The Aeronautical Journal (2025), **129**, pp. 788–801
doi:10.1017/aer.2024.90

RESEARCH ARTICLE

Collaborative navigation method based on adaptive time-varying factor graph

H. Wang , L. Hu and J. Tao

School of Information and Communication Engineering, Harbin Engineering University, Harbin, China
Corresponding author: L. Hu; Email: hu35682022@163.com

Received: 28 May 2024; **Revised:** 26 July 2024; **Accepted:** 2 August 2024

Keywords: factor graph; cooperative navigation; UAV formation; double-threshold detection; robust estimation

Abstract

Aiming at the problems of poor coordination effect and low positioning accuracy of unmanned aerial vehicle (UAV) formation cooperative navigation in complex environments, an adaptive time-varying factor graph framework UAV formation cooperative navigation algorithm is proposed. The proposed algorithm uses the factor graph to describe the relationship between the navigation state of the UAV fleet and its own measurement information as well as the relative navigation information, and detects the relative navigation information at each moment by the double-threshold detection method to update the factor graph model at the current moment. And the robust estimation is combined with the factor graph, and the weight function measurements are used in the construction of the factor nodes for adaptive adjustment to make the system highly robust. The simulation results show that the proposed method realises the effective fusion of airborne multi-source sensing information and relative navigation information, which effectively improves the UAV formation cooperative navigation accuracy.

Nomenclature

$\delta \boldsymbol{v}$	velocity error
$\delta \boldsymbol{p}$	position error
∇_a	accelerometer zero bias
∇_g	gyroscope zero bias
t_k	time(s)
$\boldsymbol{x}_i^k$	UAV i at moment t_k are defined as variable nodes
$\boldsymbol{U}$	All UAVs in the cluster
$\boldsymbol{Z}^k$	the set of all measurements received by the UAV swarm at moment t_k
$\boldsymbol{X}_{MAP}^k$	the maximum a posteriori estimate of the state variable
$\boldsymbol{A}_i^k$	the local Jacobian matrix
$\boldsymbol{b}_i^k$	the local error vector
$\boldsymbol{C}_{i,j}$	direction cosine matrix
$\boldsymbol{d}_{i,j}^0$	the distance value of the drone calculated at the linearisation point

Greek symbol

$\boldsymbol{\phi}$	attitude
Σ_i	the sensor noise covariance matrix
∂_k	distance measurement residual
Δ_h	upper limit of dual threshold detection method
Δ_l	lower limit of dual threshold detection method

© The Author(s), 2025. Published by Cambridge University Press on behalf of Royal Aeronautical Society.

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

1.0 Introduction

Due to the continuous maturation of UAV-related technologies and the development of related industries, UAVs have become increasingly popular in civilian fields such as courier delivery, aerial photography and programme performance, as well as military reconnaissance and detection fields. This is due to their small size, low cost, flexible mobility and wide viewing angle [1]. Simultaneously, due to the growing diversity and complexity of tasks in both military and civilian fields, as well as the diversification of combat modes, the operational capacity of a single UAV is limited and unable to meet the demand. This has led to the development of UAV formation technology [2]. Accurate navigation and positioning information are crucial for formation technology. This is an essential guarantee for UAV swarm trajectory planning, stable flight, and smooth mission execution [3, 4].

Combined inertial/satellite navigation systems are currently the most widely used means of navigation for UAVs [5]. Real-time kinematic (RTK) technology, based on satellite navigation, can provide centimetre-level positioning accuracy for UAVs [6]. However, equipping all UAVs in the cluster with RTK equipment is costly. In addition, in some restricted environments, such as cities and forests, RTK technology may not be effective. The accuracy and reliability of navigation information provided by Global Navigation Satellite System (GNSS) receivers may be compromised due to their inability to search for a sufficient number of satellites [7].To fully utilise the benefits of clusters, scholars both domestically and internationally have started researching cooperative navigation methods that combine inter-UAV measurement information. Literature [8, 9] achieved the fusion of local and relative measurement information in a cooperative network through Kalman filtering. Reference (10) proposes the adaptive modified federal Kalman filter (FKF) algorithm, which introduces a dynamic information sharing factor of singular value decomposition to improve the efficiency of information allocation factors.The distributed collaborative navigation method based on Kalman filtering is simple to implement, but it only fuses the collaborative information of neighbouring nodes and does not make full use of the constraints of non-neighbouring nodes in the network. Factor graph (FG) is an effective framework for collaborative navigation problems [11]. It was first applied to the navigation algorithms of underwater submarines [12, 13] due to its ability to effectively fuse different types of sensors and its plug-and-play advantages [14]. It can also effectively coordinate the information fusion of each node in UAV clusters. Therefore, the Bayesian collaborative navigation method based on the FG framework is currently a popular research topic in the field of collaborative navigation. Literature [15] proposes the maximum entropy factor graph algorithm for the cooperative positioning problem of multiple UAVs under non-Gaussian heavy-tailed measurement noise. The experimental results demonstrate that its positioning accuracy is superior to that of the extended Kalman filter and the traceless Kalman filter algorithm. However, it is a centralised processing method. One disadvantage of this approach is that it requires a high communication bandwidth and significant computing power from the fusion node. Additionally, if the fusion node fails, the entire multi-unmanned system will lose its ability to perform cooperative positioning. Literature [16] proposes a distributed cooperative navigation method based on graph optimisation for densely clustered UAVs. The method combines the GNSS information of the UAVs and the distance information between them to establish a cooperative navigation factor graph. The factor graph is processed in a distributed manner through message iteration, which can construct a local graph model based on the distance between the UAVs, further simplifying the complexity of large-scale clustering. However, its single configuration condition is not applicable to complex and dynamic UAV flight scenarios.

This paper presents an adaptive time-varying factor graph framework for UAV formation cooperative navigation. The proposed method enhances the anti-interference ability of the UAV formation to cope with complex environments and achieves higher positioning accuracy while ensuring real-time performance. The relationship between the navigation state of the UAV swarm and its measurements, as well as the relative navigation information, is described using the factor graph. The factor graph model is updated at each moment using the relative navigation information. The factor graph algorithm is enhanced by integrating anti-differential estimation with the factor graph. This integration allows for adaptive adjustment based on current measurements, resulting in a highly robust system.

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

2.0 Methods

2.1 Distributed collaborative navigation factor graph model

The UAVs are equipped with a variety of sensors that have different output frequencies, including an inertial measurement unit (IMU) with high frequency output, an ultra-wide band (UWB) ranging module with lower frequency, and GNSS as a navigational device. These sensors are not synchronised, but they work together to achieve collaborative navigation of the UAV formation. The state variables $\boldsymbol{x}_i^k$ of UAV i at moment t_k are defined as variable nodes, which include attitude, velocity error, position error, accelerometer zero bias and gyroscope zero bias:

$$\boldsymbol{x}_i^k = \begin{bmatrix} \boldsymbol{\phi} & \delta\boldsymbol{v} & \delta\boldsymbol{p} & \nabla_a & \nabla_g \end{bmatrix}^{\mathrm{T}} \tag{1}$$

All UAVs in the cluster belong to the set $\boldsymbol{U}$. The navigation states of all UAVs in the cluster $\boldsymbol{U}$ at time t_k can be defined as:

$$\boldsymbol{X}^k = \{\boldsymbol{x}_i^k | i \in \boldsymbol{U}\} \tag{2}$$

The set of all measurements received by the UAV swarm at moment t_k can be defined as:

$$\boldsymbol{Z}^k = \{\boldsymbol{z}_i^{k-1,IMU}, \boldsymbol{z}_i^{k,UWB}, \boldsymbol{z}_i^{k,GNSS} | i \in \boldsymbol{U}\} \tag{3}$$

where $\boldsymbol{z}_i^{k-1,IMU}, \boldsymbol{z}_i^{k,UWB}, \boldsymbol{z}_i^{k,GNSS}$ are all IMU measurements received by UAV i from time t_{k-1} to time t_k, all UWB ranging information and GNSS measurements received at time t_k, respectively.

The joint probability distribution function of the navigation state of the UAV swarm can be obtained based on the above definition as:

$$p(\boldsymbol{X}^k | \boldsymbol{Z}^k) \tag{4}$$

Then the maximum a posteriori estimate of the state variable $\boldsymbol{X}$ is expressed as:

$$\boldsymbol{X}_{MAP}^k = \underset{\boldsymbol{X}^k}{\operatorname{argmax}}\, p(\boldsymbol{X}^k | \boldsymbol{Z}^k) \tag{5}$$

According to literature [13], the joint probability distribution Equation (4) can be factorised as:

$$\begin{aligned} p(\boldsymbol{X}^k \mid \boldsymbol{Z}^k) = p\left(\boldsymbol{X}^0\right) \cdot \prod_{t=1}^{k} (\boldsymbol{x}^t \mid \boldsymbol{z}^{t-1,IMU}, \boldsymbol{x}^{t-1}) \cdot \\ \prod_{t=1}^{k} p(\boldsymbol{z}^{t,GNSS} \mid x^t) \cdot \prod_{t=1}^{k} p(\boldsymbol{z}^{t,UWB} \mid \boldsymbol{x}_i^t, \boldsymbol{x}_j^t) \end{aligned} \tag{6}$$

The FG framework shown in Fig. 1 is constructed based on Equation (6), where $p(\boldsymbol{x}^t \mid z^{t-1,IMU}, \boldsymbol{x}^{t-1})$ represents the posterior probability density of IMU, $p(\boldsymbol{z}^{t,GNSS} \mid \boldsymbol{x}^t)$ represents the posterior probability density of GNSS, and $p(\boldsymbol{z}^{t,UWB} \mid \boldsymbol{x}_i^t, \boldsymbol{x}_j^t)$ represents the posterior probability density of UWB. Each sensor probability density can be described by an independent term of Equation (6), and each term corresponds to a factor node f (•) in the factor graph. Therefore, the probability density function can be written as a factor product:

$$p(\boldsymbol{X}^k \mid \boldsymbol{Z}^k) \propto \prod_i f_i\left(\boldsymbol{X}_i^k\right) \tag{7}$$

The local function f (•) is related to the error function $err_i\left(\boldsymbol{x}_i^k, \boldsymbol{z}^k\right)$, which is defined as:

$$f_i\left(\boldsymbol{x}_i^k\right) = d\left(err_i\left(\boldsymbol{x}_i^k, \boldsymbol{z}^k\right)\right) \tag{8}$$

where d (•) is the cost function.

In this paper, the input noise is assumed to conform to a zero-mean Gaussian distribution.The factor graph nodes for Gaussian noise models can be described as follows:

$$f_i\left(\boldsymbol{x}_i^k\right) = \exp\left\{-\frac{1}{2}\left\|\mathrm{err}_i\left(\boldsymbol{x}_i^k, \boldsymbol{z}^k\right)\right\|_{\Sigma_i}^2\right\} \tag{9}$$

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

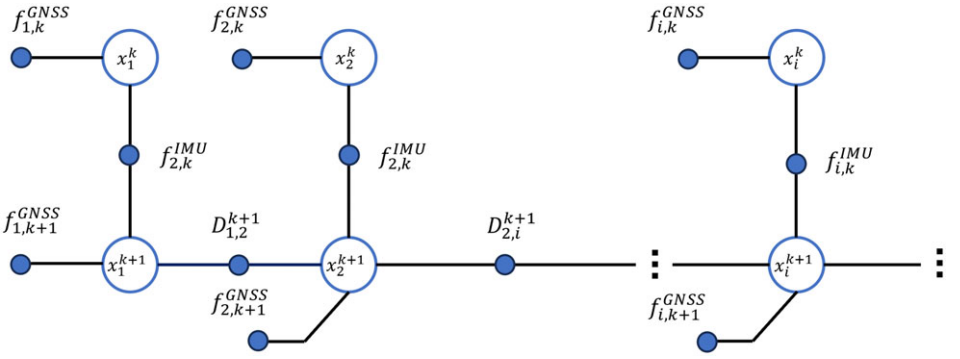


Figure 1. *Cooperative navigation factor graph of UAVs.*

where $|\,|\bullet|\,|_{\Sigma}^2$ is the square of the Mahalanobis distance and Σ_i is the sensor noise covariance matrix. Equation (9) is redefined according to the measurement model of the sensor as:

$$f_i\left(\boldsymbol{x}_i^k\right) = \exp\left\{-\frac{1}{2}\left\|h_i\left(\boldsymbol{x}_i^k\right) - \boldsymbol{z}^k\right\|_{\Sigma_i}^2\right\} \tag{10}$$

where $h(\bullet)$ is the non-linear measurement model and $\boldsymbol{z}^k$ is the actual measurement. Thus, the maximum posteriori estimation problem of Equation (5) becomes a minimising nonlinear least squares problem:

$$\boldsymbol{X}_{MAP}^k = \underset{X^k}{\operatorname{argmax}} \sum_i \left\|h_i\left(\boldsymbol{x}_i^k\right) - \boldsymbol{z}^k\right\|_{\Sigma_i}^2 \tag{11}$$

The Equation (11) is linearised using Taylor expansion and transformed into a standard least squares problem, as shown in Equation (12). The optimal estimate is then computed using the Gaussian Newton iteration method, following the specific simplification steps outlined in the literature [16].

$$\Delta^* = \underset{\Delta}{\operatorname{argmin}} \sum_i \|\boldsymbol{A}\Delta - \boldsymbol{b}\|_2^2 \tag{12}$$

2.2 Description of the sensor factor nodel

The factor node formulas for the UAV formation co-operative navigation sensor measurement model, including IMU, GNSS and UWB, are derived in this section. The derivation of the expression f_i for the factor nodes is crucial to the factor graph algorithm, as previously mentioned.

2.2.1 IMU factor node

INS is a dead reckoning system that integrates acceleration and angular velocity measurements to obtain the position, velocity, and attitude information of the carrier. The state equation of the system is

$$\boldsymbol{X} = \boldsymbol{F}\boldsymbol{X} + \boldsymbol{G}\boldsymbol{W} \tag{13}$$

where $\boldsymbol{F}$ is the linearised INS error states matrix, $\boldsymbol{G}$ is the noise transfer matrix, and $\boldsymbol{W}$ is the system process noise, which is determined by the parameters of the accelerometer and gyro and has zero mean multivariate normal distribution with variance $\boldsymbol{Q}$.

This paper uses pre-integration method to synthesise IMU measurement data within one update cycle. The calculation formula for the incremental position, velocity and attitude obtained by pre-integrating IMU measurement data within an update cycle t_k to t_{k+1} under the load system at time t is:

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

$$\begin{cases} \Delta \boldsymbol{p}_{t_{k+1}}^{t_k} & = \int \int_{t_k}^{t_{k+1}} \left(\boldsymbol{C}_{bt}^{bt_k} \left(\boldsymbol{f}_t^b - \nabla_f \right) \right) \mathrm{d}t^2 \\ \Delta \boldsymbol{v}_{t_{k+1}}^{t_k} & = \int_{t_k}^{t_{k+1}} \left(\boldsymbol{C}_{bt}^{bt_k} \left(\boldsymbol{f}_t^b - \nabla_f \right) \right) \mathrm{d}t \\ \Delta \boldsymbol{\phi}_{t_{k+1}}^{t_k} & = \int_{t_k}^{t_{k+1}} \left(\boldsymbol{E}_{bt}^{bt_k} \left(\boldsymbol{\omega}_t^b - \nabla_\omega \right) \right) \mathrm{d}t \end{cases} \tag{14}$$

where $\boldsymbol{C}_{bt}^{bt_k}$ and $\boldsymbol{E}_{bt}^{bt_k}$ are the rotation matrix and rotation rate matrix from the current carrier coordinate system to the t_k carrier coordinate system, respectively; $\boldsymbol{f}_t^b$ and $\boldsymbol{\omega}_t^b$ are the measured values of the accelerometer and gyroscope, respectively; ∇_f and ∇_ω are the deviation variables of the accelerometer and gyroscope, respectively.

According to the principle of INS dead reckoning, the position, velocity and attitude of the carrier at time t_{k+1} can be further obtained:

$$\begin{cases} \boldsymbol{p}_{t_{k+1}}^n & = \boldsymbol{p}_{t_k}^n + (t_{k+1} - t_k)\, \boldsymbol{v}_{t_k}^n + \frac{1}{2}(t_{k+1} - t_k)^2 \boldsymbol{g}^n + \boldsymbol{C}_{bt_k}^n \Delta \boldsymbol{p}_{t_{k+1}}^{t_k} \\ \boldsymbol{v}_{t_{k+1}}^n & = \boldsymbol{v}_{t_k}^n + (t_{k+1} - t_k)\, \boldsymbol{g}^n + \boldsymbol{C}_{bt_k}^n \Delta \boldsymbol{v}_{t_{k+1}}^{t_k} \\ \boldsymbol{\phi}_{t_{k+1}}^n & = \boldsymbol{\phi}_{t_k}^n + \boldsymbol{C}_{bt_k}^n \Delta \boldsymbol{\phi}_{t_{k+1}}^{t_k} \end{cases} \tag{15}$$

where $\boldsymbol{g}^n$ is gravity vector.

At moment t_k, the UAV k receives measurement information $\boldsymbol{z}_{k,t_k}^{IMU} \triangleq \left\{ \omega^b, \boldsymbol{f}^b \right\}$ from the IMU, which defines the IMU factor node as $f_{k,t_k}^{IMU}\left(\boldsymbol{x}_k^{t_k} \right)$. An IMU factor node connects two variable nodes, $\boldsymbol{x}_k^{t_k}$ and $\boldsymbol{x}_k^{t_{k+1}}$, at neighbouring moments t and t_{k+1}. The temporal update of the navigation state can be expressed as $\boldsymbol{x}_k^{t_{k+1}} = h\left(\boldsymbol{x}_k^{t_k}, \boldsymbol{z}_{k,t_k}^{IMU} \right)$ in its discretised form. The predicted state $\hat{\boldsymbol{x}}_k^{t_{k+1}}$ at the next moment is based on the IMU measurement information $\boldsymbol{z}_{k,t_k}^{IMU}$ and the current estimated state $\boldsymbol{x}_k^{t_k}$. According to Equation (15), it can be expressed as

$$\hat{\boldsymbol{x}}_k^{t_{k+1}} = h\left(\boldsymbol{x}_k^{t_k}, \boldsymbol{\alpha}_k^{t_k}, z_{k,t_k}^{IMU} \right) \tag{16}$$

where $\boldsymbol{\alpha}_k^{t_k}$ is inertial device deviation variable.

The error function of the IMU factor node is represented by the difference between the predicted and estimated values.

$$f_{k,t_k}^{IMU}\left(\boldsymbol{x}_k^{t_{k+1}}, \boldsymbol{x}_k^{t_k} \right) = d\left(\hat{\boldsymbol{x}}_k^{t_{k+1}} - \boldsymbol{x}_k^{t_{k+1}} \right) \tag{17}$$

The IMU measurement data is used to create IMU factor nodes in a single update cycle, which generates the next variable node. The update frequency of the IMU factor corresponds to the frequency at which the system outputs the estimated navigation state quantity. The outputted navigation state quantity is the target variable estimated by the system. Creating a variable node in the factor graph model is essentially the process of outputting the target variable once.

2.2.2 GNSS factor node

The general GNSS measurement equation can be expressed as:

$$\boldsymbol{z}_k^{GNSS} = h^{GNSS}\left(\boldsymbol{x}_k \right) + \boldsymbol{n}^{GNSS} \tag{18}$$

where $\boldsymbol{n}^{GNSS}$ is the measurement noise of GNSS and $h^{GNSS}\left(\boldsymbol{x}_k \right)$ is the observation function. When UAV k receives the measurement information $\boldsymbol{z}_{k,t_k}^{GNSS}$ from GNSS at moment t_k, the factor node $f_{k,t_k}^{GNSS}\left(\boldsymbol{x}_k^{t_k} \right)$ is defined as:

$$f_{k,t_k}^{GNSS}\left(\boldsymbol{x}_k^{t_k} \right) = d\left(\boldsymbol{z}_{k,t_k}^{GNSS} - h^{GNSS}\left(\boldsymbol{x}_k^{t_k} \right) \right) \tag{19}$$

and the GNSS factor node is only related to the navigation state $\boldsymbol{x}_k^{t_k}$ at the current moment.

2.2.3 UWB factor node

The information regarding the range of UAV i and UAV j can be modelled as follows:

$$\boldsymbol{d}_{i,j}^{UWB} = \boldsymbol{d}_{i,j} + \boldsymbol{n}^{UWB} \tag{20}$$

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

where $\boldsymbol{d}_{i,j}$ is the true distance between UAV i and UAV j, and $\boldsymbol{n}^{UWB}$ is the ranging noise. And the observation model $h\left(\boldsymbol{X}_{i,j}\right)$ can be expressed as:

$$h\left(\boldsymbol{X}_{i,j}\right)=\sqrt{\boldsymbol{x}_i^T\boldsymbol{x}_j} \tag{21}$$

If the range factor of UAV k and UAV $k+1$ at moment t_k is $\boldsymbol{D}_{k,k+1}^{t_k}$, then:

$$\boldsymbol{D}_{k,k+1}^{t_K}=f_{k,k+1}^{UWB}\left(\boldsymbol{X}_{k,k+1}^{t}\right)=d\left(h\left(\boldsymbol{X}_{k,k+1}^{t_k}\right)-\boldsymbol{d}_{k,k+1}^{UWB}\right) \tag{22}$$

2.3 Design of time-varying factor graph algorithm

2.3.1 Preferential reconstruction based on ranging residuals

The process of using the factor graph algorithm to solve the navigation problem is such that the system generates a new factor in the graph for each new sensor measurement it receives. This is equivalent to adding a new row to the measurement Jacobi matrix of the linearised least squares problem, and usually when the Jacobi matrix changes it is necessary to re-decompose the new Jacobi matrix, but this becomes increasingly computationally intensive as time increases. Literature [17] and literature [18] proposed the incremental smoothing algorithm to optimise the variable nodes within the window by setting a sliding window. This method reduces the computation time to a certain extent, but in large clusters where multiple variable nodes exist at a single moment, the method does not effectively reduce the computation amount, so in this paper, we use a time-varying factor graph model and introduce a double threshold detection method and anti-differential estimation to improve the stability and reliability of the system.

Firstly, distributed optimisation of the UAV formation is required, each UAV receives the UWB range information from other UAVs and calculates the range residuals, and the range residuals that satisfy the constraints are filtered using the double threshold detection method. Each UAV processes only its own IMU and GNSS information and the UWB ranging information that satisfies the constraints at the current time, and achieves the position optimisation of the UAV swarm by message iteration. Taking UAV 1 as an example, set its own state $\boldsymbol{x}_1^k$ and the UAV states $\boldsymbol{x}_2^k, \boldsymbol{x}_3^k, \boldsymbol{x}_4^k$ that satisfy the constraints at time t_k, and establish the local state $\boldsymbol{X}_1^k=\begin{bmatrix}\boldsymbol{x}_1^k & \boldsymbol{x}_2^k & \boldsymbol{x}_3^k & \boldsymbol{x}_4^k\end{bmatrix}^{\mathrm{T}}$,the local Jacobian matrix $\boldsymbol{A}_1^k$ and the local error vector $\boldsymbol{b}_1^k$, which will lead to the equations shown below, and then we will construct the local factor graph centred on UAV 1 at the time t_k.

$$\begin{aligned}
&\Delta\boldsymbol{X}_1^k=\underset{\Delta}{\operatorname{argmin}}\sum_i\left\|\boldsymbol{A}_i\Delta_i-\boldsymbol{b}_i\right\|_2^2=\\
&\underset{\Delta}{\operatorname{argmin}}\{\left\|\Delta\boldsymbol{X}_1-\left(z_1^{IMU}-\boldsymbol{x}_1^0\right)\right\|_{\Sigma_{IMU}}^2+\\
&\left\|\Delta\boldsymbol{X}_1-\left(\boldsymbol{z}_1^{GNSS}-\boldsymbol{z}_1^0\right)\right\|_{\Sigma_{GNSS}}^2+\\
&\left\|\boldsymbol{C}_{1,2}-\left(\boldsymbol{d}_{1,2}^{UWB}-\boldsymbol{d}_{1,2}^0\right)\right\|_{\Sigma_{UWB}}^2+\\
&\left\|\boldsymbol{C}_{1,3}-\left(\boldsymbol{d}_{1,3}^{UWB}-d_{1,3}^0\right)\right\|_{\Sigma_{UWB}}^2+\\
&\left\|\boldsymbol{C}_{1,4}-\left(\boldsymbol{d}_{1,4}^{UWB}-\boldsymbol{d}_{1,4}^0\right)\right\|_{\Sigma_{UWB}}^2
\end{aligned} \tag{23}$$

If the local factor graph centred on UAV 1 satisfies the constraints at the moment t_{k+1}, the corresponding state change of the coordinated UAV is $\boldsymbol{x}_3^{k+1}, \boldsymbol{x}_5^{k+1}, \boldsymbol{x}_7^{k+1}, \boldsymbol{x}_{11}^{k+1}$, the real-time updating of the factor graph framework can be achieved by simply replacing the UWB part of Equation (23) by the ranging factor of the corresponding UAV, and the simultaneous combined with the double threshold detection method to improve the system stability. The time-varying factor graph framework is shown in Fig. 2.

2.3.2 Double threshold detection method

The single threshold setting of traditional residual cardinality detection [19] makes it difficult to compromise between the false alarm rate and the missed alarm rate, and the two cannot be balanced. Therefore,

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

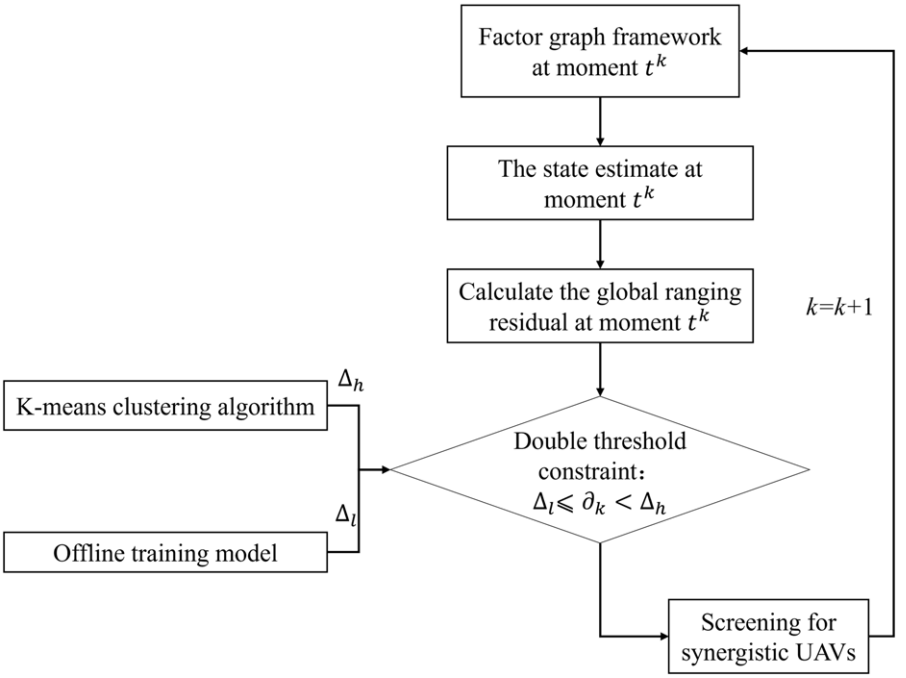


Figure 2. *Design flowchart of time-varying factor graph.*

this paper adopts a dual-threshold approach to set the size threshold for detecting whether a fault occurs or not. The assumption basis for determining whether a fault occurs in the system is:

(1) H0: $\partial_k > \Delta_h$, it is determined that the system fails;

(2) H1: $\Delta_l < \partial_k \leq \Delta_h$, it is determined that the system receives interference;

(3) H2: $\partial_k \leq \Delta_l$, that the system is functioning normally.

Δ_l and Δ_h are preset detection size thresholds, through the processing of the above conditions, to improve the detection quality of the system for small amplitude gradient faults, the main procedures are:

Step 1: Calculate and obtain the global range residual matrix $\boldsymbol{V}_t$ at the current moment t, where matrix element ∂_{ij} $(i \neq j)$ denotes the range residual between UAV i and UAV j.

Step 2: Cluster analysis of the elements in the ranging residual matrix using the K-means algorithm to obtain the upper threshold Δ_h.

Step 3: Setting up the offline training model to obtain the threshold lower limit Δ_l, which is obtained by calculating the position estimate after 50 iterations of message optimisation through the traditional distributed factor graph co-navigation framework.

Step 4: The dual-threshold method is used for fault diagnosis, and if $\Delta_l < \partial_k \leq \Delta_h$, the UWB ranging factor is adjusted using the principle of anti-differential estimation to further increase the system stability:

$$l(\partial_k) = \begin{cases} 1 & 0 \leq |\partial_k| < \Delta_l \\ \frac{\Delta_l}{|\partial_k|}\left(\frac{\Delta_h - |\partial_k|}{\Delta_h - \Delta_l}\right)^2 & \Delta_l \leq |\partial_k| < \Delta_h \\ 0 & \Delta_h \leq |\partial_k| \end{cases} \tag{24}$$

$$\boldsymbol{D}_{k,k+l}^t = \boldsymbol{f}_{k,k+1}^{UWB}\left(\boldsymbol{X}_{k,k+1}^t\right) = \boldsymbol{d}\left(l(\partial_k)\left(h\left(\boldsymbol{X}_{k,k+1}^t\right) - \boldsymbol{d}_{k,k+1}^{UWB}\right)\right) \tag{25}$$

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

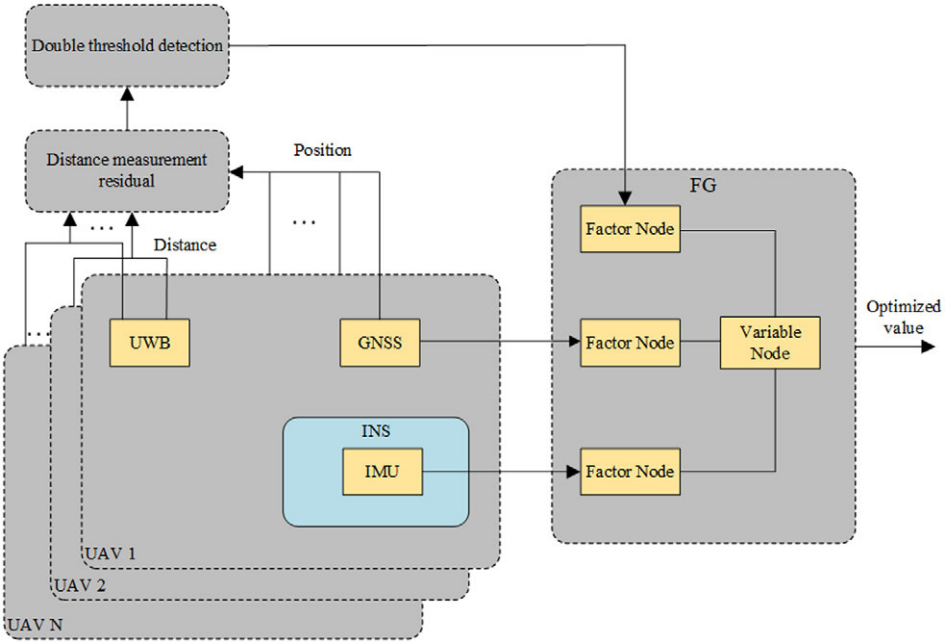


Figure 3. *Architecture for the cooperative navigation method.*

Step 5: Select the UAVs that satisfy the ranging residuals corresponding to the dual threshold detection method to construct the collaborative navigation FG framework.

2.3.3 Algorithm description

Figure 3 shows the architecture of cooperative navigation method, which can be divided into the following steps:

Step 1: All UAVs receive GNSS position information and broadcast their own position information and GNSS measurement covariance. Simultaneously calculate the relative distance, relative heading and relative pitch angle between UAVs based on the obtained position information.The specific measurement function for relative navigation information is:

$$z_k' = h'\left(X_k\right) + \boldsymbol{R}_k' \tag{26}$$

$$h'\left(\boldsymbol{X}_k\right) = \begin{bmatrix} \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2 + \left(z_i - z_j\right)^2} \\ \arctan\left[\left(y_i - y_j\right) / \left(x_i - x_j\right)\right] \\ \arctan\left[\left(z_i - z_j\right) / \sqrt{\left(x_i - x_j\right)^2 + \left(y_i - y_j\right)^2}\right] \end{bmatrix} \tag{27}$$

where x_i, y_i, z_i and x_j, y_j, z_j are the position information of UAV i and UAV j, respectively; $\boldsymbol{R}_k'$ is the relative observation noise.

Step 2: Each UAV receives distance measurements from other UAVs and collaborative information broadcasted.

Step 3: Obtain the global ranging residual matrix by calculating the position information and ranging information from Step 1 and Step 2. Construct a local factor map for each drone using the algorithms mentioned in Sections 2.3.1 and 2.3.2, and solve for the estimated state values.

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

Table 1. *Related parameters of each sensor*

Sensor	Parameter	Data
Gyroscope	Random constant/(°).h^{-1}	0.02
	White noise/(°).h^{-1}	0.02
	First-order Markov-driven white noise/(°).h^{-1}	0.02
	Update frequency/Hz	50
Accelerometer	First-order Markov-bias error/g	10^{-4}
	Update frequency/Hz	50
GPS	Position error/m	5
	Speed error/(m/s)	0.1
	Update frequency/Hz	5
UWB	Distance measurement error/m	0.5
	Update frequency/Hz	5

3.0 Result

3.1 Simulation condition setting

To verify the feasibility of the algorithm and the precision of the navigation effect accuracy, the paper conducted offline simulation of experimental data, and the sensor parameters set were referenced from the open-source navigation website (www.psins.org.cn). Table 1 lists the configuration parameters of the sensor devices carried by the master and slave drones in the formation. Each drone is equipped with IMU, GPS and UWB. The sensor noise follows a Gaussian model of $N(0, \sigma)$. The navigation information of the fleet is obtained by the sensors carried.

Simulate the motion of UAVs using MATLAB and verify the feasibility of the algorithm. The simulation conditions are as follows: the UAV formation consists of 18 UAVs, the simulation duration is 1,000s, the sampling period is 1s. During this period, each UAV obtains navigation information through absolute navigation using IMU and GPS devices. Simultaneously measure the distance of other UAVs in the local factor map and obtain the relative heading angle and relative pitch angle for collaborative navigation. The initial position and motion trajectory of the UAV swarm are shown in Figs 4 and 5. For the convenience of displaying the relative positions and distances between UAVs, Fig. 4 takes (108°, 34°, 0m) as the coordinate origin, and all the 3D scales are converted to m. To facilitate the display of the relative position and distance between drones, provide the motion trajectory of UAV 1, as shown in Fig. 6. The paper selects the simulation results of UAV 1 for analysis. Its initial position in the ENU coordinate system is (139m, 6m, 151m), and its initial velocity is 0m/s.

3.2 Simulation results and analysis

Based on the aforementioned simulation conditions, simulations of UAV formations were conducted, and the position estimation errors of the UAVs were analysed. To validate the effectiveness of the experimental results, the algorithm proposed in this paper is compared with cooperative navigation algorithms based on FKF and FG. Taking the simulation of one UAV in the cluster as an example, the simulation results are shown in Figs 7 and 8.

From Fig. 7, it can be observed that the FKF filtering algorithm requires isolation and system reconstruction of the corresponding sub-filters when a sensor failure occurs. Therefore, compared to the FG algorithm, the FKF algorithm tends to have a positioning error that is 1–3m higher. The FG algorithm, due to its plug-and-play nature, does not require adjustments for sensor issues, thus possessing strong flexibility and scalability. Its positioning error remains at around 20m. From Fig. 8, it can be seen that

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

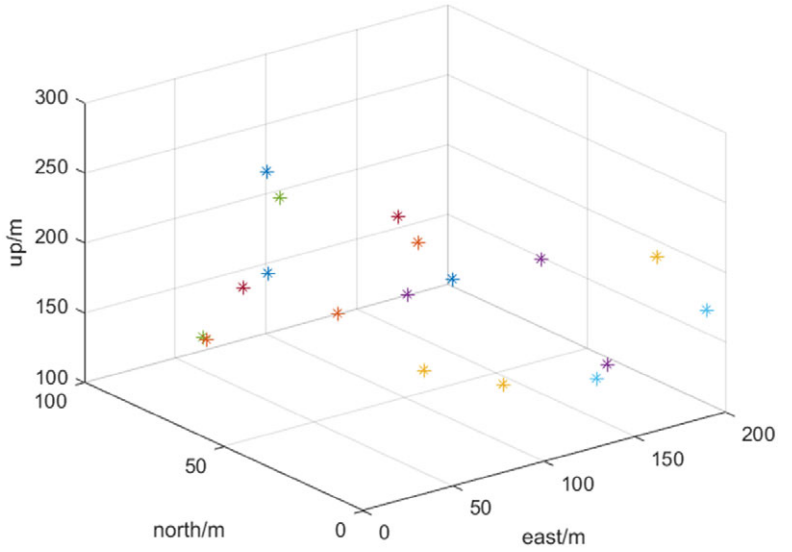


Figure 4. *Initial position coordinates of UAVs.*

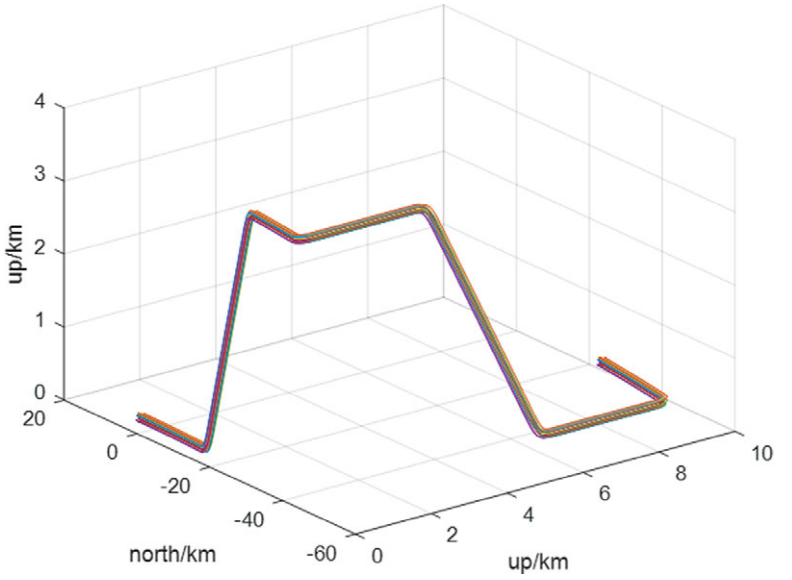


Figure 5. *The flight trajectory of UAVs.*

the directional velocity errors of the FKF and FG algorithms differ by no more than 10%. The proposed time-varying factor graph algorithm not only inherits the advantages of the FG algorithm but also optimises the factor graph framework. Its localisation performance is the best, with position errors in the northeast-up (ENU) directions generally within 12m, and velocity errors ranging from 0.5 to 1m. To quantitatively analyse the positioning errors under the three different methods, the root mean square error (RMSE) of the UAV in each direction is calculated and summarised in Table 2. The formula for calculating the RMSE is:

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

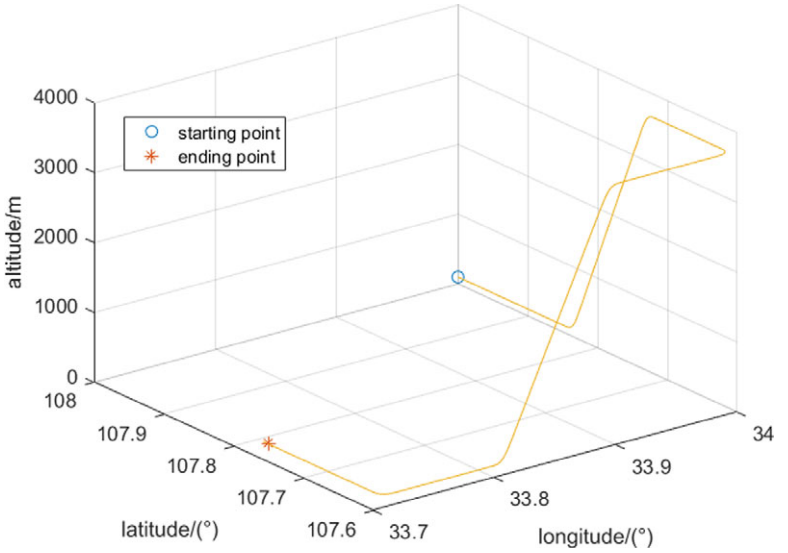


Figure 6. *The flight trajectory of UAV 1.*

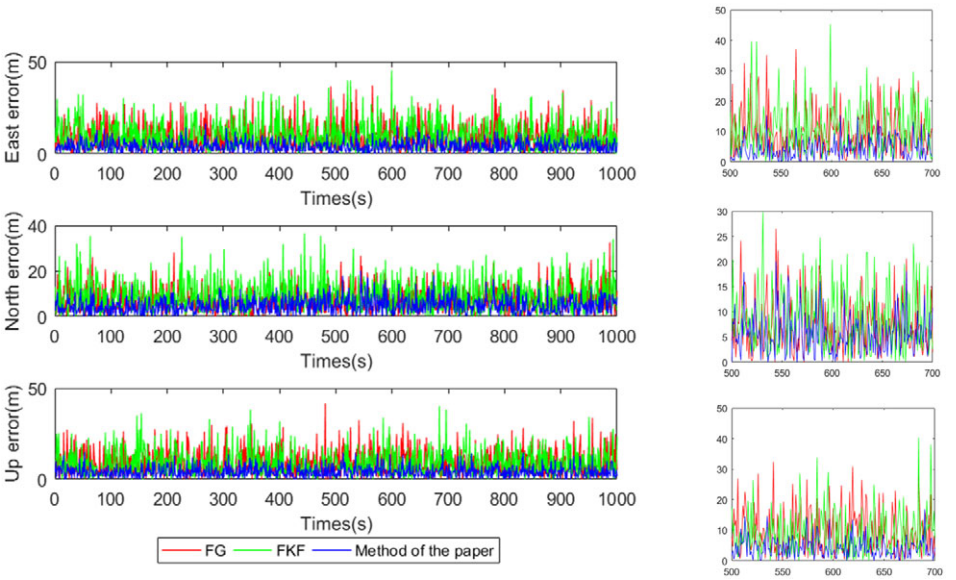


Figure 7. *Comparison of positional errors among various algorithms.*

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left[\left(\hat{\boldsymbol{X}}_k^i - \boldsymbol{X}\right)^2\right]} \tag{28}$$

From Table 2, it can be observed that both the cooperative navigation algorithms based on FKF and the ones based on factor graphs achieve positioning accuracy of around 20m. The algorithm proposed in this paper autonomously filters ranging information through preset constraints, thereby avoiding the influence of other UAVs on the positioning accuracy of the host UAV in UWB-constrained scenarios. Compared to the positioning accuracy of FKF and FG algorithms, the proposed method improves

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

Table 2. *Root mean square position error of each algorithm*

Error	RMSE/m		
	FKF	FG	Method of the paper
Eastward	12.45	11.45	5.14
Northward	11.29	8.80	5.25
Upward	11.49	11.39	6.27
Position error	20.39	18.39	9.66

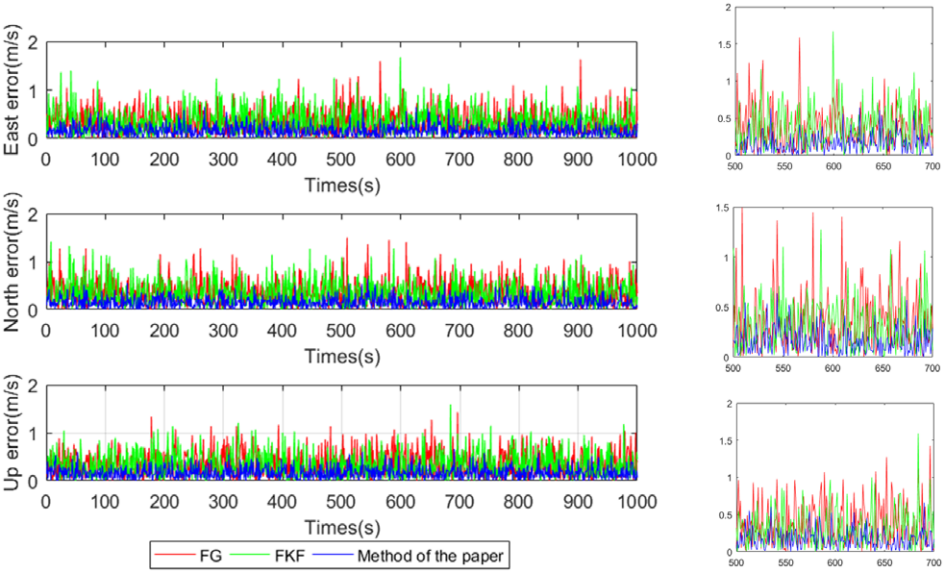


Figure 8. *Comparison of velocity errors among various algorithms.*

by 52.6% and 47.4%, respectively. To verify the real-time performance of the proposed method, the computation time for a single step is calculated as 0.066s for FG and 0.061s for the proposed method.

When executing missions in complex environments such as cities and forests, drone formations may encounter situations where satellite navigation signals are lost [20]. To further validate the robustness of the algorithm proposed in this paper, simulations were conducted in a simulated GNSS-denied environment based on the original simulation. Figures 9 and 10 respectively show the comparison curves of position error and velocity error between GNSS-denied and normal conditions. Table 3 provides statistical results of position error and velocity error in both environments.

From Fig. 9, it can be seen that the method proposed in this paper can still maintain high positioning accuracy in the GNSS-denied environment. This is because, even when GNSS positioning information is lost, drones can correct their positions through UWB cooperative information among drones, and the actual error values under denial conditions remain close to those under normal conditions. In Fig. 10, GPS can measure velocity information. Due to the loss of GNSS information, velocity errors are generally higher in denied environments, but still remain around 1.35m/s.

According to the statistical results in Table 3, it can be concluded that the method proposed in this paper enables the drone formation to maintain relatively high navigation accuracy even in GNSS-denied environments.

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

Table 3. *Root mean square position error of each algorithm*

Error	RMSE/m	
	Normal environment	GNSS-denied
The average three-dimensional position error (m)	9.66	12.32
The average three-dimensional velocity error (m/s)	0.86	1.35

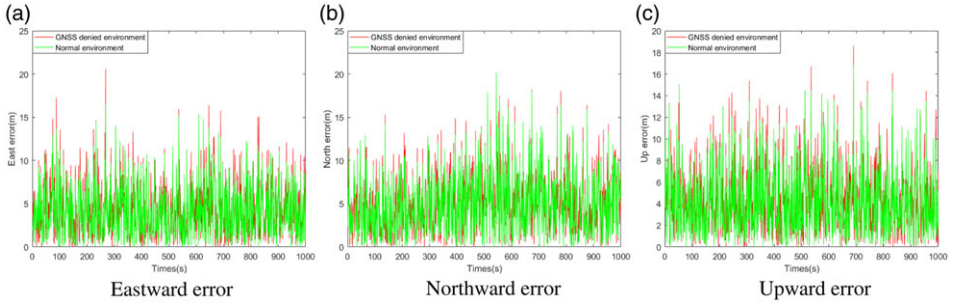


Figure 9. *Comparison of positional errors in complex environment.*

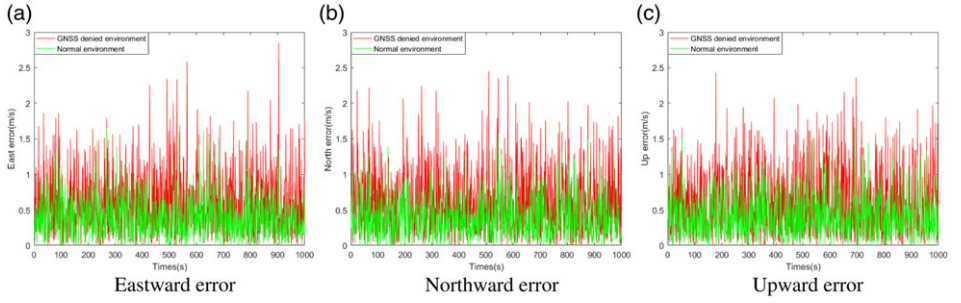


Figure 10. *Comparison of velocity errors in complex environment.*

4.0 Conclusion

This paper proposes a time-varying factor graph cooperative navigation algorithm, which utilises factors to describe the relationship between the state of the drone formation and measurements at the current time, updates the cooperative drones using a dual-threshold fault detection method, and achieves distributed optimisation of the factor graph through message iteration. Simulation results demonstrate that the proposed method can fuse drone measurement information, improve navigation accuracy and maintain high navigation accuracy even in GNSS-denied environments, verifying the robustness of the algorithm in complex environments.

Data availability statement. The data used in this study were obtained by the authors through simulation and are not publicly available.

Competing interests. The authors declare that they have no conflicts of interest related to this research.

References

[1] Zhou, X., Tang, Z., Wang, N.,Yang, C. and Huang, T. A novel state transition algorithm with adaptive fuzzy penalty for multi-constraint UAV path planning, *Expert Syst. Appl.*, 2024, **248,** p 123481.

[2] Upadhyay, J., Rawat, A. and Deb, D. Multiple drone navigation and formation using selective target tracking-based computer vision, *Electronics*, 2021, **10,** (17), p 2125.

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press

[3] Wang, S., Zhan, X., Zhai, Y., Shen, J. and Wang, H. Performance estimation for kalman filter based multi-agent cooperative navigation by employing graph theory, *Aerospace Sci. Technol.*, 2021, **112**, p 106628.
[4] Xu, L., Liu, J., Xie, L. and He, X. Multi-uav navigation and recharging for fair and sustainable coverage in wireless networks, Proceedings of the 3rd International Conference on Advanced Information Science and System, 2021, pp 1–6.
[5] Hu, G., Gao, B., Zhong, Y. and Gu, C. Unscented kalman filter with process noise covariance estimation for vehicular ins/gps integration system, *Inf. Fusion*, 2020, **64**, pp 194–204.
[6] Houzeng, H., Jian, W. and Mingyi, D. Gps/bds/ins tightly coupled integration accuracy improvement using an improved adaptive interacting multiple model with classified measurement update, *Chin. J. Aeronaut.*, 2018, **31,** (3), pp 556–566.
[7] Bodi, M., Zhenbao, L., Jiang, F., Wen, Z., Qingqing, D., Xiao, W., Zhang, J. and Lina, W. Reinforcement learning based uav formation control in gps-denied environment, *Chin. J. Aeronaut.*, 2023, **36,** (11), pp 281–296.
[8] Bai, M., Huang, Y., Zhang, Y. and Chen, F. A novel heavy-tailed mixture distribu tion based robust kalman filter for cooperative localization, *IEEE Trans. Ind. Inf.*, 2020, **17,** (5), pp 3671–3681.
[9] Wang, S., Zhan, X., Zhai, Y., Shen, J. and Wang, H. Performance estimation for kalman filter based multi-agent cooperative navigation by employing graph theory, *Aerospace Sci. Technol.*, 2021, **112**, p 106628.
[10] Xiao, X., Shi, C., Yang, Y., Liang, Y. and Guo, X. An adaptive ins/gps/vps federal Kalman filter for UAV based on SVM, 2017 13th IEEE Conference on Automation Science and Engineering (CASE), 2017, pp 1651–1656.
[11] Ben, Y., Sun, Y., Li, Q. and Zang, X. A novel cooperative navigation algorithm based on factor graph with cycles for auvs, *Ocean Eng.*, 2021, **241**, p 110024.
[12] Ma, X., Liu, X., Li, C.L. and Che, S. Multi-source information fusion based on factor graph in autonomous underwater vehicles navigation systems, *Assem. Autom.*, 2021, **41,** (5), pp 536–545.
[13] Huang, Z., Chai, H., Xiang, M., Li, D., Du, Z. and Wang, D. Multi-source information fusion localization algorithm based on auv factor graph considering information delay, *J. Chin. Inertial Technol.*, 2021, **29,** (5), pp 625–631.
[14] Du, X., Pang, X., Guan, F., Hu, J. and Zhang, W. A novel multi-source navigation algorithm based on factor graph in complex underwater environments of polar regions, *Ocean Eng.*, 2024, **301**, p 117516.
[15] Yan, Z., Luan, Z., Liu, J. and Xing, W. A cooperative localization method for mul- tiple unmanned underwater vehicles based on improved factor graph, 2023 IEEE International Conference on Mechatronics and Automation (ICMA), 2023, pp 894–899.
[16] Chen, M., Xiong, Z., Liu, J., Wang, R. and Xiong, J. Distributed cooperative navigation method for uav swarm based on factor graph, *J. Chin. Inertial Technol.*, 2020, **28,** (4), pp 456–461.
[17] Elisha, Y.B. and Indelman, V. Active online visual-inertial navigation and sensor calibration via belief space planning and factor graph based incremental smoothing, 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, pp 2616–2622.
[18] Dai, H., Bian, H., Ma, H. and Wang, R. Application of robust incremental smoothing algorithm based on factor graph in integrated navigation of unmanned surface vehicle, *J. Chin. Inertial Technol.*, 2018, **26,** (06), pp 778–786.
[19] Liu, S., Zhang, T., Zhang, J. and Zhu, Y. A new coupled method of sins/dvl integrated navigation based on improved dual adaptive factors, *IEEE Trans. Instrum. Meas.*, 2021, **70**, pp 1–11.
[20] Ouyang, X., Zeng, F., Lv, D., Dong, T. and Wang, H. Cooperative navigation of UAVs in GNSS-denied area with colored RSSI measurements, *IEEE Sens. J.*, 2020, **21,** (2), pp 2194–2210.

Cite this article: Wang H., Hu L. and Tao J. (2025). Collaborative navigation method based on adaptive time-varying factor graph. *The Aeronautical Journal*, **129**, 788–801. https://doi.org/10.1017/aer.2024.90

https://doi.org/10.1017/aer.2024.90 Published online by Cambridge University Press