Journal of *Marine Science and Engineering*

MDPI

Article

An Improved Factor Graph Optimization Algorithm Enhanced with ANFIS for Ship GNSS/DR Integrated Navigation

Yi Jiang [1], Heng Gao [1,*], Tianyu Zhang [1], Jin Xiang [1], Yichi Zhang [2], Jingqing Ke [2] and Qing Hu [1]

[1] College of Information Science and Technology, Dalian Maritime University, Dalian 116026, China; j_y@dlmu.edu.cn (Y.J.); zhangtianyu662025@163.com (T.Z.); xiangjin1632021@163.com (J.X.); hq0518@dlmu.edu.cn (Q.H.)

[2] Northern Navigation Service Center, Maritime Safety Administration, Tianjin 300220, China; liyuhanon@126.com (Y.Z.); kejingqing66@163.com (J.K.)

* Correspondence: gh1120200204@dlmu.edu.cn

Abstract

Accurate and reliable positioning is essential for unmanned marine vehicles (UMVs), especially in complex maritime environments. Existing algorithms often underutilize historical information, struggle with nonlinear dynamics, and lack adaptability in the GNSS Measurement Noise Covariance, leading to degraded performance. This study proposes an enhanced Factor Graph Optimization (FGO) method integrated with an adaptive neuro-fuzzy inference system (ANFIS) to overcome these challenges. First, an improved GNSS/Dead Reckoning (DR) factor graph is built using refined error models to enhance baseline accuracy. Second, a marginalization factor is introduced utilizing a sliding window and the Schur complement method to retain informative historical data while reducing computational load, thereby improving stability and field performance. Third, an ANFIS-based adaptive GNSS factor dynamically updates the GNSS Measurement Noise Covariance Matrix (GMNCM) to strengthen robustness under variable maritime conditions. Simulation and field tests demonstrate significant improvements: the proposed method achieves 29.1%, 26.5%, and 9.9% higher accuracy than EKF, UKF, and conventional FGO, respctively. Under GNSS interruptions, EKF and UKF diverge with errors exceeding 500 m, while FGO limits drift to 20 m. The proposed ANFIS–FGO shows the smallest fluctuations and fastest recovery, confirming its strong resilience and practical applicability for UMV navigation.

Keywords: unmanned marine vehicle; ship positioning; GNSS/DR integrated navigation; factor graph optimization; adaptive neuro-fuzzy inference system

Check for updates

Academic Editor: Raphael Zaccone

Received: 23 January 2026
Revised: 22 February 2026
Accepted: 26 February 2026
Published: 28 February 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

With the development of the maritime economy, conventional ships face increasing constraints in efficiency, safety, and environmental performance, falling short of modern maritime demands [1]. In this context, unmanned marine vehicles (UMVs) have emerged as a popular solution to these challenges [2]. Nevertheless, the large-scale deployment of UMVs still faces critical challenges, among which high-precision and reliable positioning technologies serve as a fundamental prerequisite for achieving robust and adaptive autonomous navigation in complex maritime environments [3].

The Global Navigation Satellite System (GNSS) serves as the cornerstone of maritime positioning due to its wide-area coverage and long-term localization capabilities [4]. However, dynamic maritime conditions and environmental disturbances frequently compromise

its performance [5]. Therefore, GNSS is often integrated with autonomous navigation systems such as the Inertial Navigation System (INS). Nevertheless, many ships lack INS equipment. Alternatively, dead reckoning (DR) systems estimate ship positions utilizing shipborne sensor data, offering high short-term accuracy but suffering from cumulative errors that compromise long-term reliability [6]. Accordingly, this study develops a GNSS/DR integrated navigation system to enhance ship positioning stability.

The ship's integrated navigation systems mainly employ the Extended Kalman Filter (EKF) [7], the Unscented Kalman Filter (UKF) [8], and various improved variants [9–11]. Although these methods have promoted maritime navigation technologies, several limitations remain. Specifically, they inadequately exploit global historical data, exhibit limited nonlinearity handling capability, and demonstrate poor scalability. Consequently, ship positioning accuracy and practical implementation are constrained. To overcome these limitations, some researchers have introduced Factor Graph Optimization (FGO), which models the relationships between state vectors and measurements as factors, transforming state estimation into a graph optimization problem. Compared with traditional filtering, FGO methods offer greater flexibility in fusing multi-source data and achieve higher accuracy and robustness in handling nonlinearities [12,13]. Chen et al. proposed an FGO-based pedestrian DR/skyward image/optical fusion navigation method, demonstrating its feasibility in complex environments [14]. Lyu et al. proposed an improved INS pre-integration-based FGO that effectively enhanced vehicle positioning accuracy [15]. Zhang et al. leveraged factor graph flexibility with INS pre-integration to significantly improve autonomous underwater vehicle positioning accuracy [16]. These methods primarily construct factor graphs based on INS, overlooking the practical reality that many ships are not equipped with INS. Therefore, this study develops a novel FGO method based on GNSS/DR integrated navigation systems according to practical ship conditions and positioning requirements, without requiring additional sensors.

However, GNSS measurement errors exhibit substantial temporal variability and uncertainty due to dynamic maritime conditions and complex ship dynamics. Consequently, the fixed GNSS Measurement Noise Covariance Matrix (GMNCM) fails to capture actual error characteristics, limiting FGO performance in complex maritime environments [17]. Solutions to this issue are generally categorized into statistical methods and Artificial Intelligence (AI) methods. For instance, Zhang et al. introduced an adaptive factor weighting function to adjust sensor measurement reliability, improving indoor robot localization [18]. Tian et al. proposed an adaptive fast incremental smoothing FGO algorithm that enhances positioning accuracy under varying sensor reliability [19]. Chen et al. developed a multi-factor optimization graph that dynamically adjusts for environmental noise and sensor errors, improving attitude estimation accuracy and stability [20]. However, they typically rely on predefined error models and assumptions, often failing in complex marine environments and thereby limiting accuracy and robustness. To overcome these limitations, recent studies have increasingly adopted AI approaches. Through data-driven modeling, AI enables adaptive covariance estimation by learning patterns from historical data without requiring precise prior models. Ben et al. combined dynamic kernel PCA with FGO for robust positioning through nonlinear feature extraction, though with high computational complexity and challenging parameter tuning [21]. Xu et al. integrated dynamic trust functions and bidirectional LSTM into FGO, improving positioning accuracy under anomalous measurements but requiring complex architecture and large quantities of high-quality training data [22]. Li et al. employed variational Bayesian networks and Gaussian mixture models for noise estimation, but the method requires prior data partitioning and exhibits significant adaptation delays [23]. Overall, these methods present varying limitations in computational efficiency, data requirements, real-time performance, and generalization ca-

pability, making it challenging to simultaneously address dynamic environments, resource constraints, and scarce high-precision data in maritime ship positioning [24–28].

To address these issues, this paper proposes an innovative FGO algorithm enhanced with ANFIS (ANFIS–FGO). ANFIS employs a lightweight five-layer feedforward network that establishes nonlinear mapping relationships between ship motion characteristics and GNSS measurement noise through fuzzy rules. This approach offers several significant advantages: First, the fuzzy rule architecture embeds expert knowledge-based structured biases, assuming piecewise–smooth relationships between ship motion states and GNSS measurement noise to effectively handle measurement uncertainty in maritime environments. Second, continuous fuzzy membership functions ensure smooth extrapolation beyond training data, avoiding unpredictable behavior of conventional AI methods in unseen scenarios and enabling effective operation under scarce high-precision data and high-noise conditions. Third, ANFIS fuzzy rules provide explicit physical interpretations and causal logic, facilitating decision-making and debugging in autonomous navigation. Finally, ANFIS relies solely on current input features, yielding low computational complexity. In summary, the proposed ANFIS–FGO method overcomes traditional filtering limitations, enhances ship positioning accuracy, and significantly improves system robustness under varying GNSS measurement noise conditions. It achieves effective positioning in resource-constrained environments with scarce high-precision data and high-noise levels, providing reliable technical support for the autonomous navigation of UMVs.

The main contributions of this paper are as follows:

1. We develop a ship GNSS/DR integrated navigation system based on shipborne sensors, improving ship positioning performance without additional hardware.
2. To address the limitations of filtering algorithms in ship localization, including insufficient utilization of global information and limited accuracy in nonlinear dynamic modeling, an improved FGO algorithm is proposed. By incorporating DR error factors, GNSS measurements, and marginalization factors, the algorithm enhances both positioning performance and computational efficiency.
3. To mitigate FGO performance degradation in complex maritime environments, an ANFIS–FGO algorithm is proposed. Leveraging ANFIS's adaptive learning and nonlinear modeling capabilities, the GMNCM is dynamically adjusted, further improving ship positioning accuracy and system robustness.

The remainder of this paper is organized as follows: Section 2 describes the factor graph model and its components; Section 3 details the proposed algorithm, which forms the core of the study; Section 4 presents simulation and experiments along with analytical verification; and Section 5 concludes the research and outlines directions for future work.

2. GNSS/DR Factor Graph Model

This section presents a factor graph-based framework for maritime GNSS/DR integrated navigation. The framework comprises shipborne DR, DR error, GNSS, and marginalization factors, forming a sparse and computationally efficient structure for state estimation. By combining the short-term accuracy of dead reckoning with the absolute positioning capability of GNSS and leveraging the structural advantages of factor graphs for multi-source fusion, the proposed method enhances estimation accuracy, continuity, and adaptability in complex marine environments.

2.1. GNSS/DR Factor Graph Framework

A GNSS/DR factor graph is developed based on the ship GNSS/DR integrated navigation system, as shown in Figure 1. The system state is defined as $\boldsymbol{X} = (\boldsymbol{X}_1, \boldsymbol{X}_2 \cdots \boldsymbol{X}_k)$, representing the navigation state of the ship at each epoch, including state variables such

as position, velocity, and heading. Meanwhile, the accumulated drift inherent to DR is modeled explicitly through an additional error variable $\boldsymbol{C} = (\boldsymbol{C}_1, \boldsymbol{C}_2 \cdots \boldsymbol{C}_{k-1})$, which is jointly estimated with the ship state vector to improve drift compensation.

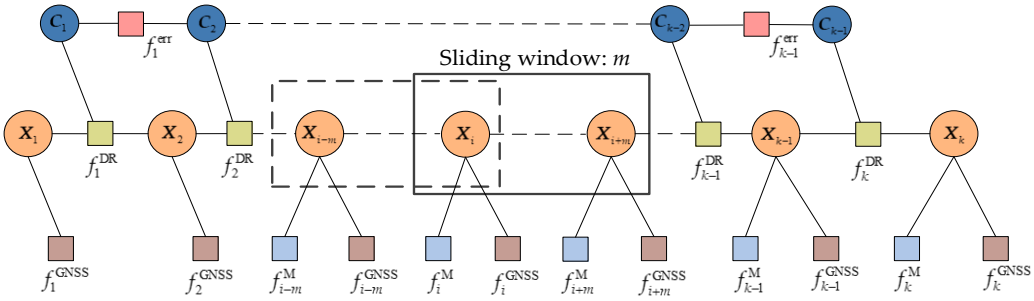


Figure 1. GNSS/DR factor graph framework.

The DR factor f^{DR} encodes the relative motion dynamics between consecutive states $\boldsymbol{X}_{i-1}$ and $\boldsymbol{X}_i$ based on the DR propagation model. By incorporating the corresponding drift term $\boldsymbol{C}_{i-1}$, this factor explicitly accounts for the long-term drift characteristics of DR and constrains the state transition accordingly. The DR error factor f^{err} further refines the estimation of $\boldsymbol{C}_i$ by constraining the temporal evolution of the DR error states through a stochastic error model driven by sensor characteristics, enabling more accurate modeling and correction of the accumulated DR drift.

The GNSS factor f^{GNSS} provides absolute position constraints derived from GNSS measurements, enabling the system to mitigate accumulated dead reckoning errors whenever GNSS observations are available.

Meanwhile, to ensure computational efficiency, a sliding window strategy is adopted. Only the most recent m state nodes are retained in the optimization. Older states and their associated factors are marginalized out, thereby reducing computational load while preserving the information necessary for accurate state estimation.

2.2. DR Factor

In the East–North (EN) local coordinate system, the DR motion model is:

$$\begin{cases} x_k = x_{k-1} + v_k^{\mathrm{T}} \sin\left(\theta_k^{\mathrm{T}}\right) \Delta t \\ y_k = y_{k-1} + v_k^{\mathrm{T}} \cos\left(\theta_k^{\mathrm{T}}\right) \Delta t \end{cases} \tag{1}$$

where (x_k, y_k) and (x_{k-1}, y_{k-1}) represent the ship's position at epoch k and $k-1$. v_k^{T} denotes the ship's speed at epoch k, Δt is the sampling interval, which is 1s in the study. θ_k^{T} represents the heading angle at epoch k.

Based on Equation (1), the ship state vector at epoch k can be described as follows:

$$\boldsymbol{X}_k = \left(\boldsymbol{p}_k, \boldsymbol{V}_k, \theta_k^{\mathrm{T}}\right) \tag{2}$$

where $\boldsymbol{X}_k \in \mathbb{R}^5$. $\boldsymbol{p}_k = (x_k, y_k)$ and $\boldsymbol{V}_k = \left(v_k^{\mathrm{T}} \sin\left(\theta_k^{\mathrm{T}}\right), v_k^{\mathrm{T}} \cos\left(\theta_k^{\mathrm{T}}\right)\right)$ represent subsets of the ship's position and speed, obtained through DR.

The DR error vector at epoch k can be formulated as follows:

$$\boldsymbol{C}_k = (\nabla_k, \varepsilon_k) \tag{3}$$

where ∇_k and ε_k represent the bias error of the log and the compass, respectively.

The DR error factor f^{err} at epoch k can be expressed as follows:

$$f_k^{\mathrm{err}}(\boldsymbol{C}_k, \boldsymbol{C}_{k-1}) = \exp\left(-\frac{1}{2}\|\boldsymbol{C}_k - f_k^{\mathrm{c}}(\boldsymbol{C}_{k-1})\|_{\zeta_k}^2\right) \tag{4}$$

where ζ_k is the covariance matrix corresponding to $\boldsymbol{C}_k$. f_k^{c} represents the state transition of the DR error vector. Based on the error characteristics of the log and compass, f_k^{c} is typically modeled as a Gaussian random walk process [29]:

$$f_k^{\mathrm{c}}(\boldsymbol{C}_k) = \begin{cases} \nabla_{k-1} + \nu_k + w_\nabla \\ \varepsilon_{k-1} + \omega_k + w_\varepsilon \end{cases} \tag{5}$$

where ν_k and ω_k represent the bias error increments in the log and compass, respectively. They are typically modeled as zero-mean Gaussian white noise processes. w_∇ and w_ε denote the measurement noise of the log and compass respectively, which are also assumed to be zero-mean Gaussian white noise processes.

In the factor graph, f^{err} connects the DR error vector nodes $\boldsymbol{C}$ at adjacent time steps, defining their joint probabilistic relationship. Through the sum-product algorithm, f^{err} enables message passing between nodes, allowing iterative updates of their marginal distributions [30]. Within the factor graph structure, these updates to the $\boldsymbol{C}$ ultimately enable more accurate error compensation for the ship state vector $\boldsymbol{X}$.

The measurements from the log and compass at epoch k are used to obtain $z_k^{\mathrm{DR}} = (v_k^{\mathrm{S}}, \theta_k^{\mathrm{H}})$, which then contributes to the construction of the measurement vector $\mathbf{Z}^{\mathrm{DR}} = (z_1^{\mathrm{DR}}, z_2^{\mathrm{DR}} \cdots z_k^{\mathrm{DR}})$. z_k^{DR} is used for DR to estimate the ship state vector $\boldsymbol{X}_k$. A residual is then computed between this estimated value and the predicted value obtained through the state transition function f_k, and this residual is subsequently used to construct the DR factor f^{DR}:

$$f_k^{\mathrm{DR}}(\boldsymbol{X}_k, \boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1}) = \exp\left(-\frac{1}{2}\|\boldsymbol{X}_k - f_k(\boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1})\|_{\boldsymbol{\Omega}_k}^2\right) \tag{6}$$

$$f_k(\boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1}) = \begin{cases} \begin{pmatrix} \mathbf{p}_{k-1} + \boldsymbol{V}_{k-1}\Delta t + \frac{1}{2}\left(\nabla_{k-1}\sin\left(\theta_{k-1}^{\mathrm{T}}\right), \nabla_{k-1}\cos\left(\theta_{k-1}^{\mathrm{T}}\right)\right)\Delta t^2 \\ \boldsymbol{V}_{k-1} + \left(\nabla_{k-1}\sin\left(\theta_{k-1}^{\mathrm{T}}\right), \nabla_{k-1}\cos\left(\theta_{k-1}^{\mathrm{T}}\right)\right)\Delta t \\ \theta_{k-1}^{\mathrm{T}} + \varepsilon_{k-1}\Delta t \end{pmatrix} + o_k \end{cases} \tag{7}$$

where o_k is the process noise, which is modeled as a Gaussian distribution with zero-mean and covariance matrix $\boldsymbol{Q}$. The covariance matrix of the DR factor f_k^{DR} is given by:

$$\boldsymbol{\Omega}_k = \boldsymbol{F}_k \boldsymbol{\Omega}_{k-1} \boldsymbol{F}_k^T + \boldsymbol{Q} \tag{8}$$

where $\boldsymbol{F}_k$ is the Jacobian matrix of f_k with respect to $\boldsymbol{X}_{k-1}$, calculated as follows:

$$\boldsymbol{F}_k = \partial f_k(\boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1}) / \partial \boldsymbol{X}_{k-1} \tag{9}$$

2.3. GNSS Factor

In the GNSS/DR factor graph, the position outputs from the GNSS receiver are used directly as external measurements. Thus, the GNSS observation model is:

$$\boldsymbol{z}_k^{\mathrm{GNSS}} = h_{\mathrm{GNSS}}(\boldsymbol{X}_k) + \boldsymbol{n}^{\mathrm{GNSS}} \tag{10}$$

where $\boldsymbol{z}_k^{\mathrm{GNSS}} = \left(x_{k,x}^{\mathrm{GNSS}}, y_{k,y}^{\mathrm{GNSS}}\right)$ denotes the position information measured by the GNSS receiver. $h_{\mathrm{GNSS}}(\boldsymbol{X}_k)$ represents the measurement equation for the GNSS observations. $\boldsymbol{n}^{\mathrm{GNSS}}$ denotes the measurement noise of the GNSS.

The GNSS factor uses the difference between the measured and predicted positions as the nonlinear optimization residual. Accordingly, the GNSS factor at epoch k is:

$$f_k^{\mathrm{GNSS}}(\boldsymbol{X}_k) = \exp\left(-\frac{1}{2}\left\|\boldsymbol{z}_k^{\mathrm{GNSS}} - h_{\mathrm{GNSS}}(\boldsymbol{X}_k)\right\|_{\boldsymbol{R}_k}^2\right) \tag{11}$$

where $\boldsymbol{R}_k$ is the measurement covariance matrix of the f_k^{GNSS}.

2.4. Marginalization Factor

To enable a unified marginalization framework in sliding window optimization, an augmented variable $\boldsymbol{Y} = (\boldsymbol{X}, \boldsymbol{C})$ is defined to jointly encapsulate the state variables and constraint terms. Marginalization is triggered when the window size exceeds threshold m; otherwise, the window is maintained.

$\boldsymbol{Y}_k^{\mathrm{M}} = \{(\boldsymbol{X}_{k-m}, \cdots, \boldsymbol{X}_k), (\boldsymbol{C}_{k-m}, \cdots, \boldsymbol{C}_k)\}$ and $\boldsymbol{Z}_k^{\mathrm{GNSS}} = \left\{\boldsymbol{z}_{k-m}^{\mathrm{GNSS}}, \cdots, \boldsymbol{z}_k^{\mathrm{GNSS}}\right\}$ represent the relevant vectors within the sliding window at epoch k, respectively. After the marginalization operation, the retained relevant vectors within the window are represented as $\boldsymbol{Y}_k^{\mathrm{r}} = \{(\boldsymbol{X}_{k-m+1}, \cdots, \boldsymbol{X}_k), (\boldsymbol{C}_{k-m+1}, \cdots, \boldsymbol{C}_k)\}$. Additionally, $\boldsymbol{Y}_k^{\mathrm{m}} = \{\boldsymbol{X}_{k-m}, \boldsymbol{C}_{k-m}\}$ denotes the variables to be marginalized in the sliding window.

After linearization of the nonlinear residual functions associated with the factors within the sliding window, the joint distribution of the current states $\boldsymbol{Y}_k^{\mathrm{r}}$ and the states to be marginalized $\boldsymbol{Y}_k^{\mathrm{m}}$ follows a Gaussian distribution, which is formulated as follows (12):

$$p\left(\boldsymbol{Y}_k^{\mathrm{r}}, \boldsymbol{Y}_k^{\mathrm{m}} \middle| \boldsymbol{Z}_k^{\mathrm{GNSS}}\right) = N\left(\begin{bmatrix}\mu_k^{\mathrm{r}} \\ \mu_k^{\mathrm{m}}\end{bmatrix}, \begin{bmatrix}\boldsymbol{P}_{rr} & \boldsymbol{P}_{rm} \\ \boldsymbol{P}_{mr} & \boldsymbol{P}_{mm}\end{bmatrix}\right) \tag{12}$$

where μ_k^{r} and μ_k^{m} represent the posterior means of the $\boldsymbol{Y}_k^{\mathrm{r}}$ and $\boldsymbol{Y}_k^{\mathrm{m}}$ conditioned on the GNSS measurements $\boldsymbol{Z}_k^{\mathrm{GNSS}}$. Matrices $\boldsymbol{P}_{rr}$ and $\boldsymbol{P}_{mm}$ represent the marginal covariances, while $\boldsymbol{P}_{rm}$ and $\boldsymbol{P}_{mr}$ denote the cross-covariances.

By the conditional property of Gaussian distributions, the retained states $\boldsymbol{Y}_k^{\mathrm{r}}$, conditioned on marginalized states $\boldsymbol{Y}_k^{\mathrm{m}}$, can be expressed as follows [31]:

$$\mu_{\mathrm{r|m}} = \mu_k^{\mathrm{r}} + \boldsymbol{P}_{rm}\boldsymbol{P}_{mm}^{-1}(\boldsymbol{y}_k^{\mathrm{m}} - \mu_k^{\mathrm{m}}) \tag{13}$$

$$\boldsymbol{P}_{\mathrm{r|m}} = \boldsymbol{P}_{rr} - \boldsymbol{P}_{rm}\boldsymbol{P}_{mm}^{-1}\boldsymbol{P}_{mr} \tag{14}$$

where $\boldsymbol{y}_k^{\mathrm{m}}$ denotes the linearization point of $\boldsymbol{Y}_k^{\mathrm{m}}$.

This gives the conditional posterior of $\boldsymbol{Y}_k^{\mathrm{r}}$ as a Gaussian distribution:

$$p\left(\boldsymbol{Y}_k^{\mathrm{r}} \middle| y_k^{\mathrm{m}}, \boldsymbol{Z}_k^{\mathrm{GNSS}}\right) = N\left(\boldsymbol{Y}_k^{\mathrm{r}}; \mu_{\mathrm{r|m}}, \boldsymbol{P}_{\mathrm{r|m}}\right) \tag{15}$$

In FGO, this conditional Gaussian distribution is equivalent to introducing a marginalization factor that imposes an identical quadratic form, thereby yielding a new prior constraint on the retained states:

$$f_k^{\mathrm{M}}(\boldsymbol{Y}_k^{\mathrm{r}}) \propto \exp\left(-\frac{1}{2}\left\|\boldsymbol{Y}_k^{\mathrm{r}} - \mu_{\mathrm{r|m}}\right\|_{\boldsymbol{P}_{r|m}}^2\right) \tag{16}$$

2.5. FGO–Solving Process

The relationships between the ship state vector and measurement vector can be modeled utilizing the DR factor f^{DR}, DR error factor f^{err}, GNSS factor f^{GNSS}, and marginalization factor f^{M} based on the factor graph. In this way, the joint probability density

function of the system can be constructed. This study formulates the optimal ship position estimation as a Maximum A Posteriori (MAP) problem [32]:

$$\hat{\boldsymbol{Y}} = \underset{\boldsymbol{Y}}{\operatorname{argmax}}\, p\left(\boldsymbol{Y}\middle|\boldsymbol{Z}^{\mathrm{DR}}, \boldsymbol{Z}^{\mathrm{GNSS}}\right) \tag{17}$$

where $\hat{\boldsymbol{Y}} = \left\{(\hat{\boldsymbol{X}}_0, \hat{\boldsymbol{X}}_1, \cdots, \hat{\boldsymbol{X}}_k), (\hat{\boldsymbol{C}}_0, \hat{\boldsymbol{C}}_1, \cdots, \hat{\boldsymbol{C}}_k)\right\}$ represents the optimal estimates of $\boldsymbol{Y}$. $(\hat{\boldsymbol{X}}_0, \hat{\boldsymbol{X}}_1, \cdots, \hat{\boldsymbol{X}}_k)$ and $(\hat{\boldsymbol{C}}_0, \hat{\boldsymbol{C}}_1, \cdots, \hat{\boldsymbol{C}}_k)$ denote the set of optimal estimates of the ship state vector and the DR error vector, respectively.

Based on the factor graph, the joint probability density function is factorized a product of factors associated with the state vector, DR error vector and measurement vector:

$$p\left(\boldsymbol{Y}\middle|\boldsymbol{Z}^{\mathrm{DR}}, \boldsymbol{Z}^{\mathrm{GNSS}}\right) \propto \begin{cases} \prod_{i=1}^{m} f_i^{\mathrm{DR}}(\boldsymbol{X}_i, \boldsymbol{X}_{i-1}, \boldsymbol{C}_{i-1}) f_i^{\mathrm{err}}(\boldsymbol{C}_i, \boldsymbol{C}_{i-1}) f_i^{\mathrm{GNSS}}(\boldsymbol{X}_i) & if\ 0 < i \le m \\ \prod_{i=m+1}^{k} f_i^{\mathrm{DR}}(\boldsymbol{X}_i, \boldsymbol{X}_{i-1}, \boldsymbol{C}_{i-1}) f_i^{\mathrm{err}}(\boldsymbol{C}_i, \boldsymbol{C}_{i-1}) f_i^{\mathrm{GNSS}}(\boldsymbol{X}_i) f_i^{\mathrm{M}}(\boldsymbol{Y}_i^{\mathrm{r}}) & if\ m < i \le k \end{cases} \tag{18}$$

Furthermore, employing the MAP approach in the factor graph allows the optimal state estimation to be reformulated as a least-squares problem:

$$\hat{\boldsymbol{Y}} = \underset{\boldsymbol{Y}}{\operatorname{argmin}} \sum_{k} \left[\left\| \boldsymbol{Y}_k^{\mathrm{r}} - \sigma_k^{P} \right\|_{\boldsymbol{P}}^2 + \left\| \boldsymbol{X}_k - f_k(\boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1}) \right\|_{\boldsymbol{\Omega}_k}^2 + \left\| \boldsymbol{C}_k - f_k^{\mathrm{c}}(\boldsymbol{C}_{k-1}) \right\|_{\zeta_\kappa}^2 + \left\| \boldsymbol{z}_k^{\mathrm{GNSS}} - h_{\mathrm{GNSS}}(\boldsymbol{X}_k) \right\|_{\boldsymbol{R}_k}^2 \right] \tag{19}$$

The least-squares problem is addressed using Google's Ceres Solver, a robust library tailored for FGO applications. Specifically, the Levenberg–Marquardt method is applied to Equation (19) to obtain the optimal estimate $\hat{\boldsymbol{Y}}$.

3. ANFIS–FGO Ship Positioning Algorithm

An ANFIS–FGO algorithm is proposed for ship GNSS/DR integrated navigation, aiming to improve positioning accuracy and robustness in complex marine environments. This chapter details the ANFIS model and algorithm architecture.

3.1. ANFIS Model

3.1.1. ANFIS Model Construction

This study employs ANFIS in a combined offline training and online inference mode. In the offline training phase, a hybrid learning mechanism is used to train and optimize the ANFIS model parameters through historical data containing typical motion patterns, thereby learning the nonlinear mapping relationship between ship motion characteristics and GNSS measurement noise. In the online inference phase, the trained model is deployed with fixed parameters in the ship navigation system to perform forward inference based on real-time ship motion characteristics, output predicted standard deviation values of GNSS measurement noise, and construct GMNCMs based on these predictions to participate in FGO. Therefore, the construction of the ANFIS model is introduced first. For the ship GNSS/DR integrated navigation system, the ANFIS model takes two inputs: (1) $\Delta e^{\mathrm{DR}} = \left\| \boldsymbol{p}_k - \boldsymbol{p}_{k-1} \right\|$ represents the position increment from the DR system; (2) $\Delta e^{\mathrm{DIF}} = \left\| \boldsymbol{z}_k^{\mathrm{GNSS}} - \boldsymbol{p}_k \right\|$ represents the position innovation from the GNSS/DR integrated system. These inputs capture system dynamics and error characteristics, comprehensively representing the navigation state. Additionally, $\widetilde{\alpha}$ is the ANFIS output label. It represents the standard deviations corresponding to the diagonal elements of the GMNCM, reflecting measurement data confidence. By learning the relationship between the navigation state and measurement data reliability, ANFIS dynamically tunes the measurement weights within the FGO framework.

Figure 2 illustrates that the ANFIS architecture consists of five sequential layers, including fuzzification, rule, normalization, consequent, and output, each performing a distinct computational function. The fuzzification layer converts crisp inputs into fuzzy sets; the rule layer forms fuzzy inference rules; the normalization layer computes normalized firing strengths; the consequent layer generates rule-specific outputs; and the final output layer aggregates these results to produce the overall system output.

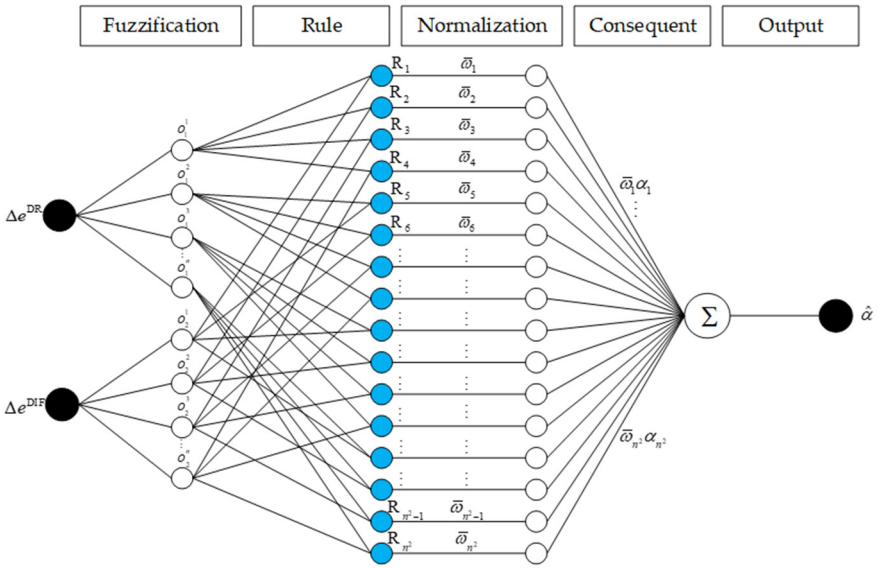


Figure 2. ANFIS network architecture.

In the fuzzification layer, membership functions are employed to compute the fuzzy membership degrees of the input variables. The resulting membership values vary depending on the specific type of membership function selected. This study adopts the triangular membership function to optimize computational efficiency and ensure actual performance, which can be mathematically expressed as follows:

$$\mu_{O_1^n}^{\mathrm{DR}}\left(\Delta e^{\mathrm{DR}};a^{\mathrm{DR}},b^{\mathrm{DR}},c^{\mathrm{DR}}\right)=\begin{cases}0, & if\ \Delta e^{\mathrm{DR}}\le a^{\mathrm{DR}}\\ \frac{\Delta e^{\mathrm{DR}}-a^{\mathrm{DR}}}{b^{\mathrm{DR}}-a^{\mathrm{DR}}}, & if\ a^{\mathrm{DR}}<\Delta e^{\mathrm{DR}}\le b^{\mathrm{DR}}\\ \frac{c^{\mathrm{DR}}-\Delta e^{\mathrm{DR}}}{c^{\mathrm{DR}}-b^{\mathrm{DR}}}, & if\ b^{\mathrm{DR}}<\Delta e^{\mathrm{DR}}<c^{\mathrm{DR}}\\ 0, & if\ \Delta e^{\mathrm{DR}}\ge c^{\mathrm{DR}}\end{cases} \tag{20}$$

$$\mu_{O_2^n}^{\mathrm{DIF}}\left(\Delta e^{\mathrm{DIF}};a^{\mathrm{DIF}},b^{\mathrm{DIF}},c^{\mathrm{DIF}}\right)=\begin{cases}0, & if\ \Delta e^{\mathrm{DIF}}\le a^{\mathrm{DIF}}\\ \frac{\Delta e^{\mathrm{DIF}}-a^{\mathrm{DIF}}}{b^{\mathrm{DIF}}-a^{\mathrm{DIF}}}, & if\ a^{\mathrm{DIF}}<\Delta e^{\mathrm{DIF}}\le b^{\mathrm{DIF}}\\ \frac{c^{\mathrm{DIF}}-\Delta e^{\mathrm{DIF}}}{c^{\mathrm{DIF}}-b^{\mathrm{DIF}}}, & if\ b^{\mathrm{DIF}}<\Delta e^{\mathrm{DIF}}<c^{\mathrm{DIF}}\\ 0, & if\ \Delta e^{\mathrm{DIF}}\ge c^{\mathrm{DIF}}\end{cases} \tag{21}$$

where $\mu_{O_1^n}^{\mathrm{DR}}\left(\Delta e^{\mathrm{DR}};a^{\mathrm{DR}},b^{\mathrm{DR}},c^{\mathrm{DR}}\right)$ and $\mu_{O_2^n}^{\mathrm{DIF}}\left(\Delta e^{\mathrm{DIF}};a^{\mathrm{DIF}},b^{\mathrm{DIF}},c^{\mathrm{DIF}}\right)$ are the triangular membership function. O_1^n and O_2^n represent the fuzzy sets for inputs and Δe^{DR} and Δe^{DIF}, respectively. n indicates the number of membership functions for each input. Considering the trade-off between input variable distribution and nonlinear modeling ability, this study assigns four membership functions to each input. $\left\{a^{\mathrm{DR}},b^{\mathrm{DR}},c^{\mathrm{DR}}\right\}$ and $\left\{a^{\mathrm{DIF}},b^{\mathrm{DIF}},c^{\mathrm{DIF}}\right\}$ denote the boundary parameters of each function, ensuring the existence of corresponding membership values for any given input.

The IF–THEN principle establishes linear input–output relationships in the rule layer. The number of generated rules depends on the input variables and their corresponding

membership functions. In this study, with four membership functions assigned per input variable, the resulting rules can be expressed as follows:

$$\mathrm{R}_j:\ \text{IF } \Delta e^{\mathrm{DR}} \text{ is } O_1^4 \text{ AND } \Delta e^{\mathrm{DIF}} \text{ is } O_2^4 \text{ THEN } \alpha_j = q_0^j + q_1^j \Delta e^{\mathrm{DR}} + q_2^j \Delta e^{\mathrm{DIF}} \tag{22}$$

where R_j denotes the rule index, and $1 \le j \le n^2$. In this study, the total number of fuzzy rules is set to 16. The firing strength of each rule is computed either through the product or the minimum operation of the membership degrees. α_j indicates the output value derived from each triggered rule. $q_k^j (k = 0, 1, 2)$ represents the linear relationship parameters between inputs and outputs within each rule.

For the normalization layer, the firing strength ω_j of each rule is calculated using a product-based method, expressed mathematically as follows:

$$\omega_j = \mu_{O_1^4}^{\mathrm{DR}}\left(\Delta e^{\mathrm{DR}}\right) \cdot \mu_{O_2^4}^{\mathrm{DIF}}\left(\Delta e^{\mathrm{DIF}}\right) \tag{23}$$

To ensure that the sum of the firing strengths of all rules equals one, a normalization procedure is applied. The normalized firing strength $\overline{\omega}_j$ is:

$$\overline{\omega}_j = \frac{\omega_j}{\sum_{i=1}^{16} \omega_i} \tag{24}$$

In the consequent layer, the normalized firing strength $\overline{\omega}_j$ is multiplied by the output α_j of each corresponding rule to obtain the weighted output $\overline{\omega}_j \alpha_j$:

$$\overline{\omega}_j \alpha_j = \overline{\omega}_j \left(q_0^j + q_1^j \Delta e^{\mathrm{DR}} + q_2^j \Delta e^{\mathrm{DIF}}\right) \tag{25}$$

The ANFIS model ultimately outputs a prediction $\hat{\alpha}$ corresponding to the label $\widetilde{\alpha}$ at the output layer.

$$\hat{\alpha} = \sum_{j=1}^{16} \overline{\omega}_j \alpha_j \tag{26}$$

3.1.2. ANFIS Training Methodology

ANFIS performance relies critically on the parameters in its fuzzification and consequent layers. This study employs a Hybrid Learning Algorithm (HLA) that combines gradient descent for fuzzification layer tuning with least-squares estimation for consequent layer updating. This integrated approach enhances training efficiency and accelerates convergence while minimizing prediction error.

During the forward pass, the ANFIS output can be expressed as a linear function of the consequent parameters. Accordingly, Equation (26) can be reformulated as follows:

$$\hat{\alpha} = \boldsymbol{\lambda}^T \phi\left(\Delta e^{\mathrm{DR}}, \Delta e^{\mathrm{DIF}}\right) \tag{27}$$

where $\boldsymbol{\lambda}$ denotes the parameter vector of the linear functions in the consequent layer. $\phi\left(\Delta e^{\mathrm{DR}}, \Delta e^{\mathrm{DIF}}\right)$ represents the corresponding input feature vector. Given the linear structure of the network at this stage, the output error can be efficiently minimized by directly solving the resulting system of linear equations using the least-squares method, yielding a rapid estimation of the consequent parameters as follows:

$$\boldsymbol{\lambda} = \left(\phi^T \phi\right)^{-1} \phi^T \widetilde{\alpha} \tag{28}$$

In the backward pass, ANFIS employs gradient descent to update the parameters of the membership functions while keeping the linear parameters fixed. In this study, each input variable is modeled using a three-parameter membership function, whose parameters a, b, and c control the width, slope, and center of the membership function, respectively. The error function is defined as follows:

$$\partial E = \frac{1}{2}(\widetilde{\alpha} - \hat{\alpha})^2 \tag{29}$$

Utilizing the chain rule, the gradients of the boundary parameters are computed and updated iteratively as follows:

$$a \leftarrow a - \eta \frac{\partial E}{\partial a},\ b \leftarrow b - \eta \frac{\partial E}{\partial b},\ c \leftarrow c - \eta \frac{\partial E}{\partial c} \tag{30}$$

HLA alternately applies gradient descent and least-squares estimation to improve convergence efficiency. In ship navigation, ANFIS uses this approach to dynamically estimate measurement covariance, boosting accuracy and reliability.

3.1.3. ANFIS Training Process

The ANFIS model is developed using MATLAB–R2018b's Neuro-Fuzzy Designer, which facilitates efficient configuration of inputs, fuzzy rules, and training. The model predicts the adaptive coefficient $\hat{\alpha}$ by analyzing the relationship between the input features Δe^{DR} and Δe^{DIF}. The trained ANFIS–FGO algorithm adaptively adjusts the GMNCM, improving positioning accuracy and robustness in complex marine environments. As shown in Figure 3, the training process includes data preparation, initialization, alternating training, and model saving after convergence.

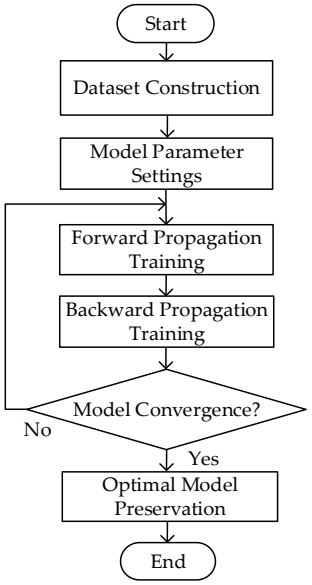


Figure 3. ANFIS model training process.

Specifically, the ANFIS training process consists of four main steps:

Step 1: Input features are derived from DR increment data and processed via a logarithmic transform to suppress outliers and enhance temporal dynamics. The output label $\hat{\alpha}$ is obtained from the converged FGO solution.

Step 2: The model parameters are configured, including membership functions, training epochs, and error tolerance. Three-parameter membership functions are adopted for each input.

Step 3: The model is trained using an alternating forward and backward propagation scheme based on the HLA method. During training, predicted errors progressively decrease as the model captures the nonlinear relationships between inputs and outputs.

Step 4: Training concludes once the model reaches the predefined error threshold or maximum number of training epochs, after which the optimal ANFIS model is saved.

3.2. ANFIS–FGO Algorithm

3.2.1. GNSS Adaptive Factor Construction Based on ANFIS

After completing the training of the ANFIS model in the offline training phase, the online inference phase begins, where the trained model is deployed with fixed parameters in the ship navigation system. In this phase, ANFIS performs forward inference based on real-time input features Δe^{DR} and Δe^{DIF}, outputting predicted standard deviation values $\hat{\alpha}$ of GNSS measurement noise that accurately reflect the current state. The $\hat{\alpha}$ is then structured into a matrix form, yielding the predicted Measurement Noise Covariance Matrix $\hat{\boldsymbol{R}}_k$ for the GNSS adaptive factor as follows:

$$\hat{\boldsymbol{R}}_k = \begin{bmatrix} \hat{\alpha}^2 & 0 \\ 0 & \hat{\alpha}^2 \end{bmatrix} \tag{31}$$

Accordingly, the GNSS adaptive factor is given by:

$$g_k^{\mathrm{GNSS}}(\boldsymbol{X}_k) = \exp\left(-\frac{1}{2} \left\| \boldsymbol{z}_k^{\mathrm{GNSS}} - h_{\mathrm{GNSS}}(\boldsymbol{X}_k) \right\|_{\hat{\boldsymbol{R}}_k}^2 \right) \tag{32}$$

Then, g_k^{GNSS} is integrated into the FGO to enable optimized estimation.

3.2.2. ANFIS–FGO Algorithm Process

With the addition of g_k^{GNSS}, the proposed ANFIS–FGO algorithm is fully established. The framework of the improved factor graph is illustrated in Figure 4.

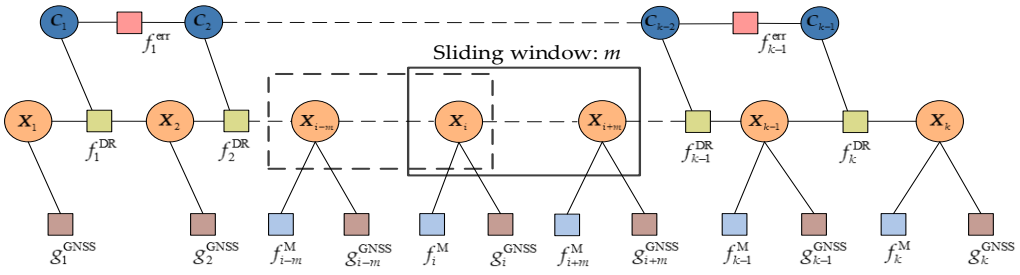


Figure 4. Improved GNSS/DR factor graph framework.

The overall process of the ANFIS–FGO algorithm is described as follows. First, the GNSS/DR factor graph is constructed as described in Chapter 2. Specifically, f^{DR} is formulated using the residual between the estimated $\boldsymbol{X}$ and its predicted value, while sensor error characteristics of the compass and log are incorporated to model f^{err}. f^{GNSS} is constructed utilizing $\boldsymbol{Z}^{\mathrm{GNSS}}$. In parallel, input features for the ANFIS model are derived from $\boldsymbol{Z}^{\mathrm{GNSS}}$ and consecutive $\boldsymbol{X}$.

The FGO process is then carried out by incorporating the combined state vector $\boldsymbol{X}$, error vector $\boldsymbol{C}$, and all previously constructed factors. During the optimization, if the number of $\boldsymbol{Y}$ exceeds a predefined sliding window threshold, a sliding window mechanism is activated, and marginalization is performed utilizing the Schur complement method to generate f^M, improving computational efficiency.

Moreover, under well-localized conditions, the standard deviation $\tilde{\alpha}$ corresponding to the diagonal elements of the GMNCM obtained during the FGO solution process is used as the output label for the ANFIS model. Once the input and output data are fully acquired, the ANFIS is trained and iteratively optimized based on performance metrics and stopping criteria to yield an optimal model. This model is then employed to adjust the GMNCM to generate the adaptive GNSS factor g_k^{GNSS}, which improves the performance of the FGO computation. Ultimately, this enhances the positioning accuracy and robustness of the maritime navigation system in complex environments.

4. Experimental Validation and Analysis

To comprehensively validate the proposed ANFIS–FGO method, this chapter employs simulation and actual ship experiments, evaluating performance across positioning accuracy, robustness, computational efficiency, and extreme condition adaptability. Regarding comparison methods, EKF and UKF are selected as representative benchmarks, representing predominant approaches in operational ship positioning systems. Basic FGO serves as the foundational framework for assessing ANFIS-driven improvements. Notably, most existing intelligent fusion methods cannot be directly applied to ship positioning due to inherent limitations, making comparisons inappropriate. Consequently, EKF, UKF, and FGO provide suitable benchmarks for fair evaluation.

The experimental design comprises four components. First, simulation experiments incorporate three typical ship motion scenarios: uniform acceleration, constant speed turning, and complex maneuvering, representing routine maritime operations. Second, actual ship experiments using real sea data verify practicality and reliability. Third, computational efficiency analysis demonstrates real-time capability for platforms with limited resources. Fourth, GNSS outage experiments validate robustness under extreme conditions. Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) are adopted as quantitative evaluation metrics to comprehensively assess the positioning performance of the ANFIS–FGO method.

$$\mathrm{MAE} = \frac{1}{MN}\sum_{i=1}^{M}\sum_{k=1}^{N}\left|y_k^i - \hat{y}_k^i\right| \tag{33}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{MN}\sum_{i=1}^{M}\sum_{k=1}^{N}\left(y_k^i - \hat{y}_k^i\right)^2} \tag{34}$$

where y_k^i and $\hat{y}_k^i$ are the field value and predicted value of the algorithm output at epoch k in the i–th Monte Carlo simulation, respectively. N is the total time spent on the positioning process. M denotes the total number of Monte Carlo simulations conducted.

4.1. Simulation Experiments

4.1.1. Simulation Setup

Simulation data in this study were generated using Qt Creator, covering three common ship motion scenarios: constant acceleration linear motion (CALM), constant turn motion (CTM), and complex motion (CM). Motion parameters for each scenario are provided in Table 1. To ensure statistical reliability, 200 Monte Carlo simulations were conducted per scenario [33].

The parameters of the shipborne sensors utilized in the simulation experiments are listed in Table 2. The GNSS receiver, compass and log update frequencies are set to 1 Hz, and their noise characteristics are statistically stable. The initial value of the covariance matrix $\boldsymbol{P}^s$ is set as $0.01 \times \boldsymbol{I}_5$, and $\boldsymbol{I}_5$ is a five-dimensional identity matrix. The ship state vector, covariance matrix, and state transition process for filtering algorithms are the same

as $\boldsymbol{X}$, $\boldsymbol{P}$, and $f_k(\boldsymbol{X}_{k-1}, \boldsymbol{C}_{k-1})$ defined in Chapter 2 for FGO. Additionally, the measurement vector is given by $\boldsymbol{Z} = \left(\boldsymbol{Z}^{\mathrm{GNSS}}, \boldsymbol{Z}^{\mathrm{DR}}\right)$.

Table 1. Motion state parameters under CALM, CTM and CM scenarios.

Motion Scenario	Motion State	Speed (m/s)	Acceleration (m/s^2)	Heading (°)	Rate of Turn (°/min)	Time (min)
CALM	constant acceleration	0	0.01714	227	0	6
	constant speed	6.1728	0	227	0	8
	constant deceleration	6.1728	−0.01714	227	0	6
CTM	straight ahead	6.1728	0	135	0	5
	turn ahead	6.1728	0	135	−6	9
	straight ahead	6.1728	0	90	0	6
CM	constant acceleration	0	0.01714	50	0	6
	turn ahead	6.1728	0	50	5	8
	constant deceleration	6.1728	−0.01714	90	0	3
	turn ahead	3.0864	0	90	−5	9
	straight ahead	3.0864	0	45	0	3

Table 2. Shipborne sensor parameters.

Sensor	Parameter	Index (σ)	Frequency
compass	bias error	0.2°/h	1 Hz
	random walk	$0.2°/\sqrt{\mathrm{h}}$	
	incremental error		
log	bias error	0.257% ∗ ship speed m/s	
	random walk	$0.00514\ \mathrm{m/s}/\sqrt{\mathrm{h}}$	
	incremental error		
GNSS receiver	position standard deviation	10 m	

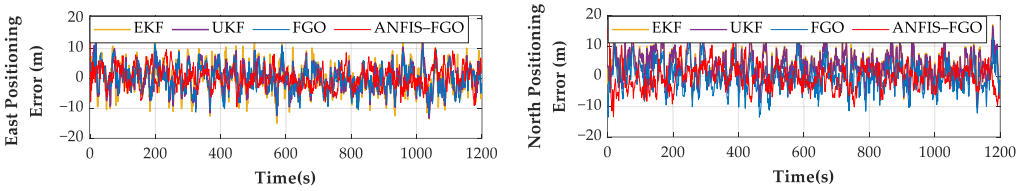
4.1.2. Simulation Experiments Under CALM, CTM and CM Scenarios

The simulation results of each algorithm under three distinct maritime motion scenarios are presented below, with performance evaluated in terms of east and north positioning errors, east and north velocity errors, and heading error.

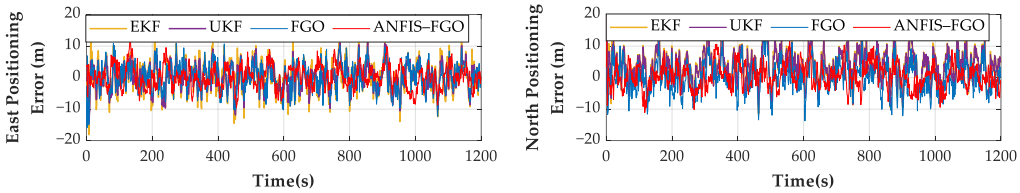
As shown in Figure 5, across the CALM, CTM, and CM scenarios, ANFIS–FGO consistently achieves the smallest east and north positioning errors, remaining within ±4–6 m after convergence. In contrast, FGO provides moderate positioning accuracy, with errors of approximately 6–12 m, and its performance degrades under highly dynamic and complex conditions due to the use of fixed noise assumptions. UKF and EKF exhibit larger deviations, particularly in CM scenarios, where EKF errors increase to 20–25 m, reflecting their sensitivity to strong nonlinearities and model mismatches. These results indicate that the adaptive error modeling enabled by ANFIS, together with the global optimization capability of the factor graph framework, significantly enhances the robustness of horizontal positioning under varying maritime motion patterns.

The simulation results presented in Figure 6 demonstrate that ANFIS–FGO consistently maintains east and north velocity errors within approximately 0.2–0.4 m/s across all three motion scenarios, exhibiting a clear performance advantage over the competing methods. In comparison, FGO yields velocity errors ranging from 0.3 to 0.6 m/s, while UKF and EKF exhibit significantly larger deviations, with errors escalating to approximately 1.2 m/s under fully dynamic conditions. This performance gain can be attributed to the adaptive error modeling capability of ANFIS, which effectively adjusts the contribution

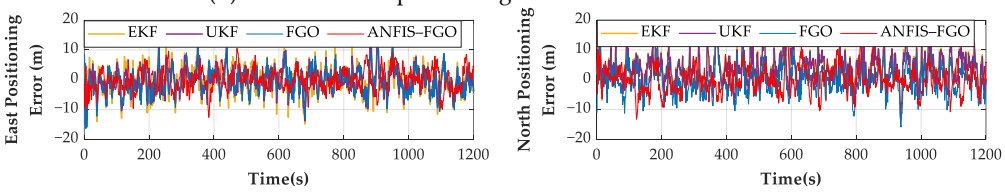
of DR–derived velocity information within the factor graph. In contrast, the fixed noise assumptions in FGO and the recursive estimation structure of UKF and EKF make them more sensitive to rapid motion changes and model mismatches, resulting in increased velocity errors.



(**a**) East and north positioning errors under CALM scenario.

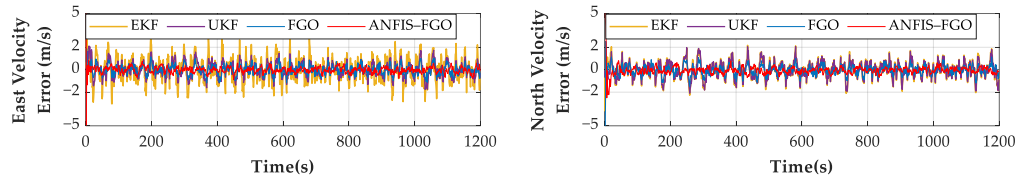


(**b**) East and north positioning errors under CTM scenario.

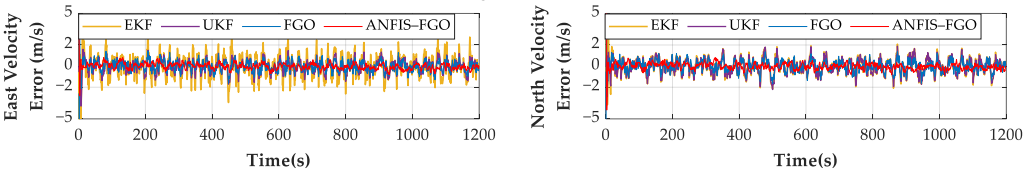


(**c**) East and north positioning errors under CM scenario.

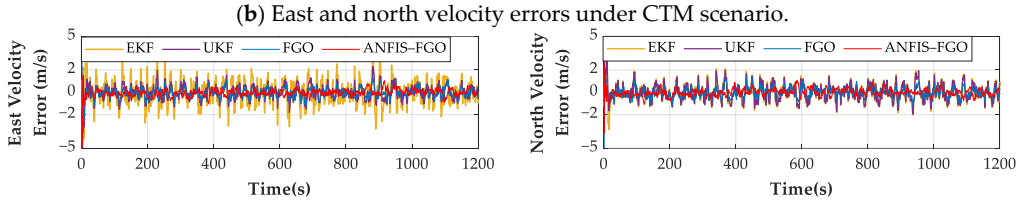
Figure 5. Comparison of horizontal positioning errors under CALM, CTM and CM scenarios.



(**a**) East and north velocity errors under CALM scenario.



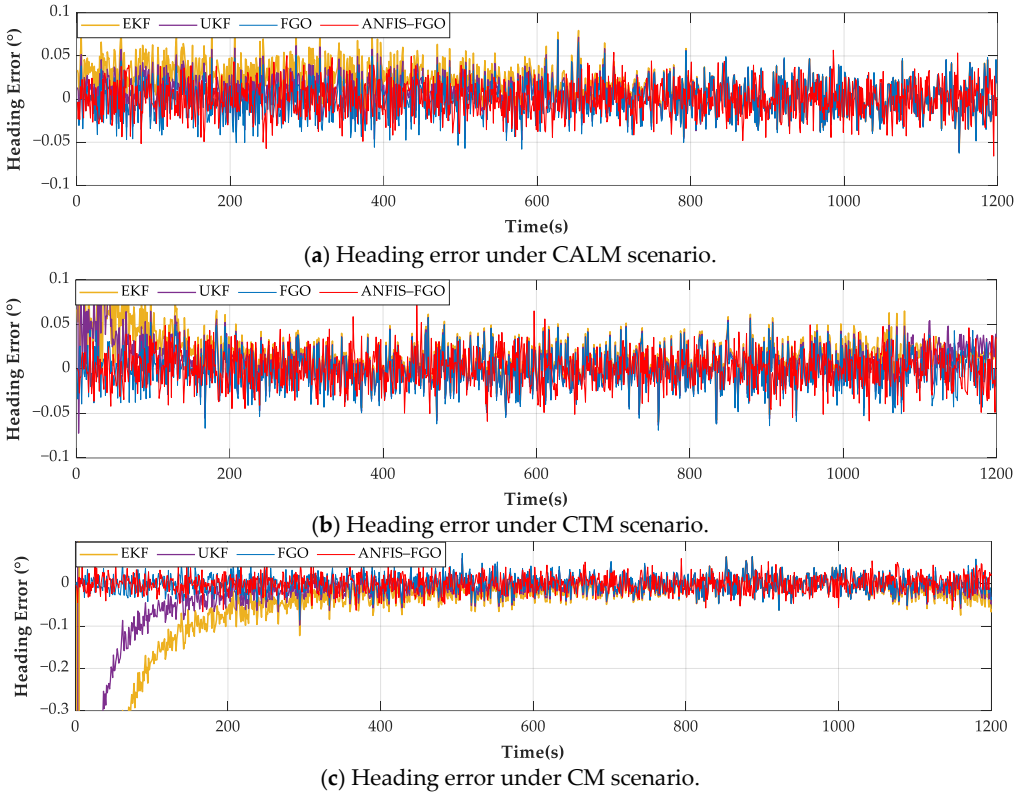
(**b**) East and north velocity errors under CTM scenario.



(**c**) East and north velocity errors under CM scenario.

Figure 6. Comparison of horizontal velocity errors under CALM, CTM and CM scenarios.

As shown in Figure 7, ANFIS–FGO consistently restricts heading errors to a range of approximately 0.04 to 0.05°, slightly surpassing the performance of FGO, which exhibits errors between 0.05 and 0.07°. In contrast, UKF and EKF demonstrate larger deviations, particularly under fully dynamic conditions, where maximum errors can exceed 0.1°. This improvement is mainly attributed to the adaptive weighting mechanism introduced by ANFIS, which effectively regulates the contribution of heading-related measurements under varying motion states. As a result, ANFIS–FGO maintains robust and reliable orientation estimation, even in complex and dynamically changing maritime environments.



(**a**) Heading error under CALM scenario.

(**b**) Heading error under CTM scenario.

(**c**) Heading error under CM scenario.

Figure 7. Comparison of heading errors under CALM, CTM and CM scenarios.

As evidenced by the quantitative results summarized in Table 3, ANFIS–FGO consistently achieves the best estimation performance across all three motion scenarios, namely CALM, CTM, and CM, demonstrating its robustness under varying maritime dynamics. Compared with EKF, UKF, and standard FGO, ANFIS–FGO reduces horizontal positioning errors by an average of 30–40%, with the most significant improvements observed in dynamic scenarios such as CM. Both MAE and RMSE values for east and north remain the lowest among all methods, typically around 2.7–3.4 m and 3.0–3.9 m, respectively, whereas the other filters often exceed 4–6 m, especially under complex maneuvers.

Table 3. Comparative error analysis under CALM, CTM and CM scenarios.

Scenario	Method	East Error (m)	North Error (m)	Velocity Error (m/s)	Heading Error (°)
Name	category	MAE/RMSE	MAE/RMSE	MAE/RMSE	MAE/RMSE
CALM	EKF	3.9329/4.9191	5.2281/6.4437	0.9086/1.1459	0.6559/0.8427
	UKF	3.3059/4.1456	5.0989/6.2855	0.5129/0.6440	0.6058/0.7654
	FGO	3.2132/4.0463	3.5643/4.4821	0.3364/0.4810	0.3360/0.4841
	ANFIS–FGO	2.7070/3.4214	3.0670/3.8547	0.2206/0.3472	0.2239/0.4086
CTM	EKF	3.8290/4.8047	5.1743/6.3379	0.9019/1.1326	0.6535/0.8415
	UKF	3.2486/4.0666	5.0501/6.1811	0.5084/0.6455	0.6052/0.7615
	FGO	3.1533/3.9775	3.4895/4.3760	0.3360/0.5223	0.3304/0.4873
	ANFIS–FGO	2.7116/3.3890	3.0563/3.8643	0.2155/0.4279	0.2228/0.4160
CM	EKF	3.8155/4.7696	5.3217/6.5322	0.8913/1.1272	0.6677/0.8528
	UKF	3.2458/4.0620	5.2075/6.3897	0.5086/0.6472	0.6248/0.7846
	FGO	3.2255/4.0232	3.8277/4.8001	0.3662/0.5187	0.4155/0.5899
	ANFIS–FGO	2.6941/3.3723	3.1000/3.8923	0.2220/0.4096	0.2439/0.4984

In terms of velocity and heading estimation, ANFIS–FGO exhibits significant advantages. The velocity errors decrease to approximately 0.22–0.35 m/s, corresponding to a reduction of nearly 60–70% compared with the EKF, and showing substantial improvement

over both UKF and FGO. Similarly, the heading errors are maintained between 0.22° and 0.25° for the MAE and between 0.41° and 0.50° for the RMSE, roughly half of the values observed for EKF and UKF. These enhancements are consistently achieved across the three representative motion patterns, demonstrating stable performance even during steady turning and under fully dynamic conditions.

Overall, the results confirm that ANFIS–FGO delivers the most accurate and robust navigation performance among all tested approaches. The integration of adaptive neuro-fuzzy inference enhances the FGO optimization process, enabling effective suppression of environmental disturbances and dynamic uncertainties. This makes ANFIS–FGO highly suitable for reliable and practical UMV navigation in real marine environments.

4.2. Field Experiments

4.2.1. Experiment Setup

Simulation results confirmed the superior positioning performance of the ANFIS–FGO algorithm in complex maritime scenarios. To validate its field efficacy, a field experiment was conducted using a ship near Tianjin Port, as illustrated in Figure 8.

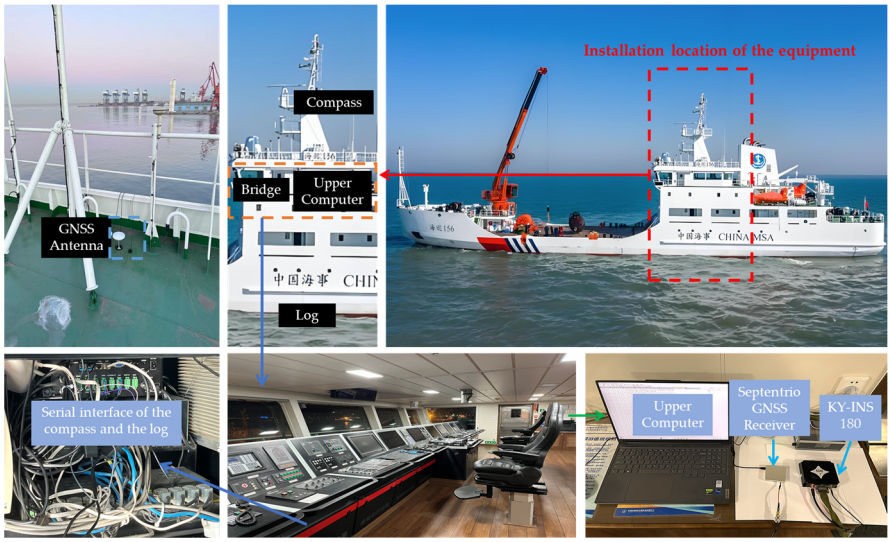


Figure 8. Working ship and experimental equipment.

The experiment lasted for one hour. Data from the shipborne compass and log were collected via serial interfaces. The high-precision KY–INS 180 system provided ground truth for evaluation, while the Septentrio GNSS receiver and shipborne sensors supplied algorithm inputs. In this study, the KY–INS 180 system is used as the reference, with a nominal positioning error of 2 cm. To ensure comparability between simulation and field tests, the algorithm configuration and parameter settings in the experiment were kept consistent with those used in the simulations.

The high-precision data obtained from the KY–INS180 are utilized as the reference trajectory to plot the field sailing route of the ship, as shown in Figure 9. In this figure, the black line represents the sailing path of the ship, the orange pentagram indicates the starting point, and the buff arrows denote the direction of travel. During the experiment, the ship maintained an approximate speed of 6.5 m/s, with a gradual deceleration of about -3.4×10^{-4} m/s^2. The heading shifted slightly from approximately 294° to 280°. This reference trajectory provides a reliable baseline for evaluating the performance of different navigation algorithms, allowing for quantitative comparison of position, velocity, and heading accuracy under realistic operating conditions. Moreover, the detailed depiction of

speed and heading changes ensures that subtle dynamic effects on sensor measurements are captured for comprehensive analysis.



Figure 9. Field navigation route of the ship.

4.2.2. Positioning Experiment

During the field test, sensor measurements were processed using the proposed ANFIS–FGO algorithm alongside three comparison methods: EKF, UKF, and conventional FGO. The resulting trajectories, illustrated in Figure 10, indicate that the ANFIS–FGO trajectory remains closely aligned with the reference path throughout the entire navigation process, exhibiting minimal fluctuations, smooth transitions, and the highest overall consistency.

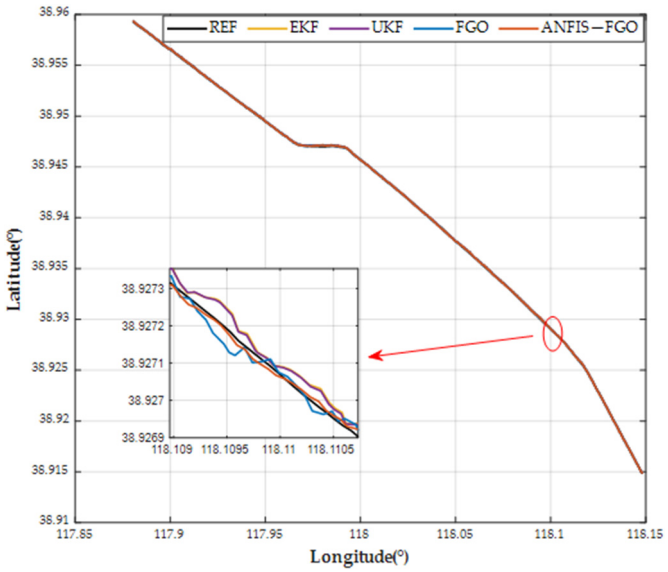


Figure 10. Comparison of trajectories estimated by each algorithm.

In comparison, the EKF, UKF, and FGO trajectories deviate noticeably at specific intervals, particularly during maneuvers or in the presence of environmental disturbances, and

display greater instability. These observations confirm that ANFIS–FGO provides superior positioning accuracy, enhanced robustness, and more reliable trajectory tracking under field maritime conditions, further validating its effectiveness and practical applicability for ship navigation operations.

In this study, only position error is evaluated in the experiments, as it directly affects navigation safety, particularly in complex environments. While velocity and heading errors are relevant, they are influenced by external factors and are more challenging to measure accurately. Position error provides a more reliable and straightforward metric for assessing algorithm performance. As shown in Figure 11, the ANFIS–FGO algorithm yields a minor positioning error and exhibits the least fluctuation in the error curves, indicating strong robustness. In contrast, the EKF algorithm demonstrates the most significant positioning error, while the UKF and FGO algorithms perform moderately, with error levels between those of EKF and ANFIS–FGO.

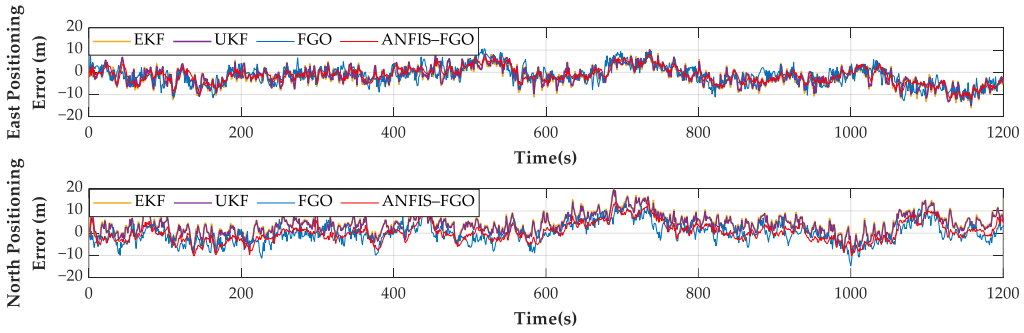


Figure 11. Comparison of the field experimental positioning errors of each algorithm.

Further analysis based on RMSE in Table 4 reveals that the ANFIS–FGO algorithm improves the overall positioning accuracy by approximately 30%, 25%, and 10% compared to the EKF, UKF, and FGO algorithms. These results demonstrate that the ANFIS–FGO algorithm significantly improves positioning accuracy and robustness in complex maritime environments compared to the EKF, UKF, and FGO algorithms, laying a solid technical foundation for the development and application of future UMV navigation and positioning systems.

Table 4. Analysis of the field experimental position errors of each algorithm.

ERROR	EKF MAE/RMSE	UKF MAE/RMSE	FGO MAE/RMSE	ANFIS–FGO MAE/RMSE
East positioning error (m)	3.2188/4.0317	3.0637/3.8328	3.0766/3.8902	2.6526/3.3538
North positioning error (m)	5.2120/6.4461	4.9966/6.2138	3.6462/4.5982	3.3803/4.3379

4.2.3. Efficiency Comparison Experiment

To comprehensively assess the practical feasibility of the ANFIS–FGO algorithm, an ablation study is conducted to provide a preliminary evaluation of its computational efficiency and resource consumption. In theory, the computational cost of ANFIS–FGO primarily originates from two components: the calculation process based on FGO and the training of the ANFIS model. As discussed in the introduction, ANFIS offers advantages such as a simple structure, a limited number of parameters, and reduced dependence on large training datasets. ANFIS requires significantly lower computational complexity and hardware resources than conventional deep learning models. Therefore, this study mainly focuses on analyzing the impact of the FGO module on computational performance. The improved ANFIS–FGO algorithm incorporating a sliding window approach and

marginalization factors are compared with the baseline version without these enhancements in the experimental design. Both versions are evaluated utilizing the same actual ship data to ensure fairness and consistency. The ANFIS configuration remains identical across all comparisons to maintain experimental control.

The experiments are conducted on a Windows 10 Professional system equipped with an Intel Core i5–7300HQ CPU and 8 GB of RAM. Given that this study primarily aims to improve positioning accuracy and robustness for the ship in complex marine environments, fifty Monte Carlo simulations were carried out to provide an initial exploration of the algorithm's computational complexity and resource usage. Future research will focus on a more detailed and multidimensional analysis of the computational performance and deployment ability of the proposed algorithm.

Figure 12 presents a comparative analysis of CPU utilization between the improved ANFIS–FGO algorithm and the baseline ANFIS–FGO across 50 Monte Carlo simulations. As illustrated in the figure, CPU utilization for both algorithms remains relatively stable throughout the simulations. However, the baseline ANFIS–FGO consistently exhibits higher CPU usage, ranging between 40% and 41%, whereas the improved version maintains a lower utilization rate of approximately 27.08%. The results demonstrate that the algorithm enhanced with a sliding window and marginalization factors significantly reduces computational demands, thereby improving its applicability and efficiency on resource-limited UMV platforms.

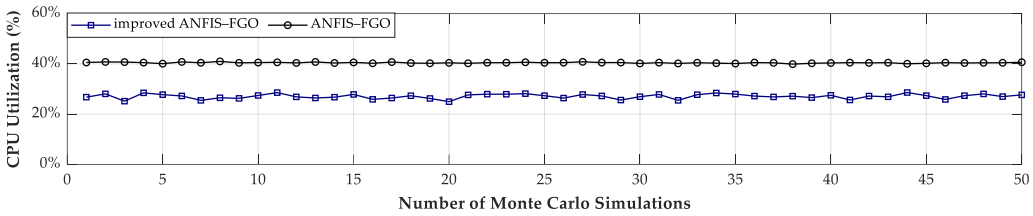


Figure 12. Comparison of CPU utilization.

Figure 13 compares execution times for the two algorithms. The improved ANFIS–FGO demonstrates significantly lower execution time, averaging around 23.8 s, in contrast to the baseline version, which approaches 381.4 s. This notable reduction indicates that the proposed improvements reduce computational resource demands and enhance overall efficiency. These results collectively validate the improved ANFIS–FGO algorithm's computational advantages. The integration of optimization mechanisms has effectively decreased CPU usage and execution time, thereby supporting its suitability for real-time or resource-constrained maritime applications.

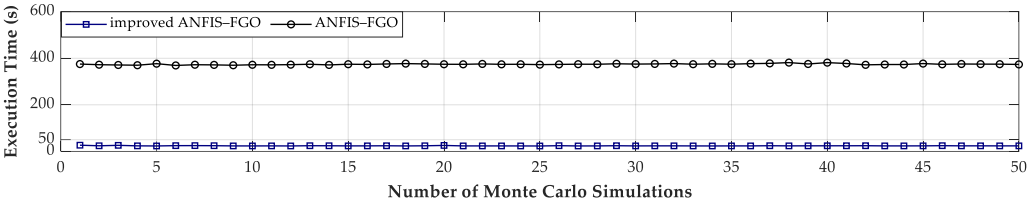


Figure 13. Comparison of execution time.

4.2.4. GNSS Outage Experiment

Existing studies on navigation algorithms in challenging environments have largely focused on system behavior during GNSS–denied periods. In this work, while the primary objective is to enhance positioning accuracy under nominal operating conditions, a GNSS outage experiment is additionally conducted to examine the robustness of the proposed

factor graph-based method. The evaluation is performed using both the CM scenario and real experimental data under consistent test conditions.

During the experiments, GNSS signals were intentionally interrupted twice. The outages were introduced at 500 s and 1100 s in the simulation, and at 400 s and 900 s in the field tests, with each interruption lasting 60 s. In Figure 14, GNSS interruptions in the CM scenario led to substantial error growth for EKF and UKF, with maximum position deviations exceeding 500 m. In contrast, FGO effectively constrained the errors within 20 m, while the proposed ANFIS–FGO exhibited the smallest fluctuations and fastest recovery, demonstrating superior robustness and stability. These results confirm that traditional filtering methods diverge without GNSS updates, whereas FGO-based approaches preserve consistency by exploiting historical optimization.

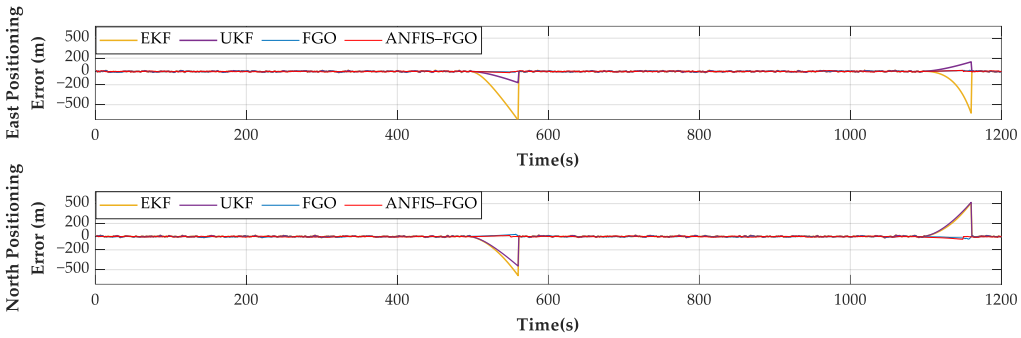


Figure 14. Positioning error comparison during GNSS outages in the CM scenario.

Similarly, as shown in Figure 15, during field ship navigation tests, EKF and UKF produced large position drifts of 200–400 m, whereas FGO maintained stable estimates within 20 m. The ANFIS–FGO algorithm achieved the highest accuracy and smoothest trajectories, consistent with the simulation findings. Overall, the results verify that FGO-based algorithms outperform conventional filters under both normal and GNSS denied conditions, with ANFIS–FGO exhibiting exceptional accuracy, resilience, and practical applicability for UMV navigation.

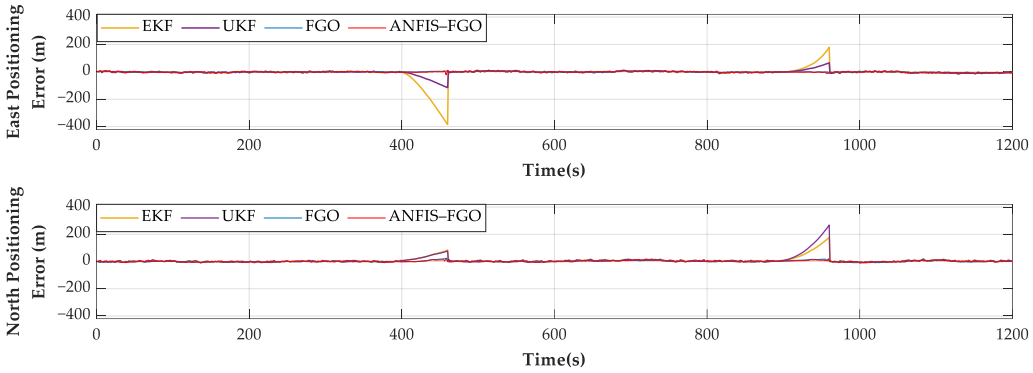


Figure 15. Positioning error comparison during GNSS outages in the field experiment.

5. Conclusions

This paper proposes an improved FGO algorithm enhanced with ANFIS, applied to ship GNSS/DR integrated navigation to improve positioning accuracy and robustness in complex marine environments. The proposed algorithm comprises two primary components. First, a novel GNSS/DR factor graph model tailored for ships is developed to address the limitations of traditional maritime navigation filtering methods. By incorporating a DR error factor, the model effectively enhances positioning accuracy. Meanwhile,

a sliding window strategy and the marginalization factor are introduced within the FGO framework to balance computational efficiency and estimation accuracy, ensuring the algorithm's suitability for resource-constrained UMVs. Second, the learning and adaptive capabilities of ANFIS are employed to optimize the FGO framework, further enhancing navigation accuracy and robustness in dynamically changing marine environments. Simulations and actual experiments are conducted to evaluate the proposed algorithm. These results demonstrate that the ANFIS–FGO algorithm achieves high positioning accuracy, strong robustness, and low computational cost, indicating its potential to ensure stable ship navigation in adverse conditions and support the development of intelligent UMV systems.

Although the proposed ANFIS–FGO algorithm has been validated through simulation experiments covering typical ship motion scenarios and real maritime data, certain limitations remain. Specifically, the method has not been extensively tested under more extreme conditions, such as extended-duration missions, sustained GNSS degradation, or highly dynamic sea states. These scenarios represent more challenging operational environments that warrant further investigation to comprehensively assess the performance boundaries and long-term stability of the proposed approach. Furthermore, other intelligent fusion methods are subject to inherent constraints, including data requirements and computational resource limitations, which make them difficult to directly apply to the ship positioning scenario addressed in this study.

In future work, we will advance the proposed method in three key directions. First, to address positioning requirements under extended-duration missions, complex sea states, and extreme conditions, we will further optimize the existing approach and explore multi-sensor fusion schemes to enhance system redundancy and robustness. Second, we will deepen the investigation of AI-enhanced FGO methods, seeking more efficient and broadly applicable intelligent optimization solutions. Finally, we will extend the method to cooperative localization scenarios for UMV fleets by establishing inter-ship information sharing and distributed optimization frameworks, which represent a critical direction for the development of maritime autonomous systems. These research efforts will provide theoretical foundations for the application of unmanned vessels in complex marine environments and facilitate their engineering deployment and practical implementation.

Author Contributions: Conceptualization, Y.J. and H.G.; methodology, J.X.; software, J.X.; validation, Q.H. and J.X.; formal analysis, H.G.; investigation, J.X.; resources, Y.J.; data curation, T.Z.; writing—original draft preparation, H.G. and J.X.; writing—review and editing, Y.J., H.G. and T.Z.; visualization, Y.Z. and J.K.; supervision, Y.J. and Q.H.; project administration, Y.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Distinguished Young Scholar Project of Dalian City, grant number 2024RJ012; the National Key Research and Development Program of China, grant number 2021YFB3901501; the Chinese National Science Foundation, grant number 62571081; and the Fundamental Research Funds for the Central Universities, grant number 3132025711.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Lun, Y.H.V.; Lai, K.H.; Cheng, T.C.E.; Yang, D. International Trade and Shipping. In *Shipping and Logistics Management*, 2nd ed.; Springer International Publishing: Cham, Switzerland, 2023; pp. 3–20.
2. International Maritime Organization (IMO). *MSC 99/WP. 9—Regulatory Scoping Exercise for the Use of Maritime Autonomous Surface Ships (MASS): Report of the Working Group*; Maritime Safety Committee of the International Maritime Organization: London, UK, 2018. Available online: https://www.imo.org/en/OurWork/Safety/Pages/MASS.aspx (accessed on 17 April 2025).

3. Alamoush, A.S.; Ölçer, A.I. Maritime Autonomous Surface Ships: Architecture for Autonomous Navigation Systems. *J. Mar. Sci. Eng.* **2025**, *13*, 122. [CrossRef]
4. U.S. Department of Homeland Security. *Resilient Positioning, Navigation, and Timing (PNT) Reference Architecture*; Science and Technology Directorate: Washington, DC, USA, 2022. Available online: https://www.dhs.gov/science-and-technology/publication/resilient-pnt-reference-architecture (accessed on 17 April 2025).
5. Medina, D.; Lass, C.; Marcos, E.P.; Ziebold, R.; Closas, P.; García, J. On GNSS Jamming Threat from the Maritime Navigation Perspective. In Proceedings of the 22nd International Conference on Information Fusion (FUSION), Ottawa, ON, Canada, 2–5 July 2019; pp. 1–7.
6. Skulstad, R.; Li, G.; Fossen, T.I.; Vik, B.; Zhang, H. Dead Reckoning of Dynamically Positioned Ships: Using an Efficient Recurrent Neural Network. *IEEE Robot. Autom. Mag.* **2019**, *26*, 39–51. [CrossRef]
7. Huang, T.; Liu, Y.; Xue, Z.; Wu, Y.; Zhang, C.; Jiang, X. USV–Tracker: A Novel USV Tracking System for Surface Investigation with Limited Resources. *Ocean Eng.* **2024**, *312*, 119196.
8. Guo, M.Z.; Guo, C.; Zhang, C. SINS/GNSS–Integrated Navigation of Surface Vessels Based on Various Nonlinear Kalman Filters and Large Ship Dynamics. *J. Electr. Eng. Technol.* **2021**, *16*, 531–546. [CrossRef]
9. Selimović, D.; Lerga, J.; Prpić–Oršić, J.; Kenji, S. Improving the Performance of Dynamic Ship Positioning Systems: A Review of Filtering and Estimation Techniques. *J. Mar. Sci. Eng.* **2020**, *8*, 234. [CrossRef]
10. Zhou, L.Q.; Wang, M.; Zhang, X.; Qin, P.; He, B. Adaptive SLAM Methodology Based on Simulated Annealing Particle Swarm Optimization for AUV Navigation. *Electronics* **2023**, *12*, 2372. [CrossRef]
11. Zhang, Z.; Li, Y.; Liu, Z.; Wang, S.; Xing, H.; Zhu, W. Enhancing the Reliability of Shipborne INS/GNSS Integrated Navigation System During Abnormal Sampling Periods Using Bi–LSTM and Robust CKF. *Ocean Eng.* **2023**, *288*, 115934.
12. Wen, W.; Pfeifer, T.; Bai, X.; Hsu, L.-T. Factor Graph Optimization for GNSS/INS Integration: A Comparison with the Extended Kalman Filter. *Navig. J. Inst. Navig.* **2021**, *68*, 315–331. [CrossRef]
13. Wu, X.; Xiao, B.; Wu, C.; Guo, Y.; Li, L. Factor Graph–Based Navigation and Positioning for Control System Design: A Review. *Chin. J. Aeronaut.* **2022**, *35*, 25–39. [CrossRef]
14. Chen, C.; Wang, Y.; Chen, Y.; Jiang, C.; Jia, J.; Bo, Y. An APDR/UPV/Light Multisensor–Based Indoor Positioning Method via Factor Graph Optimization. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 9516217. [CrossRef]
15. Lyu, P.; Wang, B.; Lai, J.; Bai, S.; Liu, M.; Yu, W. A Factor Graph Optimization Method for High–Precision IMU–Based Navigation System. *IEEE Trans. Instrum. Meas.* **2023**, *72*, 9509712. [CrossRef]
16. Zhang, L.; Gao, Y.; Guan, L. Optimizing AUV Navigation Using Factor Graphs with Side–Scan Sonar Integration. *J. Mar. Sci. Eng.* **2024**, *12*, 313. [CrossRef]
17. Huo, Z.; Jin, L.; Wang, H.; Sun, X.; He, Y. A Robust Factor Graph Optimization Method of GNSS/INS/ODO Integrated Navigation System for Autonomous Vehicle. *Meas. Sci. Technol.* **2025**, *36*, 016301.
18. Zhang, L.; Zhang, L.; Gao, R.; Pan, L.; Xu, C.; Cheng, K. Indoor Mobile Robot Localization Applying IMU/Stereo Camera/LiDAR and Graph–Based Optimization. *IEEE Sens. J.* **2024**, *24*, 21466–21478. [CrossRef]
19. Tian, Z.X.; Cheng, Y.M.; Yao, S. An Adaptive Fast Incremental Smoothing Approach to INS/GPS/VO Factor Graph Inference. *Appl. Sci.* **2024**, *14*, 5691. [CrossRef]
20. Chen, X.; Lin, Y.; Yang, X.; Zhao, S. The Tight–Coupled SLAM System Based on LiDAR and Improved VGICP Method for Waterfront Environments. *Ocean Eng.* **2025**, *326*, 120934. [CrossRef]
21. Ben, Y.; Wang, K.; Li, Q. A Robust Factor Graph Optimization Method for Navigation in Land Vehicles Based on Dynamic Kernel Principal Component Analysis. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 8503116. [CrossRef]
22. Xu, X.; Yang, X.; Lyu, P.; Li, L. Factor Graph Optimization Localization Method Based on GNSS Performance Evaluation and Prediction in Complex Urban Environment. *IEEE Sens. J.* **2025**, *25*, 12455–12465. [CrossRef]
23. Li, Z.; Meng, Q.; Shen, Z.; Wang, L.; Li, L.; Jia, H. Resilient Factor Graph–Based GNSS/IMU/Vision/Odo Integrated Navigation Scheme Enhanced by Noise Approximate Gaussian Estimation in Challenging Environments. *Remote Sens.* **2024**, *16*, 2176. [CrossRef]
24. Nam, D.V.; Gon-Woo, K. Learning Type–2 Fuzzy Logic for Factor Graph–Based Robust Pose Estimation with Multi–Sensor Fusion. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 3809–3821. [CrossRef]
25. Jang, J. ANFIS: Adaptive–Network–Based Fuzzy Inference System. *IEEE Trans. Syst. Man Cybern.* **1993**, *23*, 665–685. [CrossRef]
26. Sargunan, I.; Maram, B.; Rajan, C.; Prasad, V.; Balasubramaniam, S.R.; Nanasekaran, D. Discrete Neighborhood Difference Pattern–Based Feature Extraction and Hybrid Deep Learning Model for Melanoma Skin Cancer Detection. *Biomed. Signal Process. Control* **2025**, *107*, 107812. [CrossRef]
27. Darwito, P.A.; Indayu, N. Adaptive Neuro–Fuzzy Inference System Based on Sliding Mode Control for Quadcopter Trajectory Tracking with the Presence of External Disturbance. *J. Intell. Syst. Control* **2023**, *2*, 33–46. [CrossRef]
28. Ambuj; Machavaram, R. Intelligent Path Planning for Autonomous Ground Vehicles in Dynamic Environments Utilizing Adaptive Neuro–Fuzzy Control. *Eng. Appl. Artif. Intell.* **2025**, *144*, 110119. [CrossRef]

29. Qin, H.; Wang, X.; Wang, G.; Hu, M.; Bian, Y.; Qin, X.; Ding, R. A Novel INS/USBL/DVL Integrated Navigation Scheme against Complex Underwater Environment. *Ocean Eng.* **2023**, *286*, 115485. [CrossRef]
30. Kschischang, F.R.; Frey, B.J.; Loeliger, H.-A. Factor Graphs and the Sum–product Algorithm. *IEEE Trans. Inf. Theory* **2001**, *47*, 498–519. [CrossRef]
31. Li, X.; Wu, F.; Chang, H.; Li, X.; Tan, Y.; Shen, Z.; Zhang, X. Decimeter–level vehicular positioning in urban environments with factor graph–based PPP and robust regression. *GPS Solut.* **2025**, *29*, 134. [CrossRef]
32. Dong, X.; Hu, G.; Gao, B.; Zhong, Y.; Ruan, W. Windowing–Based Factor Graph Optimization with Anomaly Detection Using Mahalanobis Distance for Underwater INS/DVL/USBL Integration. *IEEE Trans. Instrum. Meas.* **2024**, *73*, 8501213. [CrossRef]
33. Stepanov, O.A.; Isaev, A.M. A Procedure of Comparative Analysis of Recursive Nonlinear Filtering Algorithms in Navigation Data Processing Based on Predictive Simulation. *Gyroscopy Navig.* **2023**, *14*, 213–224. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.