Overcoming Imperfect Detection Limitations: Deep Learning-Based Calibration Strategy for Rotating Interferometric Arrays

Zhaohang Zhang*, Chunzhe Wang*, Zhen Huang† and Yafeng Zhan†
* Tsinghua University, China
E-mail: zhangzh22@mails.tsinghua.edu.cn, and wangchunzhe0517@outlook.com
† Tsinghua University, China
E-mail: huangzhen@tsinghua.edu.cn (Corresponding author), and zhanyf@tsinghua.edu.cn

***Abstract*—The calibration of rotating interferometers in real-world applications is critically challenged by complex, coupled systematic errors that render traditional model-based methods ineffective. While deep learning methods offer a powerful alternative, their practical application is hindered by the scarcity of labeled training data, the "black-box" nature of most networks, and their vulnerability to measurement noise. This paper presents a comprehensive strategy that addresses these limitations. We first develop a weakly-supervised labeling scheme that generates training data from non-cooperative sources, removing the reliance on dedicated emitters. Then we design a domain-knowledge-infused network to enhance the interpretability by explicitly modeling key physical quantities to learn their nonlinear couplings. Moreover, a novel inverse-solved feature is proposed to ensure robustness against measurement noise, which provides a denoised, temporally coherent input for the network. On field-collected data where model-based algorithms fail to converge, the proposed approach is shown to reduce positioning error by 34.48%, demonstrating its practical value.**

I. Introduction

Systematic errors, arising from factors like receiver mechanical deviations and site position inaccuracies, are a primary impediment to high-precision passive localization. Consequently, effective calibration of these errors is a critical yet challenging task.

Calibration techniques traditionally fall into two categories: active calibration and self-calibration. Active methods depend on cooperative reference sources for model-based optimization [1]–[3], limiting their applicability in non-cooperative environments. Self-calibration methods [4], [5] alleviate this dependency by jointly estimating target states and system errors. However, both approaches are ineffective when confronted with the complex, non-linearly coupled errors in real-world systems.

Machine learning (ML) techniques [6]–[8] have emerged as promising alternatives to calibrate such complex error models. While promising, they introduce a distinct set of challenges. First, they typically assume the availability of abundant, accurately labeled training data, a condition rarely met in non-cooperative scenarios. Second, many existing network architectures act as black boxes that ignore valuable domain knowledge from physical error models. Finally, their performance can degrade significantly in the presence of measurement noise and pulse loss, as they lack specific mechanisms to ensure robustness.

To overcome these limitations, this paper proposes a deep learning-based calibration strategy for Rotating Long Baseline Interferometers (RLBI) [9]. To solve the label scarcity problem, we propose a novel scheme that generates high-quality training labels by spatially matching position estimates with known non-cooperative reference sources. Moving beyond the black-box paradigm, we design a domain-knowledge-infused neural network that explicitly incorporates key physical parameters as inputs, enabling it to learn their complex and nonlinear error couplings. To ensure robustness against measurement noise and pulse loss, we introduce a novel inverse-solved feature that provides a denoised and structurally stable input for the network. Experimental validation on field-collected RLBI data shows that our method achieves a 34.48% reduction in positioning error where traditional approaches fail, demonstrating its effectiveness and practical utility.

The rest of this paper is organized as follows. Section II presents the system model. Section III details the proposed calibration strategy and neural network. Section IV shows numerical examples, and Section V concludes the paper.

II. Problem Formulation

As illustrated in Fig. 1, we consider a mobile sensor equipped with a Rotating Long Baseline Interferometer (RLBI) system, which measures the phase difference of signals arriving at its two antennas from a radiation source. Let the Earth-Centered Earth-Fixed (ECEF) and body-fixed coordinate systems be denoted by subscripts e and b, respectively. The theoretical, error-free phase difference ϕ_t at time t (with the unit being s) is given by [9]

$$\begin{aligned} \phi_t &= \frac{2\pi}{\lambda} \frac{\boldsymbol{b}_{t,b}^{\top} \boldsymbol{x}_{t,b}}{\|\boldsymbol{x}_{t,b}\|_F}, \\ \boldsymbol{b}_{t,b} &= d \cdot \begin{bmatrix} \cos\beta_t & \sin\beta_t & 0 \end{bmatrix}^{\top}, \\ \boldsymbol{x}_{t,b} &= \boldsymbol{u}_b - \boldsymbol{s}_{t,b}, \end{aligned} \tag{1}$$

979-8-3315-7206-8/25/$31.00

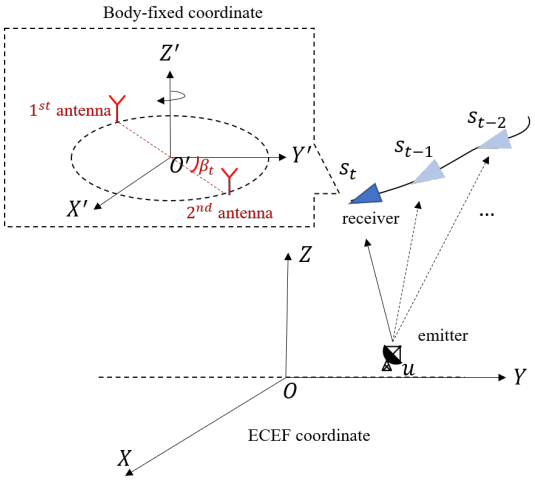


Fig. 1: The RLBI-based positioning system.

which can be reformulated in the ECEF frame as

$$\phi_t = \frac{2\pi}{\lambda} \frac{\boldsymbol{b}_{t,b}^\top \boldsymbol{R}_{t,(b,e)} \boldsymbol{x}_{t,e}}{\|\boldsymbol{R}_{t,(b,e)} \boldsymbol{x}_{t,e}\|_F}, \tag{2}$$
$$\boldsymbol{x}_{t,e} = \boldsymbol{u}_e - \boldsymbol{s}_{t,e},$$

where d represents the RLBI baseline length, β_t is the rotation angle, $\boldsymbol{u}$ and $\boldsymbol{s}$ are the coordinates of the source and sensor, and $\boldsymbol{R}_{t,(b,e)}$ is the rotation matrix from the e-frame to the b-frame, determined by the carrier's attitude angles $[\theta_t, \phi_t, \psi_t]$.

In practical operational scenarios, the measured ambiguous phase difference, $\hat{\phi}_t^n$, is contaminated by a multitude of error sources:

$$\hat{\phi}_t^n = \frac{2\pi}{\lambda'} \frac{\boldsymbol{b}'^{\top}_{t,b} \boldsymbol{R}'_{t,(b,e)} \boldsymbol{x}'_{t,e}}{\|\boldsymbol{R}'_{t,(b,e)} \boldsymbol{x}'_{t,e}\|_F} + n + \Delta_\phi + e - 2\pi N, \tag{3}$$

where λ' incorporates frequency errors, $\boldsymbol{b}'_{t,b}$ contains biases in baseline length and rotation angle, and $\boldsymbol{x}'_{t,e}$ accounts for sensor position errors. The rotation matrix $\boldsymbol{R}'_{t,(b,e)}$ is affected by both attitude measurement inaccuracies and mechanical installation deviations:

$$\boldsymbol{R}'_{t,(b,e)}(\theta, \phi, \psi) = \boldsymbol{R}_\Delta(\Delta\theta, \Delta\phi, \Delta\psi) \boldsymbol{R}_{t,(b,e)}(\theta, \phi, \psi), \tag{4}$$

where $\boldsymbol{R}(\theta, \phi, \psi)$ is the rotation matrix for Euler angles [9], and $\begin{bmatrix}\Delta\theta & \Delta\phi & \Delta\psi\end{bmatrix}$ denotes Euler angle measurement errors. Additionally, the measurement includes random noise n, phase inconsistency Δ_ϕ, an unmodeled error term e (e.g., arising from inter-channel mutual coupling error), and the integer phase ambiguity N.

The measurement model in (3) and (4) presents three fundamental challenges that hinder traditional, model-based calibration methods:

1) **Complex Error Coupling:** The systematic errors are not independent. For instance, the attitude error matrix $\boldsymbol{R}_\Delta$ and the baseline error in $\boldsymbol{b}'_{t,b}$ are non-linearly coupled, a fact often oversimplified or ignored by conventional optimizers that assume error component exclusivity.
2) **Unknown Error Sources:** The specific forms and magnitudes of many errors, particularly the residual unmodeled term e, are often unknown and difficult to model explicitly.
3) **Stochastic Disturbances:** The presence of random noise n and phase ambiguity N makes direct parameter estimation from $\hat{\phi}_t^n$ an ill-posed problem, as they can corrupt the underlying error structure.

These challenges explain why traditional algorithms struggle in real-world scenarios. In our practical field trials, this manifested as a complete failure of model-based methods to converge to a meaningful solution.

Therefore, the central problem addressed in this paper is to learn a mapping function that reconstructs the error-free phase difference ϕ_t (from (2)) from the contaminated, ambiguous measurements $\hat{\phi}_t^n$, along with other available system parameters. This is challenging because the specific structure of the systematic error is obscured by the issues outlined above, motivating our shift from traditional model-based optimization to a deep learning approach, which can learn the complex mapping directly from data without requiring an explicit and hand-crafted error model.

III. Calibration Strategy

This section presents a deep learning-based calibration strategy. It begins by designing the calibration steps and the underlying rationale behind each step. Subsequently, it elaborates on the neural network's fundamental structure and the functions of each component.

A. Deep Learning-based Calibration Algorithm

The proposed calibration strategy consists of two main components: a novel feature engineering method to enhance robustness and a weakly-supervised scheme to generate training labels from non-cooperative sources. The overall procedure is detailed in Algorithm 1.

1) Robust Feature Engineering via Inverse Solving: Raw phase measurements $\hat{\phi}_t^n$ from (3) are unsuitable as direct inputs for a neural network. They are corrupted by stochastic disturbances, such as random noise n and integer ambiguity N, which manifest as non-stationary behavior and abrupt discontinuities in the time series. These artifacts can severely degrade the network's learning convergence and representational fidelity.

To address this, we introduce a robust inverse-solved feature. The core idea is to first obtain a rough position estimate $\boldsymbol{u}^n$ from the noisy measurements, that is

$$\boldsymbol{u}^n = \arg\min_{\boldsymbol{u}} \sum_{t=0}^{T} \left\| \hat{\phi}_t^n - \phi_t(\boldsymbol{u}) + 2\pi N \right\|_F^2 = f(\hat{\boldsymbol{\phi}}^n), \tag{5}$$

where $f(\cdot)$ denotes a standard positioning function (e.g., using Gauss-Newton (GN) or spatial matching methods) that solves an optimization problem with measured phase $\hat{\phi}_t^n$. The spatial matching algorithm is recommended because it can directly match the fuzzy state variables without estimating N.

Then, we use this rough estimate $\boldsymbol{u}^n$ to invert the process. A theoretical, noise-free, and unambiguous phase trajectory $\phi_t(\boldsymbol{u}^n)$ is computed using the ideal model in (2). Though the inverse-solved feature $\phi_t(\boldsymbol{u}^n)$ still implicitly contains the systematic errors because $\boldsymbol{u}^n$ is biased, it is temporally smooth and structurally preserved, making it a much more stable and informative input for the neural network than the raw $\hat{\phi}_t^n$.

2) Weakly-Supervised Labeling with Non-cooperative Sources: A major challenge for deep learning-based calibration is the lack of ground-truth labels in non-cooperative scenarios. We overcome this by proposing a weakly-supervised labeling scheme to build a training dataset. This process is performed offline before training the network.

Assume we have a set of rough position estimates $\boldsymbol{U}^n = \{\boldsymbol{u}_1^n, \ldots, \boldsymbol{u}_M^n\}$ for M observed emitters, and a set of known locations of non-cooperative reference sources (e.g., civilian radars, communication base stations) $\boldsymbol{U}_c = \{\boldsymbol{u}_{c,1}, \ldots, \boldsymbol{u}_{c,N}\}$. The goal is to find the correct association between the estimates in $\boldsymbol{U}^n$ and the true locations in $\boldsymbol{U}_c$.

We model this as a spatial matching problem. To solve it, we first need to model how systematic errors affect the positioning results. The simplified model of systematic error is applied to roughly calculate the bias ratio of different positioning results. Assuming T pulses are involved in localization, the vectorized loss function for position estimation is formulated as

$$\boldsymbol{u}^n = \arg\min_{\boldsymbol{u}} \left\| \Phi^n(\boldsymbol{u}^0) - \Phi(\boldsymbol{u}) \right\|_F^2, \tag{6}$$

where $\Phi^n = \begin{bmatrix} \phi_1^n & \phi_2^n & \cdots & \phi_T^n \end{bmatrix}^\top$, and $\Phi = \begin{bmatrix} \phi_1 & \phi_2 & \cdots & \phi_T \end{bmatrix}^\top$. Given that $\boldsymbol{\Delta u}$ is a small perturbation relative to $\boldsymbol{u}$, a Taylor expansion of Φ^n around $\boldsymbol{u}$ yields

$$\Phi^n(\boldsymbol{u}^0) = \Phi^n(\boldsymbol{u}) + \nabla_{\boldsymbol{u}} \Phi^n(\boldsymbol{u}) \cdot \boldsymbol{\Delta u}, \tag{7}$$

where $\nabla_{\boldsymbol{x}} \boldsymbol{y} = \frac{\partial \boldsymbol{y}}{\partial \boldsymbol{x}}$ denotes the Jacobian operator. Substituting (7) into (6), the closed-form solution for $\boldsymbol{\Delta u}$ is derived as

$$\boldsymbol{\Delta u} = (\nabla_{\boldsymbol{u}} \Phi^n(\boldsymbol{u}^n))^\dagger \cdot (\Phi(\boldsymbol{u}^n) - \Phi^n(\boldsymbol{u}^n)), \tag{8}$$

where $(\cdot)^\dagger$ represents the pseudo-inverse, and the deviation of $\nabla_{\boldsymbol{u}} \Phi^n(\boldsymbol{u}^n)$ is given in appendix A.

Let $\boldsymbol{\Theta}$ denote the vector of predominant systematic errors under consideration. For a given $\boldsymbol{\Theta}$, we can compute the corrected positions $\boldsymbol{U}^0 = \boldsymbol{U}^n - \boldsymbol{\Delta u}(\boldsymbol{\Theta}, \boldsymbol{U}^n)$. Then the subset of reference sources $\boldsymbol{\mathcal{U}} \subseteq \boldsymbol{U}_c$ is found by maximizing the number of successful matches within a tolerance radius σ:

$$\begin{aligned} &\hat{\boldsymbol{\mathcal{U}}} = \arg\max_{\boldsymbol{\Theta}, \boldsymbol{\mathcal{U}} \subseteq \boldsymbol{U}_c} |\boldsymbol{\mathcal{U}}|, \\ &\text{s.t.} \quad \forall \boldsymbol{u}_{c,i} \in \boldsymbol{\mathcal{U}}, \exists \boldsymbol{u}_j^0 \in \boldsymbol{U}^0 \text{ with } \|\boldsymbol{u}_{c,i} - \boldsymbol{u}_j^0\|_F \leq \sigma. \end{aligned} \tag{9}$$

Once the optimal association $(\boldsymbol{u}_j^n \leftrightarrow \boldsymbol{u}_{c,i})$ is established, we can generate the ground-truth phase label $\phi_t(\boldsymbol{u}_{c,i})$ for the corresponding input features using (2).

3) Calibration Procedure: The complete calibration procedure, comprising an offline training phase and an online inference phase, is summarized in Algorithm 1.

Algorithm 1 Deep Learning-based Calibration Algorithm

Input: Sensor data for M emitters: $\{\hat{\boldsymbol{\phi}}_j^n\}_{j=1}^M$; Associated parameters: $\{\boldsymbol{\beta}_j, d_j, \boldsymbol{R}_j, \boldsymbol{s}_j, f_j\}_{j=1}^M$; Set of known non-cooperative source locations $\boldsymbol{U}_c$.
Output: Refined position set $\boldsymbol{U}^0$.

Offline Training Phase:
For each emitter $j = 1, \ldots, M$, compute rough position $\boldsymbol{u}_j^n$ using (5).
Solve the matching problem (9) to establish associations between rough estimates $\{\boldsymbol{u}_j^n\}$ and a subset of true locations $\hat{\boldsymbol{\mathcal{U}}} \subseteq \boldsymbol{U}_c$.
for each established pair $(\boldsymbol{u}_j^n \leftrightarrow \boldsymbol{u}_{c,k})$ **do**
  Compute inverse-solved feature: $\boldsymbol{\phi}_j^{\text{inv}} \leftarrow \phi(\boldsymbol{u}_j^n)$.
  Generate ground-truth label: $\boldsymbol{\phi}_j^{\text{label}} \leftarrow \phi(\boldsymbol{u}_{c,k})$.
  Assemble feature vector: $\mathcal{X}_j \leftarrow [\hat{\boldsymbol{\phi}}_j^n, \boldsymbol{\beta}_j, \ldots, \boldsymbol{\phi}_j^{\text{inv}}]$.
  Store training pair $(\mathcal{X}_j, \boldsymbol{\phi}_j^{\text{label}})$.
end for
Train the calibration network net$(\cdot)$ on all generated training pairs.

Online Inference Phase:
Initialize refined position set $\boldsymbol{U}^0 = \emptyset$.
for each new emitter $j = 1, \ldots, M$ **do**
  Compute rough position $\boldsymbol{u}_j^n$ and inverse-solved feature $\boldsymbol{\phi}_j^{\text{inv}}$.
  Assemble feature vector $\mathcal{X}_j$.
  Predict calibrated phase: $\boldsymbol{\phi}_j^{\text{cal}} \leftarrow \text{net}(\mathcal{X}_j)$.
  Compute refined position $\boldsymbol{u}_j^0$ by solving (6) using $\boldsymbol{\phi}_j^{\text{cal}}$.
  $\boldsymbol{U}^0 \leftarrow \boldsymbol{U}^0 \cup \{\boldsymbol{u}_j^0\}$.
end for

B. Calibration Neural Network

The architecture of the proposed calibration network is founded on a hybrid modeling idea, designed to harness the strengths of both physics-based models and data-driven deep learning. Specifically, the network is architecturally partitioned to tackle distinct error components: a Convolutional Autoencoder (CAE) branch is dedicated to mitigating stochastic, unstructured noise from raw observations, while a main pipeline, consisting of a Gate Recurrent Unit (GRU) and a Transformer, is engineered to decipher the complex, structured systematic errors embedded in a set of physically informed features. This dual-pathway design, illustrated in Fig. 2, ensures both robustness and high-fidelity modeling of intricate error couplings.

1) Robustness Enhancement via Convolutional Autoencoder: To ensure stability under high noise conditions where the inverse-solved feature may become unreliable, we incorporate a CAE branch that operates directly on the raw phase measurements $\hat{\boldsymbol{\phi}}^n$. This unit serves a dual role. Firstly, it acts as a robustness safety net. The denoised representation from the CAE provides a stable, fallback source of information for the network when high levels of random noise render the inverse-solved feature unstable. Secondly, it functions as an information complement. The inverse-solving process, while

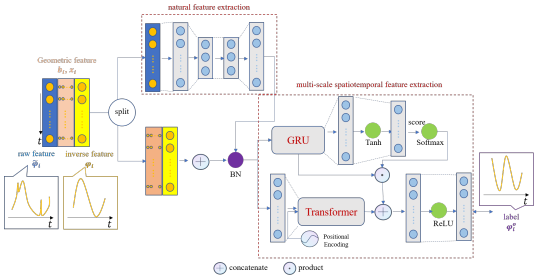


Fig. 2: The architecture of the proposed calibration neural network.

removing noise, might also discard subtle patterns within the raw observations. The CAE captures these residual patterns, providing a richer, multi-faceted input to the subsequent layers and preventing information loss.

The CAE is jointly trained with the entire network. Its output, the denoised feature $\hat{\boldsymbol{x}}_{\text{CAE}}$, is passed to the subsequent modules. The forward propagation is a standard encode-decode process:

$$\boldsymbol{h}_{\text{latent}} = \text{Encoder}(\hat{\boldsymbol{\phi}}^n; \boldsymbol{W}_{\text{enc}}), \tag{10}$$

$$\hat{\boldsymbol{x}}_{\text{CAE}} = \text{Decoder}(\boldsymbol{h}_{\text{latent}}; \boldsymbol{W}_{\text{dec}}), \tag{11}$$

where the encoder typically consists of convolutional and pooling layers, and the decoder uses corresponding transposed operations. Unlike pre-trained autoencoders, the CAE in this paper provides auxiliary information throughout the joint training process.

2) Multi-Scale Temporal Dependency Extraction: The core of our network is designed to learn the complex, time-varying relationships between different physical quantities. This is achieved by a dual-component module that captures both local and global temporal dependencies.

a) Physically-Informed Input Features: The proposed network is explicitly fed with a set of features that carry physical meaning:

$$\boldsymbol{x}_{\text{in}} = \text{concat}\left(\hat{\boldsymbol{x}}_{\text{CAE}}, \boldsymbol{\phi}^{\text{inv}}, \boldsymbol{b}(d, \boldsymbol{\beta}), \boldsymbol{rx}(\boldsymbol{R}, \boldsymbol{s}, \boldsymbol{u}^n), f\right), \tag{12}$$

where $\hat{\boldsymbol{x}}_{\text{CAE}}$ is the denoised feature from the CAE, $\boldsymbol{\phi}^{\text{inv}}$ is the robust inverse-solved feature, and $\boldsymbol{b}$ and $\boldsymbol{rx}$ are vectors representing the baseline and relative sensor-emitter positions, respectively. Feeding these components as distinct features allows the network to learn their nonlinear interactions, which are difficult to model analytically.

The choice of the baseline vector $\boldsymbol{b}$ and the relative position vector $\boldsymbol{rx}$ is directly motivated by the interferometric phase equation (2). These features constitute the fundamental geometric basis of the phase measurement. By feeding them as distinct inputs, we impose a soft physical constraint on the network, guiding it to learn the mapping within a physically plausible solution space, rather than searching blindly in a high-dimensional space. This significantly improves sample efficiency and generalization.

b) Multi-Scale Temporal Fusion with GRU and Transformer: A multi-scale fusion architecture that combines a GRU and a Transformer is proposed to extract the temporal dynamics of the systematic errors. This design is analogous to performing a spectral analysis of the error signal over time. The GRU processes the sequence chronologically, making it exceptionally effective at modeling causal relationships and high-frequency, local fluctuations in the error, such as those caused by rapid platform movements. In contrast, the Transformer's self-attention mechanism operates on the entire sequence at once. It excels at capturing long-range dependencies and global context, such as a slowly drifting bias or a system-wide scaling error that persists throughout the observation period. By fusing the outputs from these two complementary modules, the proposed network can construct a comprehensive representation of the systematic errors, capturing both their local transient behaviors and their global stationary properties.

Firstly, a GRU network is used to model local, sequential patterns in the feature time series. The GRU's recurrent nature is well-suited for capturing the forward-propagating dependencies inherent in the sensor's movement. An attention mechanism is applied to the GRU's hidden states $\boldsymbol{h}_{\text{GRU}}$ to weigh the importance of different time steps, producing an output $\boldsymbol{x}_{\text{GRU-Attn}}$:

$$\boldsymbol{h}_{\text{GRU}} = \text{GRU}(\boldsymbol{x}_{\text{in}}; \boldsymbol{W}_{\text{GRU}}), \tag{13}$$

$$\boldsymbol{\alpha} = \text{Softmax}(\text{AttentionScore}(\boldsymbol{h}_{\text{GRU}})), \tag{14}$$

$$\boldsymbol{x}_{\text{GRU-Attn}} = \boldsymbol{h}_{\text{GRU}} + \boldsymbol{\alpha} \odot \boldsymbol{h}_{\text{GRU}}, \tag{15}$$

where the residual connection enhances robustness.

While GRUs excel at local patterns, they struggle with long-range dependencies. Consequently, a Transformer encoder is employed in parallel, which uses a self-attention mechanism to model global context across the entire time series. The input $\boldsymbol{x}_{\text{in}}$ is first passed through an embedding layer with positional encoding (PE) to create query (Q), key (K), and value (V) matrices. The output is computed as:

$$\boldsymbol{X}_e = \text{Embedding}(\boldsymbol{x}_{\text{in}}) + \boldsymbol{PE}, \tag{16}$$

$$\boldsymbol{H}_{\text{Attn}} = \text{MultiHead}(\boldsymbol{Q} = \boldsymbol{X}_e, \boldsymbol{K} = \boldsymbol{X}_e, \boldsymbol{V} = \boldsymbol{X}_e), \tag{17}$$

$$\boldsymbol{x}_{\text{Trans}} = \text{LayerNorm}(\boldsymbol{X}_e + \boldsymbol{H}_{\text{Attn}}). \tag{18}$$

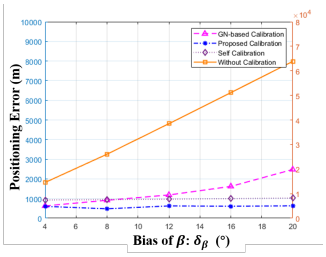
The final network output is a fusion of the features extracted by both the GRU and Transformer branches, allowing the model to leverage both local and global temporal information for a highly accurate calibration.

IV. Numerical Experiments

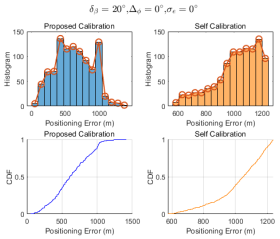
This section evaluates the performance of the proposed calibration framework through extensive simulations and real-world data.

A. Simulation Setup

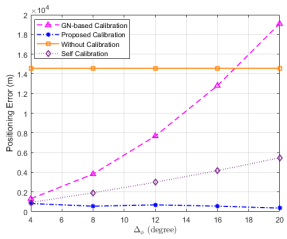
We simulate a moving RLBI system (1 m baseline, 0.5π rad/s rotation rate) and randomly generate emitter positions over 1,000 Monte Carlo trials. The system errors in RLBI are modeled, including a baseline length error of 0.01 m and



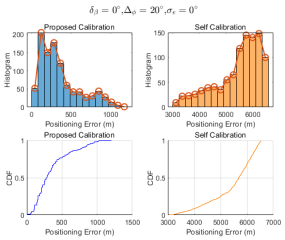
(a) Impact of rotational angle bias δ_β on average error.



(b) Error distribution at $\delta_\beta = 20°$.



(c) Impact of phase bias Δ_ϕ on average error.



(d) Error distribution at $\Delta_\phi = 20°$.

Fig. 3: Performance comparison under unmodeled systematic biases in rotation angle and phase.

attitude measurement errors of $0.15°$, $0.35°$, and $0.2°$ for roll, pitch, and yaw, respectively.

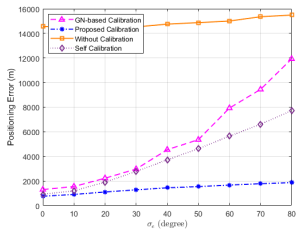
The proposed network consists of a CAE (3 conv-layers, kernels: 8,4,2), a bi-GRU (hidden dim: 64), and a 6-layer Transformer (8 heads, d_{model}: 64). It is trained for 500 epochs using the Adam optimizer (LR: 1×10^{-3}, batch size: 8).

B. Performance Comparison with Traditional Algorithms

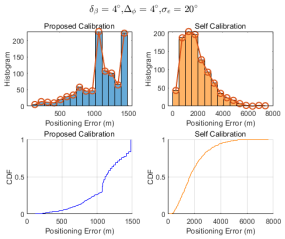
The performance of traditional calibration algorithms degrades significantly under model-mismatch conditions. Two comparative algorithms, including an active GN-based calibration method [3] and a self-calibration method [4], are chosen to validate the robustness of the proposed method.

1) Robustness to Systematic Error Model Mismatch: We first test the resilience to unmodeled systematic biases. Figs. 3(a) and 3(c) show that as biases in rotation angle (δ_β) and phase (Δ_ϕ) increase, the performance of traditional GN and self-calibration methods degrades sharply, highlighting their dependency on accurate error models. In contrast, the proposed approach achieves a 74.6% improvement over the GN method at $\delta_\beta = 20°$, showing that the proposed method learns the complex error mapping without relying on a predefined model. The histograms in Fig. 3(b) and 3(d) further confirm this, showing that the proposed method's error distribution remains compact and centered at a much lower value.

2) Robustness to Random Measurement Noise: Next, we evaluate the algorithms' resilience to the stochastic disturbance, characterized by its standard deviation σ_e. Fig. 4(a) shows that the proposed method exhibits the least performance degradation. This notable resilience is a benefit of the dual-pathway network architecture. The histogram in Fig. 4(b) at



(a) Impact of random noise σ_e on average error.



(b) Error distribution at $\sigma_e = 20°$.

Fig. 4: Performance comparison under varying levels of random measurement noise.

$\sigma_e = 20°$ illustrates that the positioning errors are still tightly clustered below 1.5 km, whereas the self-calibration errors are widely spread.

C. Performance Comparison with Deep Learning Benchmarks

Table I shows that while all DL methods perform well in low noise, the proposed approach is significantly more robust, yielding more than 67% error reduction over the AE [6], BERT [7], and DECL [8] baselines at $\sigma_e = 80°$. The proposed architecture's superior noise robustness over other networks stems from an informational fusion of CAE-denoised and inverse-solved features and the inherent robustness of its multi-scale feature space.

D. Validation on Real-World Data

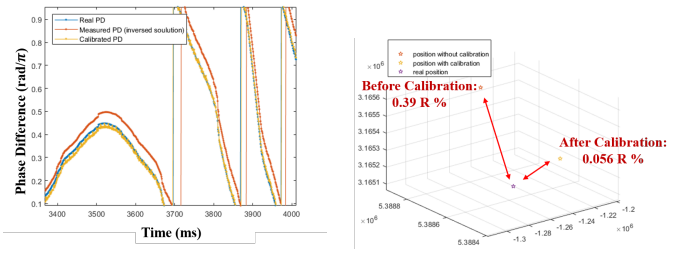
The practical viability of the proposed method is validated on a challenging real-world dataset, where traditional algorithms fail to converge. This dataset features a complex mixture of errors that are difficult to model analytically. As illustrated in a representative case in Fig. 5, our method successfully corrects the measured phase difference to track the ground truth. This dramatically reduces the positioning error, from 0.39R% to a mere 0.056R% (where R is the target range). Across the entire test set, the proposed algorithm reduces the mean positioning error from 0.40R% to 0.26R%, achieving a 34.48% improvement in localization accuracy, confirming its real-world effectiveness and practical value.

V. Conclusion

This paper presents a deep learning-based calibration framework for RLBI systems, specifically to overcome the limitations of traditional models in complex, non-cooperative

TABLE I: Positioning error (m) comparison with DL benchmarks under a fixed systematic bias ($\delta_\beta = 4°, \Delta_\phi = 4°$).

σ_e (°)	**Proposed**	**AE-based**	**BERT**	**DECL**
0	**762.23**	1578.50	1076.51	2303.30
20	**1108.55**	2057.31	1756.39	2592.88
40	**1447.46**	3162.82	3047.62	3595.14
60	**1669.76**	4475.93	4340.85	4664.12
80	**1868.12**	5980.10	5741.27	5848.58



(a) Phase difference calibration. (b) 3D positioning result.

Fig. 5: Performance validation on a real-world data sample. (a) The algorithm corrects the measured phase to align with the real phase. (b) The calibration significantly reduces the 3D positioning error.

scenarios. The method is built upon three primary contributions: a weakly-supervised paradigm that generates training labels from non-cooperative sources, a domain-knowledge-infused network for learning intricate error couplings, and a robust dual-path feature mechanism that ensures stability against measurement noise. Validation on both simulated and real-world datasets confirms the framework's effectiveness. Notably, in challenging field data where traditional algorithms fail to converge, the proposed method reduces positioning error by over 34.48%. Future work will focus on exploring more lightweight network architectures for on-platform deployment.

APPENDIX A
DEVIATION OF JACOBIAN MATRIX IN (8)

Now define

$$\tilde{\phi}_t^n = \frac{2\pi}{\lambda'} \frac{{\boldsymbol{b}'_{t,b}}^\top \boldsymbol{R}'_{t,(b,e)} \boldsymbol{x}'_{t,e}}{\|\boldsymbol{R}'_{t,(b,e)} \boldsymbol{x}'_{t,e}\|_F}. \tag{19}$$

Comparing (19) with (3), the gradient relationship satisfies $\nabla_{\boldsymbol{u}} \phi^n(\boldsymbol{u}) = \nabla_{\boldsymbol{u}} \tilde{\phi}^n(\boldsymbol{u})$. Then its matrix form is given by

$$\nabla_{\boldsymbol{u}} \tilde{\Phi}^n(\boldsymbol{u}) = \nabla_{\boldsymbol{u}} \Phi^n(\boldsymbol{u}), \tag{20}$$

where $\tilde{\Phi}^n = \begin{bmatrix} \tilde{\phi}_1^n & \tilde{\phi}_2^n & \cdots & \tilde{\phi}_T^n \end{bmatrix}^\top$.

To clarify the systemic error in (19), we define

$$\begin{aligned} \lambda' = \delta_\lambda \lambda,\ \boldsymbol{b}^\perp_{t,b} = d \begin{bmatrix} -\sin\beta_t & \cos\beta_t & 0 \end{bmatrix}^\top, \\ \boldsymbol{b}'_{t,b} = \delta_d d \begin{bmatrix} \cos(\beta_t + \delta_\beta) & \sin(\beta_t + \delta_\beta) & 0 \end{bmatrix}^\top. \end{aligned} \tag{21}$$

To simplify the derivation process, we also make extra definitions as follows:

$$\boldsymbol{E} = \delta_\lambda \delta_d \cos\delta_\beta \boldsymbol{R}_\Delta,\ \boldsymbol{E}_\epsilon = \begin{bmatrix} 1 \\ \tan\delta_\beta \end{bmatrix} \otimes \boldsymbol{E}, \tag{22}$$

where $\otimes$ denotes the Kronecker product. Then the simplified expression of $\tilde{\phi}_t^n$ and its Jacobian matrix towards $\boldsymbol{u}$ is given by

$$\begin{aligned} \tilde{\phi}_t^n &= \tilde{\boldsymbol{b}}_t^\top \boldsymbol{E}_\epsilon \tilde{\boldsymbol{x}}_t, \\ \nabla_{\boldsymbol{u}} \tilde{\phi}_t^n &= \tilde{\boldsymbol{b}}_t^\top \boldsymbol{E}_\epsilon \nabla_{\boldsymbol{u}} (\tilde{\boldsymbol{x}}_t) \\ &= \tilde{\boldsymbol{b}}_t^\top \boldsymbol{E}_\epsilon \boldsymbol{R}_{t,(b,e)} \frac{\boldsymbol{I}_3 \|\boldsymbol{x}_{t,e}\|_F^2 - \boldsymbol{x}_{t,e} \boldsymbol{x}_{t,e}^\top}{\|\boldsymbol{x}_{t,e}\|_F^3}, \end{aligned} \tag{23}$$

where $\tilde{\boldsymbol{b}}_t = \frac{2\pi d}{\lambda} \begin{bmatrix} \boldsymbol{b}_{t,b} \\ \boldsymbol{b}^\perp_{t,b} \end{bmatrix}$, and $\tilde{\boldsymbol{x}}_t = \boldsymbol{R}_{t,(b,e)} \frac{\boldsymbol{x}_{t,e}}{\|\boldsymbol{x}_{t,e}\|_F}$. Note that sensor location errors are neglected in the derivation for simplicity. Define $\tilde{\boldsymbol{B}} = \text{diag}(\tilde{\boldsymbol{b}}_1^\top, \tilde{\boldsymbol{b}}_2^\top, \cdots, \tilde{\boldsymbol{b}}_T^\top)$, and $\tilde{\boldsymbol{X}} = \begin{bmatrix} \nabla_{\boldsymbol{u}} (\tilde{\boldsymbol{x}}_1) & \nabla_{\boldsymbol{u}} (\tilde{\boldsymbol{x}}_2) & \cdots & \nabla_{\boldsymbol{u}} (\tilde{\boldsymbol{x}}_T) \end{bmatrix}^\top$, we have

$$\nabla_{\boldsymbol{u}} \tilde{\Phi}^n = \tilde{\boldsymbol{B}} (\boldsymbol{I}_T \otimes \boldsymbol{E}_\epsilon) \tilde{\boldsymbol{X}} \tag{24}$$

REFERENCES

[1] M. Piavanini, L. Barbieri, M. Brambilla, *et al.*, "A calibration method for antenna delay estimation and anchor self-localization in uwb systems," in *2022 IEEE International Workshop on Metrology for Industry 4.0 & IoT (MetroInd4. 0&IoT)*, IEEE, 2022, pp. 173–177.

[2] C. Zhao, D. Ding, Y. Shi, Y. Ji, and Y. Du, "Graph matching-based spatiotemporal calibration of roadside sensors in cooperative vehicle-infrastructure systems," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[3] F. Ma, Z.-M. Liu, and F. Guo, "Direct position determination in asynchronous sensor networks," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 9, pp. 8790–8803, 2019. DOI: 10.1109/TVT.2019.2928638.

[4] J. Wang, Y. He, D. Su, *et al.*, "Slam-based joint calibration of multiple asynchronous microphone arrays and sound source localization," *IEEE Transactions on Robotics*, 2024.

[5] P. Yin, D. Zhang, T. Zhang, *et al.*, "Autocali: Enhancing aoa-based indoor localization through automatic phase calibration," in *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2024, pp. 9046–9050.

[6] Z.-M. Liu, C. Zhang, and P. S. Yu, "Direction-of-arrival estimation based on deep neural networks with robustness to array imperfections," *IEEE Transactions on Antennas and Propagation*, vol. 66, no. 12, pp. 7315–7327, 2018. DOI: 10.1109/TAP.2018.2874430.

[7] X. Zhao, J. Atli Benediktsson, Y. Yang, K.-S. Chen, and M. Örn Úlfarsson, "Exploring transformer-based direction-of-arrival estimation over sea surface: A bert approach with physics-based loss function," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–13, 2024. DOI: 10.1109/TGRS.2024.3440224.

[8] S. Zhou, D. Zha, X. Shen, X. Huang, R. Zhang, and K. Chung, "Denoising-aware contrastive learning for noisy time series," in *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, Aug. 2024, pp. 5644–5652. DOI: 10.24963/ijcai.2024/624.

[9] Z.-M. Liu and F.-C. Guo, "Azimuth and elevation estimation with rotating long-baseline interferometers," *IEEE Transactions on Signal Processing*, vol. 63, no. 9, pp. 2405–2419, 2015. DOI: 10.1109/TSP.2015.2405506.