SCIENCE CHINA
Information Sciences

• RESEARCH PAPER •

November 2026, Vol. 69, Iss. 11, 210303:1–210303:15
https://doi.org/10.1007/s11432-026-5188-3

Special Topic: Communication, Sensing, and Target Recognition for Low-Altitude Economy

Decentralized direct localization of low-altitude targets in clock-asynchronous multi-sensor networks

Guoxin ZHANG[1], Yunfei LIANG[1], Wei YI[1*], Hien Quoc NGO[2] & Michail MATTHAIOU[2]

[1]*School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*
[2]*Centre for Wireless Innovation, Queen's University Belfast, Belfast BT3 9DT, UK*

Received 14 March 2026/Revised 26 August 2026/Accepted 21 September 2026/Published online 26 October 2026

Abstract Accurate multi-station positioning is a key enabler for low-altitude economy applications such as unmanned aerial vehicle monitoring, aerial logistics, and safety supervision. However, low-altitude sensing networks are often bandwidth-constrained, dynamically deployed, and difficult to synchronize, which makes conventional centralized direct localization unattractive. This paper develops a lightweight decentralized direct localization framework for clock-asynchronous multi-sensor networks (MSNs). The method consists of a local-estimation stage and a fusion-update stage. In the local stage, each node refines a coarse state estimate based on its local likelihood while jointly accounting for the node-specific clock offset, where a trust-region-enhanced Gauss-Newton solver is employed to achieve efficient iterative refinement without exhaustive grid search. In the fusion stage, neighboring nodes exchange only low-dimensional intermediate messages, and a Fisher-information-guided weighted consensus together with node scheduling is used to improve global positioning accuracy and scalability. The proposed framework preserves the accuracy advantage of direct localization while substantially reducing communication overhead and removing the need for a centralized fusion center. Numerical results show that the proposed scheme achieves performance close to centralized direct localization and outperforms representative decentralized baselines under asynchronous conditions.

Keywords low-altitude target, direct localization, multi-station fusion, clock asynchrony, decentralized sensing networks

Citation Zhang G X, Liang Y F, Yi W, et al. Decentralized direct localization of low-altitude targets in clock-asynchronous multi-sensor networks. Sci China Inf Sci, 2026, 69(11): 210303, https://doi.org/10.1007/s11432-026-5188-3

1 Introduction

The rapid growth of the low-altitude economy is driving an increasing demand for reliable sensing, communication, and target-positioning capabilities in airspaces populated by unmanned aerial vehicles, logistics platforms, inspection aircraft, and other low-altitude objects. In such scenarios, multi-station perception networks are expected to provide persistent surveillance, trajectory management, safety protection, and emergency response over wide areas. Unlike conventional centralized systems, however, low-altitude sensing networks are typically composed of geographically distributed, decentralized, and resource-constrained nodes. Consequently, high-precision target positioning must be achieved under stringent communication bandwidth limitations, imperfect clock synchronization, and even the lack of a fully reliable fusion center. These challenges make multi-station cooperative localization a key enabling technology for supporting the low-altitude economy [1–5].

Target localization methods are commonly classified into two-step localization and direct localization [6–9]. In two-step approaches, each node first extracts intermediate measurements such as angle of arrival, time of arrival, or time difference of arrival, and the target position is then recovered from a geometric fusion stage [10–12]. Although computationally attractive, this architecture can be fragile in low-signal-to-noise ratio (SNR) conditions because the extracted measurements are not enforced jointly at the signal level and may not be consistently associated with the same target [13]. Direct localization estimates the target position directly from baseband observations and can therefore exploit waveform-level information more effectively, usually offering superior robustness and accuracy, especially in challenging environments [13–16]. However, most existing direct-localization methods are centralized and require all nodes to transmit raw or high-dimensional observations to a fusion center [17]. Such a design is unattractive for low-altitude sensing systems because it creates large communication overhead, poor scalability, and a single point of failure.

* Corresponding author (email: kussoyi@gmail.com)

© Science China Press 2026

Decentralized architectures provide a more suitable alternative for large-scale multi-station sensing. In a decentralized network, each node exchanges information only with neighboring nodes and performs local fusion, which improves robustness and reduces the burden on any node processor [18–24]. Existing decentralized localization studies have mainly focused on two-step methods [25–29]. Related studies have also explored distributed sensing from complementary detection and localization perspectives, including learning-assisted object localization in wireless sensor networks [30] and distributed target detection under challenging clutter, interference, and limited training data [31, 32]. These efforts further demonstrate the importance of robust inference with decentralized or resource-limited sensing nodes. More recently, several distributed direct-localization schemes have also been reported [18, 33–37]. Nevertheless, these methods still face two major obstacles in the context of low-altitude economy applications. First, many of them rely on accurate inter-node clock synchronization [18, 33–35], which is costly or impractical in fast-deployed and dynamically reconfigured sensing networks. Second, several methods still require transmitting full or truncated baseband signals, cross-ambiguity matrices, or other high-dimensional statistics among neighboring nodes [18, 33, 35], which limits scalability in bandwidth-constrained systems.

This paper develops a lightweight decentralized direct localization (DDL) framework for clock-asynchronous sensor networks. The core idea is to decompose the localization task into a local-estimation stage and a fusion-update stage. Each node first performs a local likelihood refinement around a coarse state while estimating its own clock offset, and the globally meaningful position estimate is then formed by fusing compact position-related messages from multiple nodes. In this way, the proposed design preserves the accuracy advantage of direct localization while remaining compatible with decentralized deployment.

The main contributions of this work are summarized as follows.

(1) An efficient local likelihood-refinement procedure is developed for decentralized direct localization. Using only local observations and a coarse initializer, each node iteratively refines its local position–clock state without relying on a fusion center. The local estimation (LE) output is an information contribution for subsequent multi-node fusion, rather than a standalone globally identifiable position estimate. By jointly handling position and clock offset, the proposed method avoids the exhaustive grid search required by conventional direct localization methods [13, 14, 17]. A trust-region-enhanced Gauss-Newton algorithm is further introduced to improve convergence robustness and computational efficiency.

(2) A Fisher-information-guided decentralized fusion scheme is proposed for asynchronous multi-node localization. Instead of transmitting full or truncated baseband signals [18, 33, 35], the proposed scheme performs consensus-based fusion through the exchange of low-dimensional intermediate information, thereby greatly reducing communication overhead. Fisher information is used to weight heterogeneous local information, and a node-scheduling strategy is incorporated to prioritize informative nodes and improve fusion efficiency.

(3) The proposed method is theoretically analyzed and numerically validated. We provide convergence and Cramér-Rao lower bound (CRLB) analyses, and evaluate the proposed algorithm in terms of localization accuracy, robustness, computational complexity, and communication overhead. Simulation results show that the proposed decentralized method achieves competitive performance under asynchronous conditions with significantly reduced communication cost.

The rest of the paper is organized as follows. Section 2 presents the signal model and the centralized direct-localization formulation. Section 3 introduces the proposed decentralized framework. Sections 4 and 5 develop the local-estimation and fusion-update stages, respectively. Performance analysis is given in Section 6, numerical results are reported in Section 7, and conclusions are drawn in Section 8.

Notations: Throughout this paper, a (or A), $\boldsymbol{a}$, and $\boldsymbol{A}$ represent scalars, vectors and matrices, respectively; $[\cdot]^{\top}$ denotes the transpose of its argument; $\|\cdot\|_2$ represents the 2-norm of a vector; $(\cdot)^*$ denotes the conjugate, and $(\cdot)^{\mathrm{H}}$ denotes the conjugate transpose. The superscript $(\cdot)^{R(I)}$ represents the real or imaginary parts of a complex element, the operators $\mathrm{Re}\{\cdot\}$ and $\mathrm{Im}\{\cdot\}$ extract the real and imaginary parts of their arguments, j denotes the imaginary unit, and ∇_a is the gradient with respect to a. The symbol $\propto$ denotes direct proportionality, while $\mathbb{E}\{\cdot\}$ denotes mathematical expectation. The expression $|\boldsymbol{A}|$ represents the determinant of $\boldsymbol{A}$, $\mathrm{Tr}\{\boldsymbol{A}\}$ denotes the matrix trace, and $\mathrm{Diag}(\boldsymbol{a})$ produces a diagonal matrix with the elements of $\boldsymbol{a}$.

2 Problem formulation

2.1 System configuration and signal model

Consider a single stationary low-altitude target localization problem in a multi-sensor network (MSN) consisting of L sensing nodes. The network operates in a decentralized mode, where no common fusion center (FC) is available to

centrally schedule and process information from all the nodes. Due to distance and energy consumption constraints, each node can only communicate with its neighboring nodes (one-hop routing protocol). To characterize a low-altitude aerial target, we adopt a three-dimensional (3D) geometric model. The position of the lth node is denoted by $\boldsymbol{p}_l = (x_l, y_l, z_l)^\top, l = 1, \ldots, L$, and the unknown source position is denoted by $\boldsymbol{u} = (x, y, z)^\top$. The waveform emitted by the source is assumed to be known. This assumption is reasonable in MSN localization scenarios, where the target devices are typically cooperative and transmit predefined or standardized waveforms, allowing the receiving nodes to utilize known signal structures for target localization [1]. We assume that the network has good connectivity, i.e., there exists a multi-hop communication route connecting any pair of MSN nodes in the network. In this case, the network can be represented using an undirected connected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ with the set of nodes $\mathcal{V} = \{1, 2, \ldots, L\}$ and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$. Due to the absence of global positioning system (GPS) devices or atomic clocks for time synchronization, we assume that clock offsets exist and sensors operate asynchronously. Then, under the line-of-sight propagation assumption, the baseband signal observed by the lth node can be represented as [38]

$$r_l(t) = \alpha_l s(t - \tau_l(\boldsymbol{u}) - \gamma_l) + n_l(t),\ 0 < t < T, \tag{1}$$

where T is the observation interval, α_l is an unknown complex scalar representing the path loss, and $\tau_l(\boldsymbol{u})$ represents the propagation delay from the source signal to the lth node, which can be defined as

$$\tau_l(\boldsymbol{u}) = \frac{\|\boldsymbol{p}_l - \boldsymbol{u}\|_2}{c} = \frac{\sqrt{(x - x_l)^2 + (y - y_l)^2 + (z - z_l)^2}}{c}, \tag{2}$$

where c is the speed of light. Note that the position information of the target is contained in the time delay of the signal. The term γ_l is the unknown clock offset between the local clock of the lth node and the reference clock. The source waveform $s(t)$ is common to all nodes; node-dependent attenuation and phase are represented by α_l, while the node dependence of the sampled reference waveform arises only from $\tau_l(\boldsymbol{u}) + \gamma_l$. The term $n_l(t)$ represents the noise, modeled as a complex Gaussian white random process with zero mean and variance $\sigma_{w_l}^2$, which is assumed to be mutually independent between sensors.

After sampling, the continuous signal of (1) can be written in a vector form:

$$\boldsymbol{r}_l = \alpha_l \boldsymbol{s}_l(\boldsymbol{u}, \gamma_l) + \boldsymbol{n}_l, \tag{3}$$

where the kth sample of the node-dependent reference vector is $[\boldsymbol{s}_l(\boldsymbol{u}, \gamma_l)]_k = s(kT_s - \tau_l(\boldsymbol{u}) - \gamma_l)$, $k = 0, \ldots, K-1$, and $\boldsymbol{r}_l \triangleq [r_l[0], r_l[1], \ldots, r_l[K-1]]^\top$ and $\boldsymbol{n}_l \triangleq [n_l[0], n_l[1], \ldots, n_l[K-1]]^\top$ with sampling interval $T_s = T/(K-1)$. In general, $\tau_l(\boldsymbol{u}) + \gamma_l$ is not an integer multiple of T_s; the delayed waveform is therefore evaluated by bandlimited interpolation or an equivalent fractional-delay filter. The derivatives used below are obtained by differentiating the same interpolated reference waveform with respect to the total delay. The sampled noise term $\boldsymbol{n}_l$ is still regarded as a zero-mean complex Gaussian random vector with the correlation matrix

$$\boldsymbol{N}_l = \mathbb{E}\left\{\boldsymbol{n}_l \boldsymbol{n}_l^{\mathrm{H}}\right\} = \sigma_l^2 \boldsymbol{I}_K, \tag{4}$$

where $\boldsymbol{I}_d$ denotes the $d \times d$ identity matrix. Note that, in the following, $\boldsymbol{u}$ and γ_l are omitted in $\boldsymbol{s}_l(\boldsymbol{u}, \gamma_l)$ for ease of notation, i.e., $\boldsymbol{s}_l(\boldsymbol{u}, \gamma_l)$ is written as $\boldsymbol{s}_l$.

2.2 Centralized direct localization

Before we develop the proposed method, in this section, we first review the classical centralized direct localization methods based on maximum likelihood estimation (MLE). Based on the independence of the noise distribution among various nodes, the likelihood function of multi-node joint localization can be expressed as [13, 14]

$$\zeta(\boldsymbol{R}|\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\gamma}) = \lambda \prod_{l=1}^{L} \exp\left\{-\frac{1}{2}\left(\boldsymbol{r}_l - \alpha_l \boldsymbol{s}_l\right)^{\mathrm{H}} \boldsymbol{N}_l^{-1}\left(\boldsymbol{r}_l - \alpha_l \boldsymbol{s}_l\right)\right\}, \tag{5}$$

where $\boldsymbol{R} \triangleq [\boldsymbol{r}_1, \ldots, \boldsymbol{r}_L]^\top \in \mathbb{C}^{L \times K}$, $\boldsymbol{\alpha} \triangleq [\alpha_1, \ldots, \alpha_L]^\top$, and $\boldsymbol{\gamma} \triangleq [\gamma_1, \ldots, \gamma_L]^\top$. The term λ denotes a constant coefficient unrelated to the parameters of interest $\boldsymbol{u}$ and $\boldsymbol{\gamma}$. Note that the likelihood function in (5) contains three types of unknown parameters, namely the target position $\boldsymbol{u}$, the propagation coefficient $\boldsymbol{\alpha}$ and the clock offset $\boldsymbol{\gamma}$. Let $\boldsymbol{\theta} = \left[\boldsymbol{u}^\top, \boldsymbol{\gamma}^\top, \boldsymbol{\alpha}^\top\right]^\top$ denote the unknown parameters. The centralized direct localization problem can be formulated as

$$\hat{\boldsymbol{u}} = \arg\max_{\boldsymbol{\theta}} \ln \zeta(\boldsymbol{R}|\boldsymbol{u}, \boldsymbol{\alpha}, \boldsymbol{\gamma}). \tag{6}$$

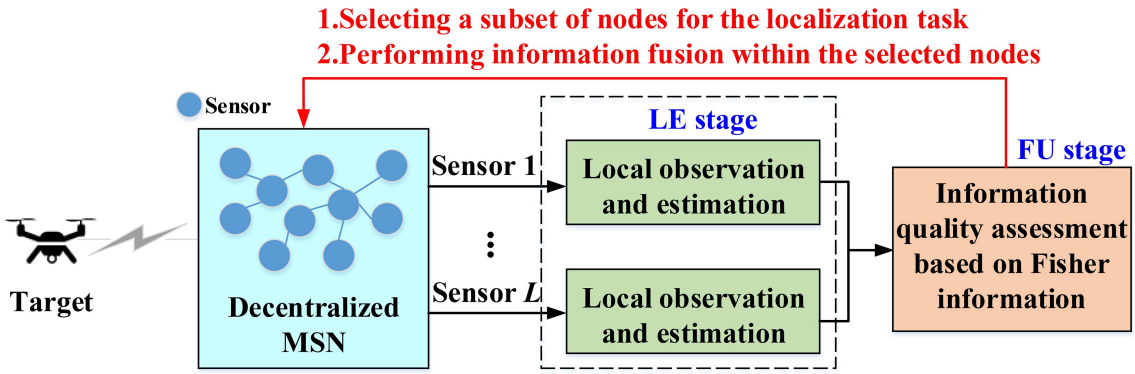

Figure 1 (Color online) Closed-loop signal processing framework for direct localization in a decentralized MSN system.

It can be seen from (6) that to perform target localization, in the centralized direct localization architecture, each node needs to transmit its received baseband signal to the FC. As indicated earlier, central computation at the FC may result in a single point of system failure in the case of FC malfunction or an adversarial attack. Furthermore, with an increase in the number of nodes, especially in large-scale systems, the dimensions of both $\boldsymbol{\alpha}$ and $\boldsymbol{\gamma}$ will also increase accordingly, leading to an unacceptable burden on communication and computation for the FC. Therefore, it is necessary to propose a decentralized architecture to address the above problem.

Remark 1. Under the direct localization framework, the impact of clock offsets on the localization performance is characterized by $\mathrm{MSE}_{localization} = \boldsymbol{\xi}\boldsymbol{\xi}^{\top} + \mathbb{E}\left\{\boldsymbol{\epsilon}\boldsymbol{\epsilon}^{\top}\right\}$, where the term $\boldsymbol{\xi}$ represents the error due to clock offsets, and the term $\boldsymbol{\epsilon}$ corresponds to the error introduced by observation noise [38]. In certain scenarios, the impact of clock offsets may even play a dominant role in the overall localization error. Therefore, it is important to emphasize that existing decentralized direct localization algorithms do not account for the effect of clock offsets, due to limitations in fusion strategies and communication bandwidth [18, 33–35].

Remark 2. To assess the communication and computational load associated with centralized direct localization, we provide an intuitive description: according to (6), the communication overhead incurred by the centralized architecture can be represented as $\mathcal{O}(LK)$. Regarding the computational complexity at FC, since the objective function is based on the joint distribution of observations at multiple nodes, the computational load of computing $\zeta(\boldsymbol{R}|\boldsymbol{u},\boldsymbol{\alpha},\boldsymbol{\gamma})$ is bounded by $\mathcal{O}(LK)$. Assuming that the parameter state space is discretized over $\Upsilon_x \times \Upsilon_y \times N_\gamma$ grids; then the total complexity of the classical centralized direct target localization is approximately given by $\mathcal{O}(LK\Upsilon_x\Upsilon_y N_\gamma)$. It can be observed that the communication load and the computational complexity of centralized direct positioning are closely related to the baseband signal and the number of nodes, and both increase substantially as these factors grow, making the system increasingly impractical.

3 Decentralized direct localization framework for low-altitude sensing

3.1 Motivation

From (6), we observe that solving the direct localization problem requires constructing the objective function $\ln\zeta(\boldsymbol{R}|\boldsymbol{u},\boldsymbol{\alpha},\boldsymbol{\gamma})$. However, its construction inherently relies on the centralized architecture. Therefore, the key to the design of decentralized direct localization needs to address two issues: (1) how each node can extract useful local likelihood information from its own asynchronous observation; and (2) how to exploit neighboring-node communication to combine these local contributions and approach centralized localization performance. To this end, we propose a DDL method, with its core architecture comprising the following two stages: the LE stage and the fusion update (FU) stage. The LE stage performs regularized local likelihood refinement around a coarse state and generates compact local update information. Because a single observation depends on $\boldsymbol{u}$ and γ_l mainly through the composite delay $\tau_l(\boldsymbol{u}) + \gamma_l$, the LE result is not interpreted as a standalone globally identifiable 3D position. The FU stage combines complementary geometry from multiple nodes to obtain the network-level position estimate. The specific signal processing flow is illustrated in Figure 1.

3.2 LE stage: construction of the local position estimator

In this subsection, we primarily address the first question mentioned in Subsection 3.1. The task of the LE stage is to construct a local likelihood refinement and its associated update information from each node's own observation.

It is not intended to make the single-node position-clock problem globally identifiable.

For a given target position $\boldsymbol{u}$, the observation models at sensor l under hypothesis H_1 (the target is present) and H_0 (the target is absent) are formulated as

$$\begin{aligned} H_1 &: \boldsymbol{r}_l = \alpha_l \boldsymbol{s}_l + \boldsymbol{n}_l, \\ H_0 &: \boldsymbol{r}_l = \boldsymbol{n}_l. \end{aligned} \tag{7}$$

The likelihood function of node observations under both hypotheses is then given by

$$\begin{aligned} p(\boldsymbol{r}_l|\boldsymbol{u}, H_1) &= \mu_1 \exp\left\{ -\frac{1}{2} \left(\boldsymbol{r}_l - \alpha_l \boldsymbol{s}_l\right)^{\mathrm{H}} \boldsymbol{N}_l^{-1} \left(\boldsymbol{r}_l - \alpha_l \boldsymbol{s}_l\right) \right\}, \\ p(\boldsymbol{r}_l|H_0) &= \mu_0 \exp\left\{ -\frac{1}{2} \boldsymbol{r}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{r}_l \right\}, \end{aligned} \tag{8}$$

where μ_1, μ_0 denote constant coefficients unrelated to the parameters $\boldsymbol{u}$ under hypotheses H_1, and H_0 respectively. Since $p(\boldsymbol{r}_l|H_0)$ is not a function of $\boldsymbol{u}$, for the estimation of $\boldsymbol{u}$, the likelihood function is equivalent to the log-likelihood ratio:

$$\ell(\boldsymbol{r}_l|\boldsymbol{u}, \alpha_l, \gamma_l) \propto \ln \frac{p(\boldsymbol{r}_l|\boldsymbol{u}, \alpha_l, \gamma_l, H_1)}{p(\boldsymbol{r}_l|H_0)} = \frac{1}{2} \left\{ \boldsymbol{r}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l \alpha_l + \left(\alpha_l \boldsymbol{s}_l\right)^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{r}_l - \left(\alpha_l \boldsymbol{s}_l\right)^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l \alpha_l \right\}. \tag{9}$$

For any parameter $\boldsymbol{u}$, the likelihood function (9) is maximized using $\alpha_l = \hat{\alpha}_l$. Since α_l is complex-valued while $\ell(\boldsymbol{r}_l|\boldsymbol{u}, \alpha_l, \gamma_l)$ is real-valued, we use Wirtinger calculus by treating α_l and α_l^* as formally independent variables. The stationary condition is therefore written as

$$\left. \frac{\partial \ell(\boldsymbol{r}_l|\boldsymbol{u}, \alpha_l, \gamma_l)}{\partial \alpha_l^*} \right|_{\alpha_l = \hat{\alpha}_l} = 0. \tag{10}$$

Substituting (9) into (10), the MLE of α_l can thus be calculated as

$$\hat{\alpha}_l = \frac{\boldsymbol{s}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{r}_l}{\boldsymbol{s}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l}. \tag{11}$$

In order to obtain the likelihood of the lth sensor node without α_l, we insert (11) into (9), and we get

$$\ell(\boldsymbol{r}_l|\boldsymbol{u}, \gamma_l) = \frac{1}{2} \boldsymbol{r}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l \left(\boldsymbol{s}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l \right)^{-1} \boldsymbol{s}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{r}_l. \tag{12}$$

At this point, we have obtained the local likelihood function at each sensor without requiring measurements from other nodes. Conditioned on signal presence, the local refinement problem is written as

$$\{\hat{\boldsymbol{u}}_l, \hat{\gamma}_l\} = \arg \max_{\{\boldsymbol{u}, \gamma_l\}} \ell(\boldsymbol{r}_l|\boldsymbol{u}, \gamma_l). \tag{13}$$

If a localization system additionally requires local signal screening, the maximized value may be compared with a threshold κ before the LE update is admitted to fusion. This optional gate is not a complete joint detection-and-localization formulation and is not part of the localization performance analysis below.

A single node observes position and clock offset primarily through the composite delay $\tau_l(\boldsymbol{u}) + \gamma_l$. Consequently, one local observation generally does not uniquely identify a 3D position together with an unknown clock offset. Eq. (13) should therefore be understood as a local likelihood refinement around the coarse initializer introduced in Section 4. The complementary propagation geometries of multiple nodes are exploited only in the FU stage to obtain the network-level position estimate.

3.3 FU stage: decentralized information fusion

One potentially intuitive solution is to utilize likelihood consensus [39]. Likelihood consensus or a distributed particle filter aims to obtain $\zeta(\mathbf{r}|\boldsymbol{u}, \boldsymbol{\gamma})$ by exchanging particles and their corresponding weights between neighboring nodes. For the direct localization problem addressed in this paper, the particles should be sampled in an extended

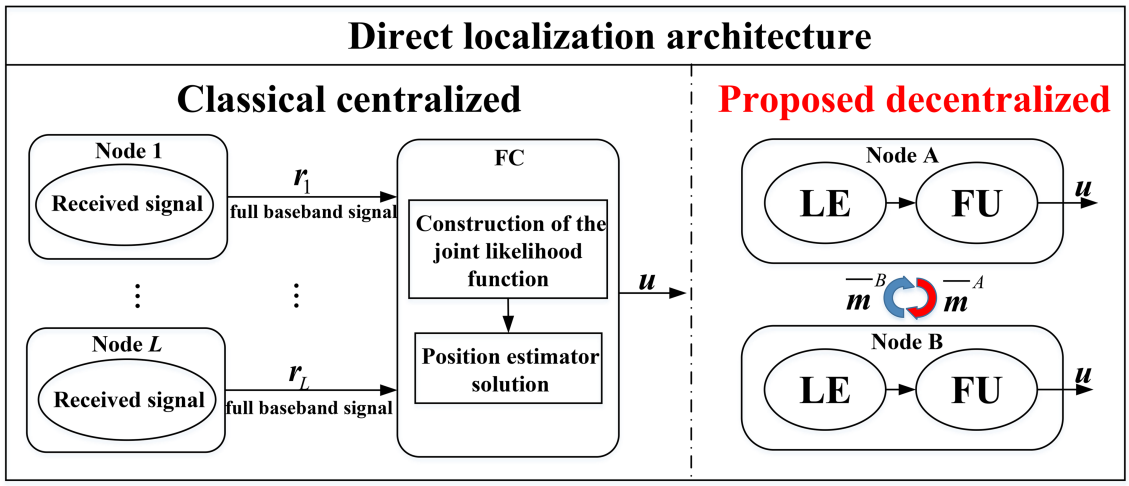

Figure 2 (Color online) Information flow in decentralized direct localization and traditional centralized direct localization frameworks, where nodes A and B are activated and are neighboring nodes. The blue arrow indicates information transfer from Node B to Node A. The red arrow indicates information transfer from Node A to Node B.

space, i.e., $\boldsymbol{z} = \left[\boldsymbol{u}^\top, \boldsymbol{\gamma}^\top\right]^\top$. Then, the local likelihood function of the lth node can be represented by O particles $\{\boldsymbol{z}^o, d_l^o\}_{o=1}^O$ as follows:

$$\zeta^{(l)}\left(\boldsymbol{r}_l|\boldsymbol{u}, \alpha_l, \gamma_l\right) \approx \sum_{o=1}^{O} d_l^o \zeta^{(l)}\left(\boldsymbol{r}_l|\boldsymbol{z}^o\right). \tag{14}$$

By exchanging and updating the local weights of particles with neighboring nodes, likelihood consensus can be achieved after several rounds of iterations, allowing each local node to approximate the global likelihood function. However, the implementation of the consensus process described above requires exchanging a large number of particles and their weights between nodes in the system, which is quite costly in terms of power and bandwidth.

We aim to develop a new, efficient information fusion strategy that must meet the following two specific requirements.

(1) Data flow. The exchanged information is generated individually by each MSN node during the LE stage, while ensuring low communication overhead and computational efficiency.

(2) Decentralized fusion. The information fusion process is based on the decentralized MSN system topology $\mathcal{G}$. The nodes reach a consensus on the direct localization solution by iteratively exchanging low-dimensional messages from the LE stage with their neighboring nodes. Additionally, in each task, only a subset of nodes that meet the requirements is activated, while the remaining nodes can remain inactive or perform other tasks.

Let Θ be the set comprising all nodes in the MSN system. Define the subset of nodes obtained through a specific selection strategy as Ψ, where $\Psi \subseteq \Theta$. In this case, the proposed fusion strategy can be expressed as

$$\hat{\boldsymbol{u}} \approx \sum_{l\in\Psi} \bar{d}_l \bar{\boldsymbol{m}}^l, \tag{15}$$

where $\bar{\boldsymbol{m}}^l$ represents the low-dimensional message generated during the LE stage, with $\bar{d}_l$ corresponding to the associated weight. The term $\hat{\boldsymbol{u}}$ denotes the target position estimated by each local node, which is expected to be sufficiently close to the solution obtained by the centralized scheme. The comparison of information flow between the proposed decentralized direct localization and traditional centralized direct localization is illustrated in Figure 2.

4 LE stage: local position estimation under clock asynchrony

In this section, we focus on the position estimation process during the LE stage. As described in Subsection 3.3, the estimates generated during this stage form the input to the FU stage. We propose a direct localization method based on an improved Gauss-Newton (GN) algorithm, which enables each node to perform local position estimation through an iterative process without relying on the presence of an FC. Let $\boldsymbol{\beta}_l = \left[\boldsymbol{u}^\top, \gamma_l\right]^\top$. It can be observed that maximizing (13) requires a high-dimensional search in the parameter space of $\boldsymbol{\beta}_l$, which is computationally prohibitive. Next, we develop a method based on GN to solve for the unknown parameter $\boldsymbol{\beta}_l$.

Let us denote the elements in (12) as follows:

$$\Lambda_l = \boldsymbol{r}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l,\ \Xi_l = \boldsymbol{s}_l^{\mathrm{H}} \boldsymbol{N}_l^{-1} \boldsymbol{s}_l. \tag{16}$$

By observing (12), we can notice that the target position is entirely contained within Λ_l and Ξ_l. Therefore, unlike directly expanding $\ell(\boldsymbol{r}_l|\boldsymbol{\beta}_l)$ in (12) using a Taylor series, we perform a first-order Taylor expansion only for Λ_l and Ξ_l at the current estimate value $\hat{\boldsymbol{\beta}}_l^q$ to obtain their approximations. The potential benefit of doing this is that we can effectively eliminate the need for computationally expensive operations, like calculating second-order derivatives required by the traditional Newton's method. Additionally, we can avail of faster convergence performance compared to the traditional steepest descent method [40]. The objective function in (12) can thus be approximated as

$$\begin{aligned}\ell(\boldsymbol{r}_l|\boldsymbol{\beta}_l) \propto & \frac{1}{2}\left[\left(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right)^{\top} \triangle_{\hat{\boldsymbol{\beta}}_l^q} + \Lambda_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right] \\ & \times\left[\left(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Xi_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right)^{\top} \triangle_{\hat{\boldsymbol{\beta}}_l^q} + \Xi_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right]^{-1}\left[\left(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right)^{\top} \triangle_{\hat{\boldsymbol{\beta}}_l^q} + \Lambda_l\left(\hat{\boldsymbol{\beta}}_l^q\right)\right]^{\mathrm{H}},\end{aligned} \tag{17}$$

where $\triangle_{\hat{\boldsymbol{\beta}}_l^q} = \boldsymbol{\beta} - \hat{\boldsymbol{\beta}}_l^q$. We define $\Lambda_l\left(\hat{\boldsymbol{\beta}}_l^q\right)$ and $\Xi_l\left(\hat{\boldsymbol{\beta}}_l^q\right)$ by Λ_l^q and Ξ_l^q, respectively. The terms in (17) can be obtained as

$$\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q = \left[\nabla_{\hat{x}_l^q}\Lambda_l^q, \nabla_{\hat{y}_l^q}\Lambda_l^q, \nabla_{\hat{z}_l^q}\Lambda_l^q, \nabla_{\hat{\gamma}_l^q}\Lambda_l^q\right]^{\top}, \tag{18}$$

$$\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Xi_l^q = \left[\nabla_{\hat{x}_l^q}\Xi_l^q, \nabla_{\hat{y}_l^q}\Xi_l^q, \nabla_{\hat{z}_l^q}\Xi_l^q, \nabla_{\hat{\gamma}_l^q}\Xi_l^q\right]^{\top}. \tag{19}$$

Based on (2), the derivative of position in (18) and (19) can be transformed into a more intuitive expression for the derivative with respect to time delay as

$$\nabla_{\hat{x}_l^q}\Lambda_l^q = \nabla_{\tau_l^q}\Lambda_l^q a_l^q,\ \nabla_{\hat{y}_l^q}\Lambda_l^q = \nabla_{\tau_l^q}\Lambda_l^q b_l^q,\ \nabla_{\hat{z}_l^q}\Lambda_l^q = \nabla_{\tau_l^q}\Lambda_l^q c_l^q, \tag{20}$$

$$\nabla_{\hat{x}_l^q}\Xi_l^q = \nabla_{\tau_l^q}\Xi_l^q a_l^q,\ \nabla_{\hat{y}_l^q}\Xi_l^q = \nabla_{\tau_l^q}\Xi_l^q b_l^q,\ \nabla_{\hat{z}_l^q}\Xi_l^q = \nabla_{\tau_l^q}\Xi_l^q c_l^q, \tag{21}$$

where

$$a_l^q = \frac{\hat{x}_l^q - x_l}{c\left\|\hat{\boldsymbol{u}}_l^q - \boldsymbol{p}_l\right\|_2},\ b_l^q = \frac{\hat{y}_l^q - y_l}{c\left\|\hat{\boldsymbol{u}}_l^q - \boldsymbol{p}_l\right\|_2},\ c_l^q = \frac{\hat{z}_l^q - z_l}{c\left\|\hat{\boldsymbol{u}}_l^q - \boldsymbol{p}_l\right\|_2}. \tag{22}$$

The estimate of $\boldsymbol{\beta}_l$ can thus be obtained through an iterative process involving $\hat{\boldsymbol{\beta}}_l^q$, i.e.,

$$\hat{\boldsymbol{\beta}}_l^{q+1} = \hat{\boldsymbol{\beta}}_l^q + \triangle_{\hat{\boldsymbol{\beta}}_l^q}, \tag{23}$$

where $\triangle_{\hat{\boldsymbol{\beta}}_l^q}$ is the increment during each iteration. After obtaining the initial value of $\hat{\boldsymbol{\beta}}_l^q$ for the current iteration, the key question is how to select an appropriate increment $\triangle_{\hat{\boldsymbol{\beta}}_l^q}$.

The MLE of the increment between the value after an iteration and the true value of $\boldsymbol{\beta}_l$ in (17) can be solved as

$$\hat{\triangle}_{\hat{\boldsymbol{\beta}}_l^q} = -\left[\boldsymbol{M}_l^q\right]^{-1} \times \left[\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Xi_l^q \ln p\left(\boldsymbol{r}_l|\hat{\boldsymbol{\beta}}_l^q\right) - \mathrm{Re}\left\{\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q(\Lambda_l^q)^*\right\}\right] \tag{24}$$

with

$$\boldsymbol{M}_l^q = \mathrm{Re}\left\{\left(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q\right)^*\left(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q\right)^{\top}\right\}. \tag{25}$$

By substituting (24) into (23), each sensor node obtains a regularized local position-clock refinement used to form the subsequent fusion message. It is important to note that the iterative solution process in (23) may require an initial value. In this paper, considering the trade-off between communication bandwidth and computational load, we use a two-step localization method similar to that in [26] to determine the initial value for $\hat{\boldsymbol{\beta}}_l^0$.

From (24), $\boldsymbol{M}_l^q$ must be inverted to obtain the local GN increment. For the single-node model, however, $\boldsymbol{M}_l^q$ is rank deficient and $|\boldsymbol{M}_l^q| = 0$. This singularity has a direct physical interpretation: the local waveform constrains $\boldsymbol{u}$ and γ_l mainly through the composite delay $\tau_l(\boldsymbol{u}) + \gamma_l$, and therefore does not provide enough independent geometric information to identify all components of the local position-clock state. Consequently, the undamped GN update is not well defined. The Levenberg-Marquardt (LM) damping introduced below regularizes the numerical update but

does not create additional geometric information; the missing geometric constraints are supplied through multi-node fusion. In addition, excessively large GN steps may slow or destabilize convergence.

To address the aforementioned issues, we further propose a modified GN iterative method, in which the position update is adaptively adjusted based on a trust-region-inspired mechanism to enhance convergence stability and estimation robustness. Specifically, when updating the estimate, we limit the distance between the new and old estimates to a specified bound. By utilizing the unit step-size LM method [41], the incremental optimization problem can be formulated as

$$\begin{aligned} \hat{\triangle}_{\hat{\boldsymbol{\beta}}_l^q}^{LM} &= \arg\max_{\triangle_{\hat{\boldsymbol{\beta}}_l^q}} \ell(\boldsymbol{r}_l|\boldsymbol{\beta}_l), \\ \text{s.t.} \quad &\left\|\triangle_{\hat{\boldsymbol{\beta}}_l^q}\right\|_2 \leqslant \rho_{\mathrm{TR}}, \end{aligned} \tag{26}$$

where ρ_{TR} denotes the trust-region radius. By introducing the scalar $\mu_l^q \geqslant 0$, the above optimization problem can be transformed into an unconstrained optimization problem. Then, by substituting (17) into (26), the feasible solution of $\triangle_{\hat{\boldsymbol{\beta}}_l^q}$ can be obtained as

$$\hat{\triangle}_{\hat{\boldsymbol{\beta}}_l^q}^{\mathrm{LM}} = -\left[\boldsymbol{M}_l^q + \mu_l^q \boldsymbol{I}_4\right]^{-1} \left[\nabla_{\hat{\boldsymbol{\beta}}_l^q} \Xi_l^q \ln p\left(\boldsymbol{r}_l|\hat{\boldsymbol{\beta}}_l^q\right) - \mathrm{Re}\left\{\nabla_{\hat{\boldsymbol{\beta}}_l^q} \Lambda_l^q (\Lambda_l^q)^*\right\}\right], \tag{27}$$

where μ_l^q represents the damping factor.

According to (27), for larger values of μ_l^q, the term μ_l^q will play a decisive role. This indicates that $\hat{\triangle}_{\hat{\boldsymbol{\beta}}_l^q}^{\mathrm{LM}}$ approaches the iteration speed of the steepest descent algorithm. For smaller values of μ_l^q, μ_l^q can be ignored and thus $\hat{\triangle}_{\hat{\boldsymbol{\beta}}_l^q}^{\mathrm{LM}}$ approaches the iteration speed of the GN algorithm. Therefore, establishing the update rule of μ_l^q in real-time and in a decentralized manner is a problem that needs to be addressed. In traditional damped GN methods, precise line search and backtracking methods are widely used to find appropriate step sizes, but these methods rely on parameter adjustments and consume a significant amount of computational resources, leading to unsatisfactory efficiency. Moreover, due to the possibility of non-convex objective functions, searching for descent directions when encountering negative curvature is also impractical [42]. Here, we use a trust region strategy to update the parameter μ_l^q, which dynamically adjusts μ_l^q based on the estimated distance between adjacent iterations and does not depend on the convexity of the objective function. To distinguish the true local objective from its local model, define

$$m_l^q(\Delta\boldsymbol{\beta}) = \frac{1}{2}\left[(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q)^\top \Delta\boldsymbol{\beta} + \Lambda_l^q\right]\left[(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Xi_l^q)^\top \Delta\boldsymbol{\beta} + \Xi_l^q\right]^{-1}\left[(\nabla_{\hat{\boldsymbol{\beta}}_l^q}\Lambda_l^q)^\top \Delta\boldsymbol{\beta} + \Lambda_l^q\right]^{\mathrm{H}}. \tag{28}$$

For the trial LM step $\Delta\hat{\boldsymbol{\beta}}_{l,q}^{LM}$, let the actual and predicted objective increases be

$$A_l^q = \ell(\boldsymbol{r}_l|\hat{\boldsymbol{\beta}}_l^q + \Delta\hat{\boldsymbol{\beta}}_{l,q}^{\mathrm{LM}}) - \ell(\boldsymbol{r}_l|\hat{\boldsymbol{\beta}}_l^q), \quad P_l^q = m_l^q(\Delta\hat{\boldsymbol{\beta}}_{l,q}^{\mathrm{LM}}) - m_l^q(\boldsymbol{0}). \tag{29}$$

We then use the safeguarded agreement ratio:

$$\upsilon_l^q = \frac{A_l^q}{P_l^q + \epsilon_{\mathrm{TR}}}, \qquad \epsilon_{\mathrm{TR}} > 0, \tag{30}$$

and update the damping factor as

$$\mu_l^{q+1} = \begin{cases} \mu_l^q/2, & \upsilon_l^q > \upsilon^{\mathrm{high}}, \\ 2\mu_l^q, & \upsilon_l^q < \upsilon^{\mathrm{low}}, \\ \mu_l^q, & \text{otherwise}. \end{cases} \tag{31}$$

The trial step is accepted when $A_l^q > 0$. Otherwise, the state is kept at $\hat{\boldsymbol{\beta}}_l^q$ and the enlarged damping factor is used to recompute a more conservative step. The thresholds $\upsilon^{\mathrm{low}} < \upsilon^{\mathrm{high}}$ are common to all nodes.

5 FU stage: decentralized information aggregation and node scheduling

Although the LE stage provides a regularized local refinement, a single asynchronous node does not provide sufficient geometry for a globally identifiable 3D solution. The FU stage therefore combines complementary information from multiple active nodes while retaining the decentralized communication architecture.

5.1 Weighted consensus-based information fusion

The exchanged message should contain only information associated with the common target position. Node-specific clock offsets should remain local. This observation leads to the following more precise decomposition result.

Proposition 1. Suppose that the active nodes construct their local LM models around a common, or consensus-aligned, position linearization point and adopt compatible damping parameters. After eliminating the node-specific clock increments, the centralized linearized position update admits an exact additive decomposition in terms of the local reduced matrix-vector pairs. Consequently, the centralized GN/LM position update can be reconstructed through decentralized aggregation of these low-dimensional local quantities, without exchanging the original baseband observations.

Proof. Please see Appendix A for details.

At the terminal LE iterate of node l, define the full local damped system with respect to $\boldsymbol{\beta}_l = [\boldsymbol{u}^\top, \gamma_l]^\top$ as

$$\boldsymbol{D}_l = \boldsymbol{M}_l + \mu_l \boldsymbol{I}_4, \tag{32}$$

$$\boldsymbol{w}_l = \nabla_{\boldsymbol{\beta}_l} \Xi_l \ln p(\boldsymbol{r}_l|\boldsymbol{\beta}_l) - \mathrm{Re}\{\nabla_{\boldsymbol{\beta}_l} \Lambda_l \Lambda_l^*\}. \tag{33}$$

Thus, $\boldsymbol{D}_l \in \mathbb{R}^{4\times 4}$ is symmetric and $\boldsymbol{w}_l \in \mathbb{R}^4$. Since $\boldsymbol{\beta}_l \in \mathbb{R}^4$, the gradients $\nabla_{\boldsymbol{\beta}_l}\Lambda_l$ and $\nabla_{\boldsymbol{\beta}_l}\Xi_l$ are both 4×1 vectors. Partition them as

$$\boldsymbol{D}_l = \begin{bmatrix} \boldsymbol{D}_{uu,l} & \boldsymbol{d}_{u\gamma,l} \\ \boldsymbol{d}_{\gamma u,l} & d_{\gamma\gamma,l} \end{bmatrix}, \qquad \boldsymbol{w}_l = \begin{bmatrix} \boldsymbol{w}_{u,l} \\ w_{\gamma,l} \end{bmatrix}. \tag{34}$$

Because γ_l is local to node l, it is eliminated before message exchange by the Schur complement:

$$\widetilde{\boldsymbol{D}}_l = \boldsymbol{D}_{uu,l} - \boldsymbol{d}_{u\gamma,l} d_{\gamma\gamma,l}^{-1} \boldsymbol{d}_{\gamma u,l}, \tag{35}$$

$$\widetilde{\boldsymbol{w}}_l = \boldsymbol{w}_{u,l} - \boldsymbol{d}_{u\gamma,l} d_{\gamma\gamma,l}^{-1} w_{\gamma,l}. \tag{36}$$

Hence, only $\widetilde{\boldsymbol{D}}_l \in \mathbb{R}^{3\times 3}$ and $\widetilde{\boldsymbol{w}}_l \in \mathbb{R}^3$ are involved in consensus.

Another key issue is how to fuse the reduced messages from neighboring sensors. We use a weighted consensus strategy [24,27,28]. Here, q indexes the local LE iterations, whereas h indexes the subsequent FU consensus rounds. Each active node first completes its local LM refinement up to its stopping criterion, uses its terminal LE state to construct $(\widetilde{\boldsymbol{D}}_l, \widetilde{\boldsymbol{w}}_l)$, and then performs H neighbor-to-neighbor consensus rounds without recomputing the local likelihood. Thus, the two indices describe sequential stages rather than nested iterations.

Let Φ_l include node l and its active one-hop neighbors. Initialize $\bar{\boldsymbol{D}}_l^0 = \widetilde{\boldsymbol{D}}_l$ and $\bar{\boldsymbol{w}}_l^0 = \widetilde{\boldsymbol{w}}_l$. For $h = 0, \ldots, H-1$,

$$\bar{\boldsymbol{D}}_l^{h+1} = \sum_{g\in\Phi_l} \eta_{lg}^h \bar{\boldsymbol{D}}_g^h, \qquad \bar{\boldsymbol{w}}_l^{h+1} = \sum_{g\in\Phi_l} \eta_{lg}^h \bar{\boldsymbol{w}}_g^h, \quad \sum_{g\in\Phi_l} \eta_{lg}^h = 1. \tag{37}$$

The fused position update is

$$\hat{\boldsymbol{u}}_l^{\mathrm{FU}} = \hat{\boldsymbol{u}}_l^{\mathrm{LE}} - (\bar{\boldsymbol{D}}_l^{\mathrm{H}})^{-1} \bar{\boldsymbol{w}}_l^{\mathrm{H}}. \tag{38}$$

The node-specific clock offset is then retained locally and can be back-substituted as $\Delta\hat{\gamma}_l = -d_{\gamma\gamma,l}^{-1}(w_{\gamma,l} + \boldsymbol{d}_{\gamma u,l}\Delta\hat{\boldsymbol{u}}_l)$. No clock-offset component is averaged across different nodes.

To avoid mixing parameters with different physical units, we do not use the trace of the complete FIM as a message-quality score. Instead, define the position-information score of node g as

$$I_g \triangleq \mathrm{Tr}\{\boldsymbol{J}_{uu}^{(g)}\}, \qquad \boldsymbol{J}_{uu}^{(g)} = \frac{2|\alpha_g|^2}{\sigma_g^2} \|\nabla_{\tau_g} \boldsymbol{s}_g\|_2^2 \begin{bmatrix} a_g^2 & a_g b_g & a_g c_g \\ a_g b_g & b_g^2 & b_g c_g \\ a_g c_g & b_g c_g & c_g^2 \end{bmatrix}. \tag{39}$$

All entries of $\boldsymbol{J}_{uu}^{(g)}$ refer to the same position parameter and therefore have a consistent physical scaling. With a small $\epsilon_J > 0$ ensuring a positive self-weight, the receiver-specific consensus weights are

$$\eta_{lg}^h = \frac{I_g + \epsilon_J}{\sum_{m\in\Phi_l}(I_m + \epsilon_J)}, \qquad g \in \Phi_l. \tag{40}$$

This is a heuristic position-sensitivity score for weighting and scheduling; the full nuisance-parameter-aware position CRLB is derived separately in Section 6.

5.2 Node scheduling

Eqs. (37) and (38) require only low-dimensional neighbor messages. To reduce the number of active sensors, we further use a Fisher-information-guided scheduling procedure. Under independent sensor observations, the global log-likelihood is the sum of local log-likelihoods. Hence, the common-position FIM block is additive,

$$\boldsymbol{J}_{uu}^{(1:l)} = \sum_{i=1}^{l} \boldsymbol{J}_{uu}^{(i)} = \boldsymbol{J}_{uu}^{(1:l-1)} + \boldsymbol{J}_{uu}^{(l)}, \tag{41}$$

where the node-specific nuisance parameters $\{\gamma_i, \alpha_i\}$ occupy distinct blocks in the full global FIM. This score additivity follows from the local *log*-likelihood scores, rather than derivatives of the likelihood itself.

For reproducibility, the scheduling procedure is specified in three steps. (1) Initialization: let l_0 be the task-initiating node, ρ_{FI} the scheduling threshold, Ψ the selected set, $\mathcal{V}_{vis}$ the visited set, and I_{acc} the accumulated position score; set $\Psi = \mathcal{V}_{vis} = \{l_0\}$ and $I_{acc} = I_{l_0}$. (2) Selection: while $I_{acc} < \rho_{\mathrm{FI}}$, form $\mathcal{C}_l = \aleph_l \setminus \mathcal{V}_{vis}$ and, if it is nonempty, select

$$m^\star = \arg\max_{m \in \mathcal{C}_l} I_m, \tag{42}$$

add $m^\star$ to Ψ and $\mathcal{V}_{vis}$, update $I_{acc} \leftarrow I_{acc} + I_{m^\star}$, and continue from $m^\star$; ties are resolved by node index (or another fixed network priority). (3) Backtracking and termination: if the current node has no unvisited neighbor, backtrack to the most recent selected node with one; if all reachable nodes are visited before the threshold is met, retain all reachable selected nodes and return an unmet-threshold flag. The notation ρ_{FI} is deliberately distinct from the trust-region radius ρ_{TR}.

At this point, the overall DDL procedure is complete: node scheduling determines the active subset, each active node performs the LE refinement, and the resulting reduced position messages are combined through the FU weighted consensus.

6 Performance analysis

In this section, we analyze the performance of the proposed DDL algorithm from the following aspects: convergence, CRLB, comparison of computational complexity with related methods, and communication overhead. For comparison, we introduce three related baselines adapted from existing decentralized localization methods. Specifically, the method in [36] is extended to the considered asynchronous setting by treating the clock offset as a latent nuisance variable and embedding it into the iterative refinement, and the resulting baseline is referred to as Async EM-DPD. The diffusion-based distributed localization method in [37] is adapted by incorporating timing uncertainty into its local update while preserving the original neighbor-to-neighbor diffusion mechanism, and is denoted by Async diffusion-DDPD. In addition, the distributed localization method in [29] is simplified into a timing-only asynchronous version by discarding angle processing and incorporating timing mismatch into the local two-step estimation, which is denoted by Async TOA-WLS. These three methods, together with the proposed DDL, are used as related benchmarks in the following complexity and simulation comparisons.

6.1 Convergence analysis

The LE and FU stages are sequential. The local trust-region GN/LM iteration is locally convergent under the usual regularity and initialization conditions [40, 41]; after LE terminates, the FU stage runs a separate weighted-consensus iteration. Let $\boldsymbol{W}^h = [\eta_{lg}^h]$ denote the global consensus weight matrix, with $\eta_{lg}^h = 0$ for $g \notin \Phi_l$. From (40), $\boldsymbol{W}^h \mathbf{1} = \mathbf{1}$, so the matrix is row stochastic, but it is generally not doubly stochastic on an irregular graph.

Lemma 1. If the active-node graph is connected (or jointly connected over finite intervals), every node has a positive self-weight, and all nonzero weights are uniformly bounded away from zero, the row-stochastic consensus iteration in (37) converges to a common weighted value [28, 42]. The limit is not necessarily the arithmetic average unless the weight matrix is additionally doubly stochastic.

Thus, the proposed FU should be interpreted as information-weighted consensus. This convergence statement concerns the consensus messages; because the underlying localization objective is non-convex, it does not imply global optimality of the original nonlinear estimation problem.

6.2 CRLB analysis

The CRLB provides a lower bound on the accuracy of any unbiased estimator. For the asynchronous direct localization model considered in this paper, define the complete unknown parameter vector as

$$\boldsymbol{\vartheta} = \left[\boldsymbol{u}^\top, \boldsymbol{\psi}^\top\right]^\top, \qquad \boldsymbol{\psi} = \left[\boldsymbol{\gamma}^\top, (\boldsymbol{\alpha}^R)^\top, (\boldsymbol{\alpha}^I)^\top\right]^\top, \tag{43}$$

where $\boldsymbol{\gamma} = [\gamma_1, \ldots, \gamma_L]^\top$, $\boldsymbol{\alpha}^R = [\alpha_1^R, \ldots, \alpha_L^R]^\top$, and $\boldsymbol{\alpha}^I = [\alpha_1^I, \ldots, \alpha_L^I]^\top$. Then, the CRLB of the target position is

$$\mathrm{CRLB}(\boldsymbol{u}) = \left[\boldsymbol{J}_{\boldsymbol{uu}} - \boldsymbol{J}_{\boldsymbol{u\psi}}\boldsymbol{J}_{\boldsymbol{\psi\psi}}^{-1}\boldsymbol{J}_{\boldsymbol{\psi u}}\right]^{-1}. \tag{44}$$

For notational simplicity, define

$$\dot{\boldsymbol{s}}_{\tau,l} \triangleq \nabla_{\tau_l}\boldsymbol{s}_l, \qquad \dot{\boldsymbol{s}}_{\gamma,l} \triangleq \nabla_{\gamma_l}\boldsymbol{s}_l, \qquad a_l = \frac{x - x_l}{c\|\boldsymbol{u} - \boldsymbol{p}_l\|_2}, \qquad b_l = \frac{y - y_l}{c\|\boldsymbol{u} - \boldsymbol{p}_l\|_2}, \qquad c_l = \frac{z - z_l}{c\|\boldsymbol{u} - \boldsymbol{p}_l\|_2}. \tag{45}$$

The corresponding FIM submatrices are given by

$$\boldsymbol{J}_{\boldsymbol{uu}} = \sum_{l=1}^{L} \frac{2|\alpha_l|^2}{\sigma_l^2}\|\dot{\boldsymbol{s}}_{\tau,l}\|_2^2 \begin{bmatrix} a_l^2 & a_l b_l & a_l c_l \\ a_l b_l & b_l^2 & b_l c_l \\ a_l c_l & b_l c_l & c_l^2 \end{bmatrix}, \qquad \boldsymbol{J}_{\boldsymbol{u\gamma}} = \begin{bmatrix} J_{x\gamma_1} & \cdots & J_{x\gamma_L} \\ J_{y\gamma_1} & \cdots & J_{y\gamma_L} \\ J_{z\gamma_1} & \cdots & J_{z\gamma_L} \end{bmatrix}, \qquad \boldsymbol{J}_{\boldsymbol{\gamma u}} = \boldsymbol{J}_{\boldsymbol{u\gamma}}^\top, \tag{46}$$

where $J_{x\gamma_l} = \frac{2|\alpha_l|^2 a_l}{\sigma_l^2}\mathrm{Re}\left\{\dot{\boldsymbol{s}}_{\tau,l}^{\mathrm{H}}\dot{\boldsymbol{s}}_{\gamma,l}\right\}$, $J_{y\gamma_l} = \frac{2|\alpha_l|^2 b_l}{\sigma_l^2}\mathrm{Re}\left\{\dot{\boldsymbol{s}}_{\tau,l}^{\mathrm{H}}\dot{\boldsymbol{s}}_{\gamma,l}\right\}$, and $J_{z\gamma_l} = \frac{2|\alpha_l|^2 c_l}{\sigma_l^2}\mathrm{Re}\left\{\dot{\boldsymbol{s}}_{\tau,l}^{\mathrm{H}}\dot{\boldsymbol{s}}_{\gamma,l}\right\}$.

Let $A_l \triangleq \alpha_l^*\dot{\boldsymbol{s}}_{\tau,l}^{\mathrm{H}}\boldsymbol{s}_l$, $C_l \triangleq \alpha_l^*\dot{\boldsymbol{s}}_{\gamma,l}^{\mathrm{H}}\boldsymbol{s}_l$. Then, we have $\boldsymbol{J}_{\boldsymbol{u\alpha}} = [\boldsymbol{J}_{\boldsymbol{u\alpha}^R}, \boldsymbol{J}_{\boldsymbol{u\alpha}^I}]$, $\boldsymbol{J}_{\boldsymbol{\alpha u}} = \boldsymbol{J}_{\boldsymbol{u\alpha}}^\top$, with

$$\boldsymbol{J}_{\boldsymbol{u\alpha}^R} = \begin{bmatrix} \frac{2a_1}{\sigma_1^2}\mathrm{Re}\{A_1\} & \cdots & \frac{2a_L}{\sigma_L^2}\mathrm{Re}\{A_L\} \\ \frac{2b_1}{\sigma_1^2}\mathrm{Re}\{A_1\} & \cdots & \frac{2b_L}{\sigma_L^2}\mathrm{Re}\{A_L\} \\ \frac{2c_1}{\sigma_1^2}\mathrm{Re}\{A_1\} & \cdots & \frac{2c_L}{\sigma_L^2}\mathrm{Re}\{A_L\} \end{bmatrix}, \qquad \boldsymbol{J}_{\boldsymbol{u\alpha}^I} = \begin{bmatrix} \frac{2a_1}{\sigma_1^2}\mathrm{Re}\{jA_1\} & \cdots & \frac{2a_L}{\sigma_L^2}\mathrm{Re}\{jA_L\} \\ \frac{2b_1}{\sigma_1^2}\mathrm{Re}\{jA_1\} & \cdots & \frac{2b_L}{\sigma_L^2}\mathrm{Re}\{jA_L\} \\ \frac{2c_1}{\sigma_1^2}\mathrm{Re}\{jA_1\} & \cdots & \frac{2c_L}{\sigma_L^2}\mathrm{Re}\{jA_L\} \end{bmatrix}. \tag{47}$$

Moreover,

$$\boldsymbol{J}_{\boldsymbol{\psi\psi}} = \begin{bmatrix} \boldsymbol{J}_{\boldsymbol{\gamma\gamma}} & \boldsymbol{J}_{\boldsymbol{\gamma\alpha}^R} & \boldsymbol{J}_{\boldsymbol{\gamma\alpha}^I} \\ \boldsymbol{J}_{\boldsymbol{\alpha}^R\boldsymbol{\gamma}} & \boldsymbol{J}_{\boldsymbol{\alpha}^R\boldsymbol{\alpha}^R} & \boldsymbol{0} \\ \boldsymbol{J}_{\boldsymbol{\alpha}^I\boldsymbol{\gamma}} & \boldsymbol{0} & \boldsymbol{J}_{\boldsymbol{\alpha}^I\boldsymbol{\alpha}^I} \end{bmatrix}, \tag{48}$$

where

$$\boldsymbol{J}_{\boldsymbol{\gamma\gamma}} = \mathrm{Diag}\left(\frac{2|\alpha_1|^2}{\sigma_1^2}\|\dot{\boldsymbol{s}}_{\gamma,1}\|_2^2, \ldots, \frac{2|\alpha_L|^2}{\sigma_L^2}\|\dot{\boldsymbol{s}}_{\gamma,L}\|_2^2\right), \tag{49}$$

$$\boldsymbol{J}_{\boldsymbol{\gamma\alpha}^R} = \mathrm{Diag}\left(\frac{2}{\sigma_1^2}\mathrm{Re}\{C_1\}, \ldots, \frac{2}{\sigma_L^2}\mathrm{Re}\{C_L\}\right), \qquad \boldsymbol{J}_{\boldsymbol{\alpha}^R\boldsymbol{\gamma}} = \boldsymbol{J}_{\boldsymbol{\gamma\alpha}^R}^\top, \tag{50}$$

$$\boldsymbol{J}_{\boldsymbol{\gamma\alpha}^I} = \mathrm{Diag}\left(\frac{2}{\sigma_1^2}\mathrm{Re}\{jC_1\}, \ldots, \frac{2}{\sigma_L^2}\mathrm{Re}\{jC_L\}\right), \qquad \boldsymbol{J}_{\boldsymbol{\alpha}^I\boldsymbol{\gamma}} = \boldsymbol{J}_{\boldsymbol{\gamma\alpha}^I}^\top, \tag{51}$$

and

$$\boldsymbol{J}_{\boldsymbol{\alpha}^R\boldsymbol{\alpha}^R} = \boldsymbol{J}_{\boldsymbol{\alpha}^I\boldsymbol{\alpha}^I} = \mathrm{Diag}\left(\frac{2\|\boldsymbol{s}_1\|_2^2}{\sigma_1^2}, \ldots, \frac{2\|\boldsymbol{s}_L\|_2^2}{\sigma_L^2}\right). \tag{52}$$

Substituting (46)–(52) into (44) yields the final CRLB expression of the target-position estimate.

6.3 Comparison of computational complexity with related methods

Let $\tilde{L}$ be the number of active nodes, $\tilde{G}$ the maximum active degree, K the number of temporal samples, Q the number of local LM iterations, and H the number of FU consensus rounds. Under the white-noise model $\boldsymbol{N}_l = \sigma_l^2\boldsymbol{I}_K$ in (4), the likelihood, matched-filter terms, and their first-order derivatives are dominated by length-K vector operations and therefore scale as $\mathcal{O}(K)$, rather than $\mathcal{O}(K^2)$. The aggregate scheduling cost is approximately $\mathcal{O}(\tilde{L}\tilde{G}K)$ and the aggregate LE cost is $\mathcal{O}(\tilde{L}KQ)$. Because nodes process in parallel, the corresponding per-node LE

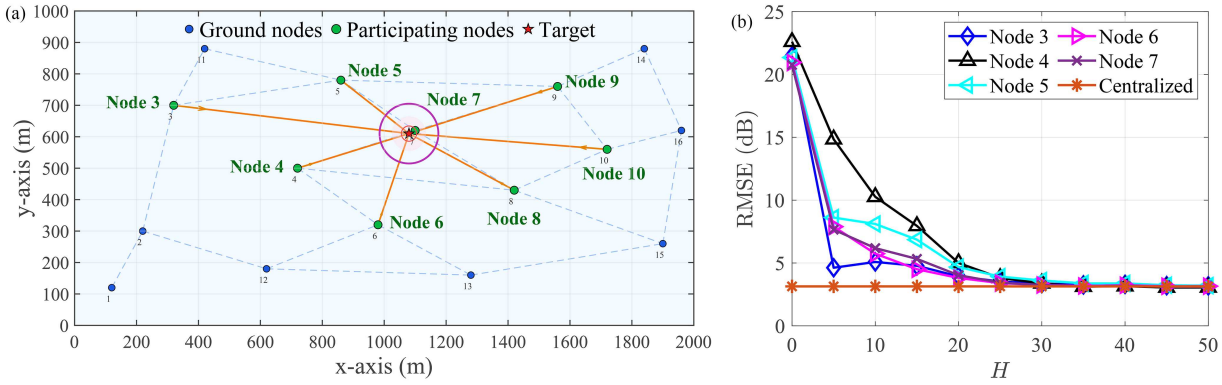

Figure 3 (Color online) Illustrations of the decentralized network operation and consensus behavior. (a) 2D mapping of the target-localization scenario in the decentralized multi-sensor network. The red pentagram denotes the target at an altitude of 125 m; the blue dashed lines and light red solid lines denote communication and sensing links, respectively; the green circles represent the activated sensing nodes selected by the scheduling strategy, while the blue circles denote sleeping nodes. (b) Localization performance of representative nodes, where $\gamma_l \sim \mathcal{N}(0, 10)$ and SNR $= -5$ dB.

cost is $\mathcal{O}(KQ)$; the inversion of a 4×4 LM matrix is constant with respect to K. A K^2 term would arise only if a dense colored-noise covariance were explicitly multiplied or inverted.

Under the same white-noise counting, Async EM-DPD and Async diffusion-DDPD scale approximately as $\mathcal{O}(\tilde{L} I_{\mathrm{EM}} J_{\mathrm{EM}} K)$ and $\mathcal{O}(\tilde{L} I_{\mathrm{DF}} K)$, respectively, whereas Async TOA-WLS remains approximately $\mathcal{O}(\tilde{L} \bar{M} J_{\mathrm{WLS}})$. The classical centralized 3D grid-search direct-localization method requires approximately $\mathcal{O}(LK\Upsilon_x\Upsilon_y\Upsilon_z N_\gamma)$. These expressions report aggregate arithmetic work; wall-clock latency of the decentralized method also depends on parallel hardware and neighbor communication.

6.4 Comparison of communication overhead with related methods

The scheduling stage requires approximately $\mathcal{O}(\tilde{L}\tilde{G})$ neighbor exchanges. In FU, $\widetilde{\boldsymbol{D}}_l$ is a symmetric 3×3 matrix, so only six independent entries are transmitted, together with the three entries of $\widetilde{\boldsymbol{w}}_l$ and one scalar information score used to form the local weight. Thus the algorithmic payload is $d_{\mathrm{DDL}} = 10$ real scalars per neighbor per consensus round, excluding headers, node identifiers, timestamps, and channel coding. Since the H consensus rounds are executed after LE convergence, the aggregate communication load scales as $\mathcal{O}(10\tilde{L}\tilde{G}H + \tilde{L}\tilde{G})$.

For the Async EM-DPD baseline, local nodes repeatedly exchange low-dimensional timing-related statistics during each EM iteration. If d_{EM} denotes the dimension of the exchanged variables and ς_{EM} the average hop count, its communication load is approximately $\mathcal{O}(d_{\mathrm{EM}} \tilde{L} I_{\mathrm{EM}} \varsigma_{\mathrm{EM}})$. For the Async diffusion-DDPD baseline, each diffusion step requires neighbor-to-neighbor exchange, yielding approximately $\mathcal{O}(d_{\mathrm{DF}} \tilde{L}\tilde{G} I_{\mathrm{DF}})$. Async TOA-WLS uses lightweight one-hop and two-hop timing information and therefore has the smallest raw communication overhead, approximately $\mathcal{O}(d_{\mathrm{WLS}} \tilde{L}\tilde{G})$. Overall, the proposed DDL provides a favorable balance between waveform-level localization performance and decentralized communication cost.

7 Numerical results

This section evaluates the proposed method in a representative low-altitude sensing scenario. We consider 3D localization of a single target in a 1000 m $\times$ 2000 m $\times$ 300 m surveillance region. In each Monte-Carlo run, the target position is randomly generated within this volume. The emitted signal is an LFM waveform with bandwidth 40 MHz and sampling rate 320 MHz. Because the numerical study evaluates localization conditioned on signal presence, no false-alarm parameter is required. The node-scheduling threshold is denoted by $\rho_{\mathrm{FI}} = 5 \times 10^{-16}$, distinct from the trust-region radius ρ_{TR}. The initial damping factor is $\mu_l^0 = 10$, and the number of Monte-Carlo trials is 1000. The clock offset γ_l follows $\gamma_l \sim \mathcal{N}(\mu_{\gamma_l}, \sigma_{\gamma_l}^2)$, and is converted into the corresponding distance by multiplying it by the speed of light. To provide a comprehensive performance comparison, the root mean square error (RMSE) and cumulative distribution function (CDF) are adopted as two evaluation metrics.

Figure 3 illustrates the overall localization performance of the proposed DDL method. Figure 3(a) shows the localization scenario and the selected sensing nodes. It is seen that the scheduling strategy preferentially activates nodes that are closer to the target and more favorable in topology. This is consistent with practical localization, since such nodes usually have higher SNRs and can provide more Fisher information. Figure 3(b) shows the convergence behavior of several representative nodes, where H denotes the number of consensus rounds. Although

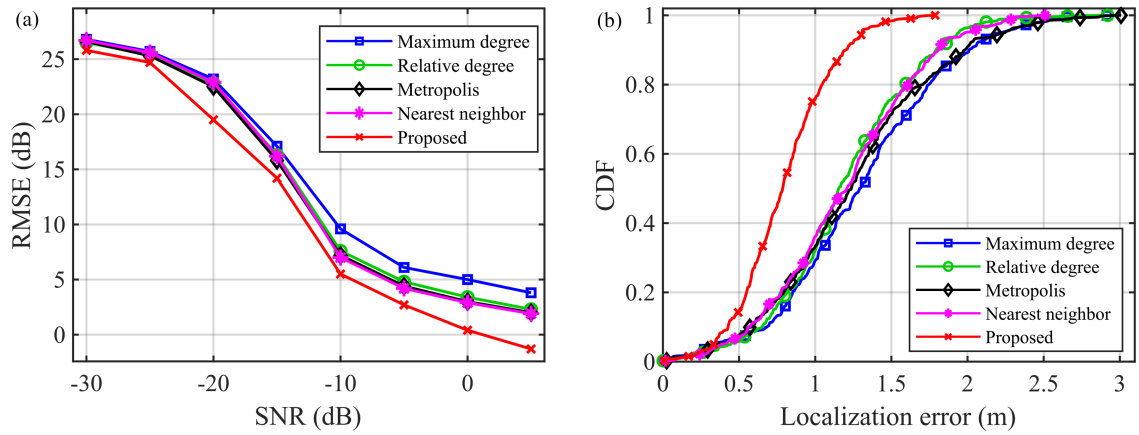

Figure 4 (Color online) Comparison of decentralized fusion strategies from complementary RMSE and CDF perspectives. (a) Localization RMSE of different fusion strategies versus the SNR; (b) localization error CDFs of different fusion strategies under $\gamma_l \sim \mathcal{N}(0, 15)$.

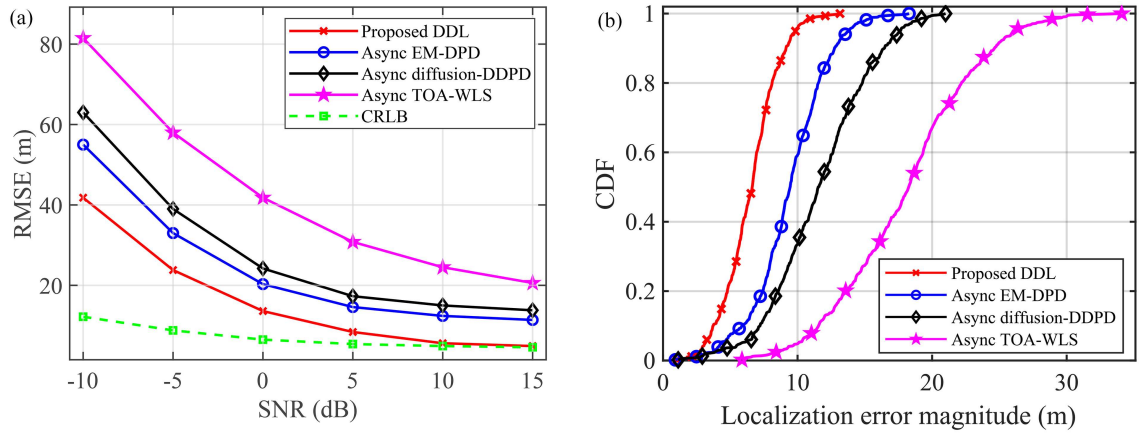

Figure 5 (Color online) Comparison with related decentralized localization methods. (a) Localization RMSE versus the SNR under $\gamma_l \sim \mathcal{N}(0, 10)$; (b) CDFs of the localization error under SNR = 0 dB and $\gamma_l \sim \mathcal{N}(0, 10)$.

their initial accuracies are different, all nodes gradually converge to a steady-state solution as consensus proceeds, and eventually approach the performance of the centralized scheme.

Figure 4 compares the proposed Fisher-information-weighted fusion rule with representative decentralized fusion strategies, including maximum-degree [43], relative-degree [44], Metropolis [27], and nearest-neighbor [45] methods. The proposed method consistently achieves the lowest RMSE and the most favorable CDF among all compared schemes. The gain comes from the fact that the proposed weights reflect the heterogeneous information quality across nodes, whereas degree-based rules mainly rely on network connectivity and do not fully exploit the informativeness of local measurements.

Figure 5 compares the proposed DDL with the three previously mentioned asynchronous baseline methods. It is observed that the proposed method achieves the lowest RMSE over the whole SNR range and also exhibits the most favorable CDF, especially in the low- and medium-SNR regime. The main reason is that the proposed DDL explicitly performs local joint estimation of the target position and clock offset, and then carries out fully decentralized information-weighted fusion over the network. As a result, both local timing mismatch and inter-node information heterogeneity are effectively handled in a unified framework. By contrast, Async EM-DPD still suffers from the approximation loss caused by repeated latent-variable refinement and aggregation-oriented updates, Async diffusion-DDPD is constrained by the diffusion-type surrogate update under asynchronous perturbation, and Async TOA-WLS inevitably loses part of the waveform-level information due to its lightweight two-step construction. Therefore, the proposed method provides a more robust and accurate decentralized solution in the considered asynchronous localization scenario.

Figure 6 further evaluates clock-bias robustness and sensor-scale scalability. In Figure 6(a), the localization error of all methods increases with the equivalent ranging bias standard deviation, while the proposed method degrades the slowest and remains closest to the CRLB. This indicates that the proposed local clock-offset compensation is more effective under asynchronous perturbation. In Figure 6(b), the RMSE decreases as the number of sensors

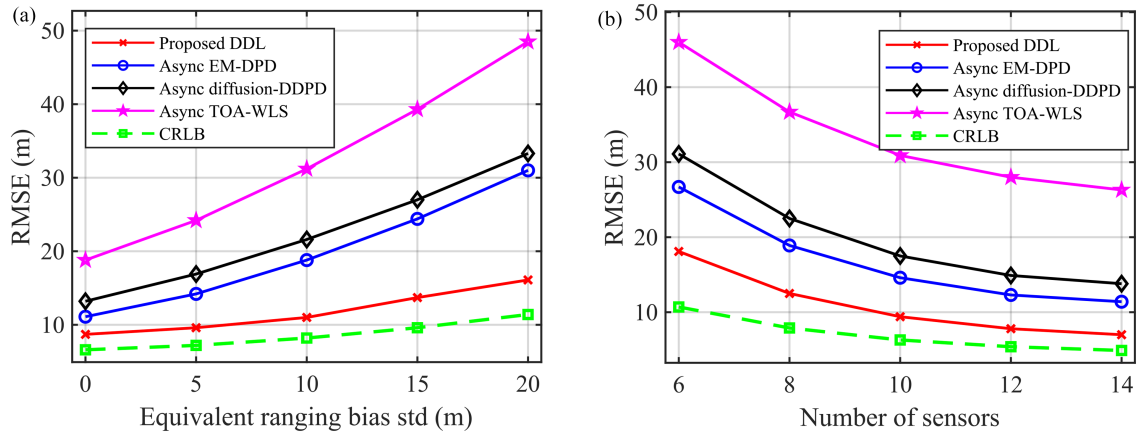

Figure 6 (Color online) Performance comparison of different methods in terms of clock-bias robustness and sensor-scale scalability. (a) Localization RMSE versus the equivalent ranging-bias standard deviation under $\gamma_l \sim \mathcal{N}(0, 10)$; (b) localization RMSE versus the number of sensors under SNR = 0 dB.

increases for all methods, but the proposed method benefits the most from additional nodes. This is because each extra node contributes both new local observations and additional compensated information to the decentralized fusion process, leading to better scalability than the competing baselines.

8 Conclusion

This paper developed a lightweight decentralized direct localization framework for clock-asynchronous multi-station sensing networks. The method consists of a local-estimation stage and a fusion-update stage. In the local stage, each node performs a regularized likelihood refinement of its local position-clock state around a coarse initializer; the single-node result is not interpreted as a globally identifiable 3D position. In the fusion stage, node-specific clock increments are eliminated locally and only reduced position-related messages are exchanged among neighbors. A position-Fisher-information-guided weighted consensus together with node scheduling is then used to improve global positioning accuracy while controlling communication cost. Numerical results show that the proposed approach achieves performance close to centralized direct localization and outperforms representative decentralized baselines in asynchronous conditions. These features make the framework attractive for low-altitude economy scenarios in which multiple decentralized sensing nodes must cooperatively localize aerial targets under bandwidth, energy, and robustness constraints.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 62601132, 62231008).

References

1 Li J Z, Guo F C, Jiang W L. Source localization and calibration using TDOA and FDOA measurements in the presence of sensor location uncertainty. Sci China Inf Sci, 2014, 57: 042315
2 Gao F, Cai Y Y, Deng F, et al. Prior area searching for energy-based sound source localization. Sci China Inf Sci, 2022, 65: 222204
3 Guo K X, Li X X, Xie L H. Simultaneous cooperative relative localization and distributed formation control for multiple UAVs. Sci China Inf Sci, 2020, 63: 119201
4 Xing T, Sun Y, Zhang K, et al. 3-D localization integrating 1-D SA and TDOA measurements: analysis and bias-reduced solutions. IEEE Trans Veh Technol, 2026, 75: 3099–3114
5 Pu Q, Jiang F, Zhou M, et al. Direct position determination algorithm toward coherent signals in low SNR localization scenarios. IEEE Trans Instrum Meas, 2025, 74: 1–13
6 Yi W, Zhou T, Ai Y, et al. Suboptimal low complexity joint multi-target detection and localization for non-coherent MIMO radar with widely separated antennas. IEEE Trans Signal Process, 2020, 68: 901–916
7 Fan X, Liu L, Guo M. Direct position determination via joint sparse representation of array covariance matrices based on angle and doppler. IEEE Trans Aero Elec Sys, 2025, 61: 17772–17784
8 Wang Z, Sun Y, Xie L, et al. An iterative direct position determination approach based on doppler frequency shifts. IEEE Trans Veh Technol, 2024, 73: 2431–2443
9 Li J, Dong X, Zhang X, et al. COMET-based direct position determination of incoherently distributed sources: algorithms and performance analysis. IEEE Trans Veh Technol, 2025, 74: 19801–19806
10 Zhang R, Wang S, Lu X, et al. Two-dimensional DoA estimation for multipath propagation characterization using the array response of PN-Sequences. IEEE Trans Wirel Commun, 2016, 15: 341–356
11 He J, Ho D K C, Xiong W, et al. Cramér-Rao lower bound analysis for elliptic localization with random sensor placements. IEEE Trans Aero Elec Sys, 2024, 60: 5587–5595
12 Wen G, Song H, Gao F, et al. Target localization in asynchronous distributed MIMO radar systems with a cooperative target. IEEE Trans Wirel Commun, 2022, 21: 4098–4113
13 Weiss A J. Direct position determination of narrowband radio frequency transmitters. IEEE Signal Proc Let, 2004, 11: 513–516
14 Godrich H, Haimovich A M, Blum R S. Target localization accuracy gain in MIMO radar-based systems. IEEE Trans Inform Theor, 2010, 56: 2783–2803

15 Tzoreff E, Weiss A J. Expectation-maximization algorithm for direct position determination. Signal Process, 2017, 133: 32–39
16 Yu Z, Li J, Guo Q, et al. Efficient direct target localization for distributed MIMO radar with expectation propagation and belief propagation. IEEE Trans Signal Process, 2021, 69: 4055–4068
17 Zhang G, Yi W, Varshney P K, et al. Direct target localization with quantized measurements in noncoherent distributed MIMO radar systems. IEEE Trans Geosci Remote Sens, 2023, 61: 1–18
18 Xia W, Liu W. Distributed adaptive direct position determination of emitters in sensor networks. Signal Process, 2016, 123: 100–111
19 Cattivelli F S, Sayed A H. Diffusion LMS strategies for distributed estimation. IEEE Trans Signal Process, 2010, 58: 1035–1048
20 Varshney P K. Distributed Detection and Data Fusion. New York: Springer, 2012
21 Zhao T, Nehorai A. Information-driven distributed maximum likelihood estimation based on Gauss-Newton method in wireless sensor networks. IEEE Trans Signal Process, 2007, 55: 4669–4682
22 Nedic̀ A, Olshevsky A, Rabbat M G. Network topology and communication-computation tradeoffs in decentralized optimization. Proc IEEE, 2018, 106: 953–976
23 Zhao X, Tu S Y, Sayed A H. Diffusion adaptation over networks under imperfect information exchange and non-stationary data. IEEE Trans Signal Process, 2012, 60: 3460–3475
24 Soatti G, Nicoli M, Savazzi S, et al. Consensus-based algorithms for distributed network-state estimation and localization. IEEE Trans Signal Inf Pr, 2017, 3: 430–444
25 Zhang Y, Lou Y, Hong Y, et al. Distributed projection-based algorithms for source localization in wireless sensor networks. IEEE Trans Wirel Commun, 2015, 14: 3131–3142
26 Wang Y, Ma X, Leus G. Robust time-based localization for asynchronous networks. IEEE Trans Signal Process, 2011, 59: 4397–4410
27 Lopes C G, Nascimento V H, Chamon L F O. Distributed universal adaptive networks. IEEE Trans Signal Proces, 2023, 71: 1817–1832
28 Yuan W, Wu N, Etzlinger B, et al. Expectation-maximization-based passive localization relying on asynchronous receivers: centralized versus distributed implementations. IEEE Trans Commun, 2019, 67: 668–681
29 Liu Y, Wang Y, Shen Y, et al. Hybrid TOA-AOA WLS estimator for aircraft network decentralized cooperative localization. IEEE Trans Veh Technol, 2023, 72: 9670–9675
30 Fang K, Chen J X, Zhu H, et al. Explainable-AI-based two-stage solution for WSN object localization using zero-touch mobile transceivers. Sci China Inf Sci, 2024, 67: 170302
31 Liu W J, Wu Y T, Jiang Q, et al. Eigenvalue-based distributed target detection in compound-Gaussian clutter. Sci China Inf Sci, 2025, 68: 199301
32 Xiao D P, Liu W J, Liu J, et al. Bayesian Rao test for distributed target detection in interference and noise with limited training data. Sci China Inf Sci, 2026, 69: 122301
33 Xia W, Xia X, Li H, et al. A noise-constrained distributed adaptive direct position determination algorithm. Signal Process, 2017, 135: 9–16
34 Pourhomayoun M, Fowler M. Distributed computation for direct position determination emitter location. IEEE Trans Aero Elec Sys, 2014, 50: 2878–2889
35 Ma F, Liu Z M, Guo F. Distributed direct position determination. IEEE Trans Veh Technol, 2020, 69: 14007–14012
36 Wang Z, Tan B, Valkama M, et al. Decentralized indoor direct localization with multiple Wi-Fi access points. IEEE Trans Wirel Commun, 2026, 25: 3091–3105
37 Chen Z, Weng Y, Peng D. A Low-complexity direct position determination method based on the diffusion strategy in wireless sensor array networks. IEEE Trans Circuits-II, 2023, 70: 4584–4588
38 Ma F, Liu Z M, Guo F. Direct position determination in asynchronous sensor networks. IEEE Trans Veh Technol, 2019, 68: 8790–8803
39 Hlinka O, Sluciak O, Hlawatsch F, et al. Likelihood consensus and its application to distributed particle filtering. IEEE Trans Signal Process, 2012, 60: 4334–4349
40 Li J, Li G, Xie H, et al. Low-complexity Gaussian-Newton method for multi-modulus algorithm-based blind equalization. Signal Process, 2022, 201: 108722
41 Yuan Y X. A review of trust region algorithms for optimization. In: Proceedings of the 4th International Congress on Industrial & Applied Mathematics. Oxford: Oxford University Press, 2000. 271–282
42 Wu M, Zhong L, Xu B, et al. A consensus-based diffusion levenberg-marquardt method for collaborative localization with extension to distributed optimization. IEEE Access, 2020, 8: 215649
43 Pérez Herrera D, Chen Z, Larsson E G. Faster convergence with less communication: broadcast-based subgraph sampling for decentralized learning over wireless networks. IEEE Open J Commun Soc, 2025, 6: 1497–1511
44 Liang C, Wen F, Wang Z. Trust-based distributed Kalman filtering for target tracking under malicious cyber attacks. Inf Fusion, 2019, 46: 44–50
45 Shao H, Pan L, Mesbahi M, et al. Distributed neighbor selection in multiagent networks. IEEE Trans Autom Contr, 2023, 68: 6711–6726

Appendix A Proof of proposition 1

At a common position linearization point, the centralized damped local model is the sum of the independent node log-likelihood approximations. For node l, partition the 4×4 local LM system as in (34). The corresponding local normal equation is

$$\begin{bmatrix} \boldsymbol{D}_{uu,l} & \boldsymbol{d}_{u\gamma,l} \\ \boldsymbol{d}_{\gamma u,l} & d_{\gamma\gamma,l} \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{u} \\ \Delta\gamma_l \end{bmatrix} = - \begin{bmatrix} \boldsymbol{w}_{u,l} \\ w_{\gamma,l} \end{bmatrix}. \tag{A1}$$

Eliminating the node-specific $\Delta\gamma_l$ gives

$$\widetilde{\boldsymbol{D}}_l \Delta \boldsymbol{u} = -\widetilde{\boldsymbol{w}}_l \tag{A2}$$

with the reduced quantities in (35). Since the observations are independent, the centralized linearized position normal equation is obtained by summing these reduced node contributions,

$$\left(\sum_{l \in \Psi} \widetilde{\boldsymbol{D}}_l \right) \Delta \boldsymbol{u}^C = - \sum_{l \in \Psi} \widetilde{\boldsymbol{w}}_l, \tag{A3}$$

so that $\Delta \boldsymbol{u}^C = -(\sum_l \widetilde{\boldsymbol{D}}_l)^{-1} \sum_l \widetilde{\boldsymbol{w}}_l$. Hence, the reduced matrix-vector pairs contain the curvature and gradient contributions required to reconstruct the centralized current linearized position update. This proves the stated decomposition. It does not establish statistical sufficiency for the original nonlinear observations and does not guarantee a global optimum of the non-convex likelihood.