(35).

$$\frac{\partial P_i}{\partial V_i} = \sum_{j\in N_i} V_j\big(G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij}\big) + V_iG_{ii}. \quad (32)$$

$$\frac{\partial P_i}{\partial V_j} = V_i\big(G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij}\big). \quad (33)$$

$$\frac{\partial P_i}{\partial \delta_{ij}} = V_iV_j\big(-G_{ij}\sin\delta_{ij} + B_{ij}cos\delta_{ij}\big). \quad (34)$$

$$\frac{\partial P_i}{\partial \delta_{ji}} = V_iV_j\big(G_{ij}\sin(-\delta_{ji}) - B_{ij}\cos(-\delta_{ji})\big). \quad (35)$$

The elements corresponding to reactive power injection measurements are summarized in (36)-(39).

$$\frac{\partial Q_i}{\partial V_i} = \sum_{j\in N_i} V_j\big(G_{ij}\sin\delta_{ij} - B_{ij}cos\delta_{ij}\big) - V_iB_{ii}. \quad (36)$$

$$\frac{\partial Q_i}{\partial V_j} = V_i\big(G_{ij}\sin\delta_{ij} - B_{ij}\cos\delta_{ij}\big). \quad (37)$$

$$\frac{\partial Q_i}{\partial \delta_{ij}} = V_iV_j\big(G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij}\big). \quad (38)$$

$$\frac{\partial Q_i}{\partial \delta_{ji}} = -V_iV_j\big(G_{ij}\cos(-\delta_{ij}) + B_{ij}\sin(-\delta_{ij})\big). \quad (39)$$

The elements corresponding to real power flow measurements are summarized in (40)-(43).

$$\frac{\partial P_{ij}}{\partial V_i} = -V_j\big(g_{ij}\cos\delta_{ij} + b_{ij}sin\delta_{ij}\big) + 2\big(g_{ij}+g_{si}\big)V_i. \quad (40)$$

$$\frac{\partial P_{ij}}{\partial V_j} = -V_i\big(g_{ij}\cos\delta_{ij} + b_{ij}sin\delta_{ij}\big). \quad (41)$$

$$\frac{\partial P_{ij}}{\partial \delta_{ij}} = V_iV_j\big(g_{ij}\sin\delta_{ij} - b_{ij}cos\delta_{ij}\big). \quad (42)$$

$$\frac{\partial P_{ij}}{\partial \delta_{ji}} = -V_iV_j\big(g_{ij}\sin(-\delta_{ij}) - b_{ij}\cos(-\delta_{ij})\big). \quad (43)$$

The elements corresponding to reactive power flow measurements are summarized in (44)-(47).

$$\frac{\partial Q_{ij}}{\partial V_i} = -V_j\big(g_{ij}\sin\delta_{ij} - b_{ij}cos\delta_{ij}\big) - 2\big(b_{ij}+b_{si}\big)V_i. \quad (44)$$

$$\frac{\partial Q_{ij}}{\partial V_j} = -V_i\big(g_{ij}\sin\delta_{ij} - b_{ij}cos\delta_{ij}\big). \quad (45)$$

$$\frac{\partial Q_{ij}}{\partial \delta_{ij}} = -V_iV_j\big(g_{ij}\cos\delta_{ij} + b_{ij}sin\delta_{ij}\big). \quad (46)$$

$$\frac{\partial Q_{ij}}{\partial \delta_{ji}} = V_iV_j\big(g_{ij}\cos(-\delta_{ij}) + b_{ij}sin(-\delta_{ij})\big). \quad (47)$$

The elements corresponding to voltage magnitude measurements are summarized in (48).

$$\frac{\partial V_i}{\partial V_i} = 1. \quad (48)$$

APPENDIX B. EQUATIONS FOR IMPLEMENTING THE MFBP METHOD

A. For Real Power Flow Measurements

Forward propagation:

$$\alpha_1^{pf} = \alpha_1^{pf} \cap \Big(\mathrm{sub}\big(g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}, \delta_{ij}\big)\Big). \quad (49)$$

$$\alpha_2^{pf} = \alpha_2^{pf} \cap \big(V_j\alpha_1^{pf}\big). \quad (50)$$

$$P_{ij} = P_{ij} \cap \Big(\mathrm{sub}\Big((g_{si}+g_{ii})V_i^2 - V_i\alpha_2^{pf}, V_i\Big)\Big). \quad (51)$$

Backward propagation:

$$V_i = \mathrm{sub\_tst}(P_{ij} = (g_{si}+g_{ii})V_i^2 - V_i\alpha_2^{pf}, V_i). \quad (52)$$

$$\alpha_2^{pf} = \alpha_2^{pf} \cap \mathrm{sub}(-(P_{ij} - (g_{si}+g_{ii})V_i^2)/V_i, V_i). \quad (53)$$

$$\alpha_1^{pf} = \alpha_1^{pf} \cap (\alpha_2^{pf}/V_j). \quad (54)$$

$$V_j = V_j \cap (\alpha_2^{pf}/\alpha_1^{pf}). \quad (55)$$

$$\delta_{ij} = \mathrm{sub\_tst}(\alpha_1^{pf} = g_{ij}\cos\delta_{ij} + b_{ij}\sin\delta_{ij}, \delta_{ij}). \quad (56)$$

B. For Reactive Power Flow Measurements

Forward propagation:

$$\alpha_1^{qf} = \alpha_1^{qf} \cap \Big(\mathrm{sub}\big(g_{ij}sin\delta_{ij} - b_{ij}cos\delta_{ij}\big)\Big). \quad (57)$$

$$\alpha_2^{qf} = \alpha_2^{qf} \cap \big(V_j\alpha_1^{qf}\big). \quad (58)$$

$$Q_{ij} = Q_{ij} \cap \Big(\mathrm{sub}\big(-\big(b_{si}+b_{ij}\big)V_i^2 - V_i\alpha_2^{qf}, V_i\big)\Big). \quad (59)$$

Backward propagation:

$$V_i = \mathrm{sub\_tst}(Q_{ij} = -\big(b_{si}+b_{ij}\big)V_i^2 - V_i\alpha_2^{qf}, V_i). \quad (60)$$

$$\alpha_2^{qf} = \alpha_2^{qf} \cap \mathrm{sub}(-(Q_{ij} + \big(b_{si}+b_{ij}\big)V_i^2)/V_i, V_i). \quad (61)$$

$$\alpha_1^{qf} = \alpha_1^{qf} \cap (\alpha_2^{qf}/V_j). \quad (62)$$

$$V_j = V_j \cap (\alpha_2^{qf}/\alpha_1^{qf}). \quad (63)$$

$$\delta_{ij} = \mathrm{sub\_tst}(\alpha_1^{qf} = g_{ij}sin\delta_{ij} - b_{ij}cos\delta_{ij}, \delta_{ij}). \quad (64)$$

C. For Real Power Injection Measurements

Forward propagation:

$$\alpha_{1,j}^{pi} = \alpha_{1,j}^{pi} \cap \Big(\mathrm{sub}\big(G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij}\big)\Big). \quad (65)$$

$$\alpha_{2,j}^{pi} = \alpha_{2,j}^{pi} \cap \big(V_j\alpha_1^{pi}\big). \quad (66)$$

$$P_i = P_i \cap (\mathrm{sub}(G_{ii}V_i^2 + V_i\sum_{j\in N_i}\alpha_{2,j}^{pi}, V_i)). \quad (67)$$

Backward propagation:

$$V_i = \mathrm{sub\_tst}(P_i = G_{ii}V_i^2 + V_i\sum_{j\in N_i}\alpha_{2,j}^{pi}, V_i). \quad (68)$$

$$\alpha_{2,j}^{pi} = \alpha_{2,j}^{pi} \cap \text{sub}((P_i - G_{ii}V_i^2 - V_i \sum_{l\in N_i, l\neq j} \alpha_{2,l}^{pi})/V_i, V_i). \quad (69)$$

$$\alpha_{1,j}^{pi} = \alpha_{1,j}^{pi} \cap (\alpha_{2,j}^{pi}/V_j). \quad (70)$$

$$V_j = V_j \cap (\alpha_2^{pi}/\alpha_1^{pi}). \quad (71)$$

$$\delta_{ij} = \text{sub\_tst}(\alpha_{1,j}^{pi} = G_{ij}\cos\delta_{ij} + B_{ij}sin\delta_{ij}, \delta_{ij}). \quad (72)$$

D. Reactive Power Injection Measurements:

Forward propagation:

$$\alpha_{1,j}^{qi} = \alpha_{1,j}^{qi} \cap \left(\text{sub}\left(G_{ij}\sin\delta_{ij} - B_{ij}cos\delta_{ij}\right)\right). \quad (73)$$

$$\alpha_{2,j}^{qi} = \alpha_{2,j}^{qi} \cap \left(V_j\alpha_1^{qi}\right). \quad (74)$$

$$Q_i = Q_i \cap (\text{sub}(-B_{ii}V_i^2 + V_i \sum_{j\in N_i} \alpha_{2,j}^{qi}, V_i)). \quad (75)$$

Backward propagation:

$$V_i = \text{sub\_tst}(Q_i = -B_{ii}V_i^2 + V_i \sum_{j\in N_i} \alpha_{2,j}^{qi}, V_i). \quad (76)$$

$$\alpha_{2,j}^{qi} = \alpha_{2,j}^{qi} \cap \text{sub}((Q_i + B_{ii}V_i^2 - V_i \sum_{l\in N_i, l\neq j} \alpha_{2,l}^{qi})/V_i, V_i). \quad (77)$$

$$\alpha_{1,j}^{qi} = \alpha_{1,j}^{qi} \cap (\alpha_{2,j}^{qi}/V_j). \quad (78)$$

$$V_j = V_j \cap (\alpha_2^{qi}/\alpha_1^{pi}). \quad (79)$$

$$\delta_{ij} = \text{sub\_tst}(\alpha_{1,j}^{qi} = G_{ij}\sin\delta_{ij} - B_{ij}cos\delta_{ij}, \delta_{ij}). \quad (80)$$

REFERENCES

[1] A. Monticelli, "Electric power system state estimation," in *Proceedings of the IEEE*, vol. 88, no. 2, pp. 262-282, Feb. 2000.

[2] F. C. Schweppe, "Power system static-state estimation, part III: implementation," in *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-89, no. 1, pp. 130-135, Jan. 1970.

[3] L. Mili, V. Phaniraj and P. J. Rousseeuw, "Least median of squares estimation in power systems," in *IEEE Transactions on Power Systems*, vol. 6, no. 2, pp. 511-523, May 1991.

[4] L. Mili, M. G. Cheniae and P. J. Rousseeuw, "Robust state estimation of electric power systems," in *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 41, no. 5, pp. 349-358, May 1994.

[5] S. Park, R. Mohammadi-Ghazi and J. Lavaei, "Nonlinear least absolute value estimator for topology error detection and robust state estimation," in *IEEE Access*, vol. 9, pp. 137198-137210, 2021.

[6] J. Zhao, G. Zhang, M. L. Scala and Z. Wang, "Enhanced robustness of state estimator to bad data processing through multi-innovation analysis," in *IEEE Transactions on Industrial Informatics*, vol. 13, no. 4, pp. 1610-1619, Aug. 2017.

[7] L. Mili, M. G. Cheniae, N. S. Vichare and P. J. Rousseeuw, "Robust state estimation based on projection statistics [of power systems]," in *IEEE Transactions on Power Systems*, vol. 11, no. 2, pp. 1118-1127, May 1996.

[8] Y. Chen and N. Zhou, "A comparative study on state estimation algorithms for power systems," *2020 52nd North American Power Symposium (NAPS)*, Tempe, AZ, USA, 2021, pp. 1-6.

[9] J. Zhao, G. Zhang, Z. Y. Dong and K. P. Wong, "Forecasting-aided imperfect false data injection attacks against power system nonlinear state estimation," in *IEEE Transactions on Smart Grid*, vol. 7, no. 1, pp. 6-8, Jan. 2016.

[10] X. Huang, Z. Qin, M. Xie, H. Liu and L. Meng, "Defense of massive false data injection attack via sparse attack points considering uncertain topological changes," in *Journal of Modern Power Systems and Clean Energy*, vol. 10, no. 6, pp. 1588-1598, November 2022.

[11] J. F. Dopazo, O. A. Klitin and A. M. Sasson, "Stochastic load flows," in *IEEE Transactions on Power Apparatus and Systems*, vol. 94, no. 2, pp. 299-309, March 1975.

[12] S. Wang, J. Zhao, Z. Huang and R. Diao, "Assessing Gaussian assumption of PMU measurement error using field data," in *IEEE Transactions on Power Delivery*, vol. 33, no. 6, pp. 3233-3236, Dec. 2018.

[13] F. C. Schweppe, "Recursive state estimation: Unknown but bounded errors and system inputs," *Sixth Symposium on Adaptive Processes*, Chicago, IL, USA, 1967, pp. 102-107.

[14] D. Bertsekas and I. Rhodes, "Recursive state estimation for a set-membership description of uncertainty," in *IEEE Transactions on Automatic Control*, vol. 16, no. 2, pp. 117-128, April 1971.

[15] M.A. Brdys and K. Chen, "Joint state and parameter estimation of dynamic water supply system under bounded uncertainty using geometric programming," in *IFAC Proceedings Volumes*, vol. 27, no. 8, pp. 1287-1292, 1994.

[16] International Bureau of Weights and Measures. (1993). *Guide to the Expression of Uncertainty in Measurement* (vol. 94). DIANE Publishing.

[17] J. Qi, G. He, S. Mei and F. Liu, "Power system set membership state estimation," *2012 IEEE Power and Energy Society General Meeting*, San Diego, CA, USA, 2012, pp. 1-7.

[18] X. Zhang, W. Yan, M. Huo and H. Li, "Robust interval state estimation for distribution systems considering pseudo-measurement interval prediction," in *Journal of Modern Power Systems and Clean Energy*, vol. 12, no. 1, pp. 179-188, January 2024.

[19] I. Krasnochtanova, A. Rauh, M. Kletting, H. Aschemann, E. P. Hofer, and K.-M. Schoop, "Interval methods as a simulation tool for the dynamics of biological wastewater treatment processes with parameter uncertainties," in *Applied Mathematical Modelling*, vol. 34, no. 3, pp. 744-762, 2010.

[20] L. Jaulin and E. Walter, "Set inversion via IA for nonlinear bounded-error estimation," *Automatica*, vol. 29, no. 4, pp. 1053-1064, 1993, doi: 10.1016/0005-1098(93)90106-4.

[21] K. E. Martin et al., "IEEE standard for synchrophasors for power systems," in *IEEE Transactions on Power Delivery*, vol. 13, no. 1, pp. 73- 77, Jan. 1998.

[22] L. Jaulin, M. Kieffer, O. Didrit, and E. Walter, "Interval analysis," in *Applied Interval Analysis*, Springer, pp. 11–43, 2001.

[23] E. Davis, "Constraint propagation with interval labels," *Artificial Intelligence*, vol. 32, no. 3, pp. 281–331, 1987.

[24] G. Nassreddine, F. Abdallah and T. Denoux, "State estimation using IA and belief-function theory: application to dynamic vehicle localization," in *IEEE Transactions on Systems, Man, and Cybernetics*, Part B (Cybernetics), vol. 40, no. 5, pp. 1205-1218, Oct. 2010.

[25] D. Ratz and T. Csendes, "On the selection of subdivision directions in interval branch-and-bound methods for global optimization," *Journal of Global Optimization*, vol. 7, no. 2, pp. 183-207, 1995.

[26] A. Abur and A. G. Exposito, *Power System State Estimation: Theory and Implementation*. CRC press, 2004.

[27] R. E. Moore, *Interval Analysis*, vol. 4. Englewood Cliffs: Prentice-Hall, 1966.

[28] R. E. Moore, R. B. Kearfott, and M. J. Cloud, "8.2 the Krawczyk method," Introduction to Interval Analysis, pp.116-121, SIAM, 2009.

[29] Working Group on a Common Format for the Exchange, "Common data format for the exchange of solved load flow data," *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-92, no. 6, pp. 1916-1925, 1973.

[30] S. H. Dolatabadi, M. Ghorbanian, P. Siano and N. D. Hatziargyriou, "An enhanced IEEE 33-bus benchmark test system for distribution system studies," in *IEEE Transactions on Power Systems*, vol. 36, no. 3, pp. 2565-2572, May 2021.

[31] T. Athay, R. Podmore, and S. Virmani, "A practical method for the direct analysis of transient stability," in *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-98, no. 2, pp. 573-584, Mar.-Apr. 1979.

[32] A. A. Anderson, S. Kincic, B. A. Jefferson, B. J. McGary, C. K. Fallon, D. K. Ciesielski, J. E. Wenskovitch, and Y. Chen, *A Real-Time Operations Manual for the IEEE 118 Bus Transmission Model*, 2022. [Online]. Available: https://www.osti.gov/biblio/1922917.

[33] S.M. Rump, "INTLAB - INTerval LABoratory," In *Developments in Reliable Computing*, Kluwer Academic Publishers, Dordrecht, pp. 77-104, 1999.

[34] T. Raissi, D. Efimov, and A. Zolghadri, "Interval state estimation for a class of nonlinear systems," in *IEEE Transactions on Automatic Control*, vol. 57, no. 1, pp. 260-265, Jan. 2012.

[35] M. B. Do Coutto Filho and J. C. Stacchini de Souza, "Forecasting-aided state estimation—Part I: Panorama," in *IEEE Transactions on Power Systems*, vol. 24, no. 4, pp. 1667-1677, Nov. 2009