Cluster-based Accurate Localization in Noisy and Sparse Unmanned Sensing Systems

Bingjie Han[1], Yingxu Lai[1,2], Haodi Ping[1]
{hbj1995@emails.bjut.edu.cn, laiyingxu@bjut.edu.cn, haodi.ping@bjut.edu.cn}

[1]School of Computer Science, Beijing University of Technology, Beijing, P.R.China
[2]Engineering Research Center of Intelligent Perception and Autonomous Control, Ministry of Education, Beijing, P.R.China

Abstract. In unmanned sensing systems, ranging-based localization over noisy and sparse measurements often converges to stable yet incorrect configurations that satisfy sensing constraints but deviate from the true spatial geometry. To mitigate this issue, we propose Clustered Localization via Variance-weighted Edge Reinforcement (CLOVER), which forms robust local clusters, derives coarse inter-cluster geometry from lightweight hop-based cues, and refines the global layout via variance-weighted graph optimization. CLOVER enforces global geometric consistency while maintaining efficiency by compressing nodes into clusters and progressively stitching them at the cluster level. Evaluation results show that CLOVER preserves localization accuracy and geometric stability under noisy and sparse deployments.

Keywords: ranging-based localization, noisy and sparse sensing, unmanned system

1 Introduction

1.1 Motivation

With the rapid proliferation of unmanned sensing systems, large numbers of interconnected IoT devices are being deployed to build ubiquitous sensing and pervasive computing environments. In these settings, precise localization has become an important capability, since many higher-level functions in autonomous interaction, machine perception, and situational awareness rely on accurate knowledge of where sensing and computing nodes are located [1][2]. A common way to obtain such positional information is to exploit sensing-based localization, where IoT devices measure physical signals from their surroundings and infer locations from these observations. Among these sensing modalities, wireless ranging has emerged as a particularly practical due to recent improvements in distance measurement technologies, including Ultra Wideband [3], WiFi RTT [4], and LoRa RTT [5]. In this paradigm, pairwise ranges are collected as sensing data for an optimization problem whose solution provides the estimated node positions.

However, in practical sensing environments, multipath propagation and non-line-of-sight effects introduce ranging noise, while the naturally sparse spatial connectivity among nodes, caused

IIKI 2025, December 18-21, Chengdu, People's Republic of China

DOI 10.4108/eai.18-12-2025.2365296

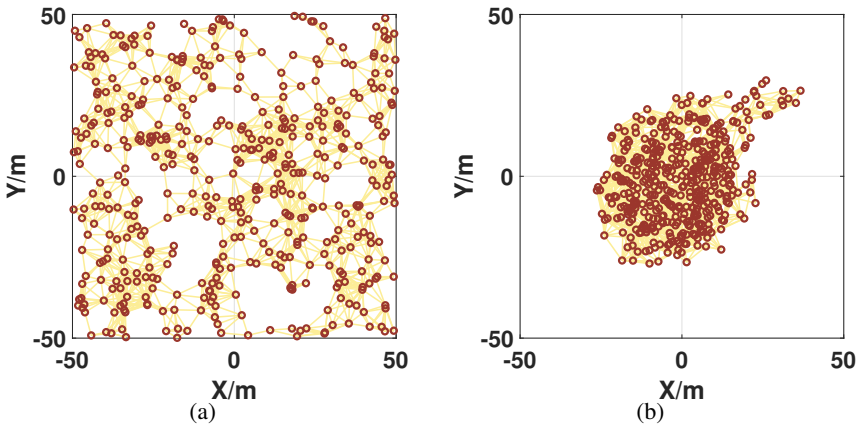


Fig. 1. (a) The ground truth of a network. (b) The localization result.

by limited communication range, obstacles, or energy constraints, further deteriorates geometric stability [6][7][8]. As a result, the underlying optimization problem often converges to a stable yet incorrect suboptimal solution, yielding node configurations that are algebraically admissible but markedly differ from the physical ground truth, as illustrated in Fig. 1. This mismatch undermines the reliability of ubiquitous sensing applications that depend on accurate localization. To address this issue, a series of research develops likelihood-based residual refinement and theoretical performance bounds [9, 10], relying on prior environmental knowledge or extensive measurement data to enable effective noise suppression. Another line of work exploits geometric relationships among nodes to infer additional constraints beyond the available distance measurements, such as negative-edge inference and path-correction mechanisms [11, 12]. Yet these techniques remain effective only in local neighborhoods and cannot guarantee algebraic–geometric consistency at larger network scales. Therefore, there is a pressing need for localization algorithms that can reliably solve the underlying optimization problem in realistic sensing conditions, while preventing solutions that are mathematically feasible but physically implausible.

1.2 Related Works

In the context of ubiquitous sensing, ranging-based localization is typically formulated as a nonlinear least-squares problem over a measurement graph. General graph optimization (G2O) frameworks have been widely adopted for solving such problems due to their efficiency and simplicity in iterative refinement [13]. However, G2O is highly sensitive to the initialization of node positions. To obtain a feasible starting configuration, various preprocessing methods, including SDP [14], SMACOF [15] and ARAP [16], are typically combined to produce reasonable layouts under noise-free or low-noise conditions. Yet, when applied to real deployments characterized by significant ranging errors and sparse network connectivity, these initialization strategies still fail to ensure accurate localization. On one hand, several studies attempt to mitigate measurement noise through

filtering techniques or system-model-based inference [9, 10]. Such approaches rely on prior environmental knowledge, historical measurement patterns, or temporal sequences to compensate for noise, making them ineffective in unfamiliar scenarios or one-shot localization settings. On the other hand, a different class of methods exploits geometric relationships among nodes to infer additional constraints beyond the available distance measurements [11, 12]. These techniques, however, are often restricted to local neighborhoods or require extensive rollback mechanisms when extended to the global scale, which conflicts with the real-time and scalability requirements of practical IoT and ubiquitous sensing deployments.

1.3 Proposed Method and Contributions

To overcome the critical issue that optimization-based ranging localization converging to algebraically feasible but physically implausible results, we introduce a clustered localization via variance-weighted edge reinforcement (CLOVER) method. CLOVER first forms stable local clusters that can be robustly recovered even under noisy ranging and sparse connectivity. It then extracts lightweight hop-based cues to capture coarse inter-cluster relations without requiring additional sensing overhead. Finally, it performs global refinement through variance-weighted graph optimization, reinforcing structurally reliable edges while down-weighting uncertain ones, thereby improving algebraic–geometric consistency at the network scale. The main contributions are as follows.

- We propose CLOVER, which constructs reliable local clusters, derives coarse inter-cluster geometry from hop information, and refines the entire layout with a variance-weighted optimizer tailored to sparse and noisy ranging graphs in ubiquitous sensing environments.
- Through evaluations across different node densities and noise levels that mimic typical IoT and pervasive computing deployments, we demonstrate that CLOVER improves localization accuracy and geometric stability, providing a more reliable localization backbone for higher-level machine perception and intelligent applications.

The problem formulation and System Characteristics are presented in Section 2. Section 3 details the CLOVER algorithm. Experimental validations are provided in Section 4. Section 5 concludes this work.

2 Unmanned Sensing System

2.1 Problem Formulation

In an unmanned sensing system, each device is modeled as a node, and the collection of all nodes is denoted by $\mathscr{V} = \{v_1, \ldots, v_n\}$. Whenever a node v_i detects another node v_j within its sensing radius R_{det}, a distance reading d_{ij} is produced, and an undirected edge (i, j) is included in the measurement set $\mathscr{E}$. The recorded distance deviates from the true separation due to various environmental factors, and can be expressed as

$$d_{ij} = \|\mathbf{p}_i - \mathbf{p}_j\| + \sigma_{ij} \tag{1}$$

where σ_{ij} denotes the ranging error. This error consists of multiplicative and additive components:

$$\sigma_{ij} = \varepsilon_{ij}^{m} \|\mathbf{p}_i - \mathbf{p}_j\| + \varepsilon_{ij}^{a}, \tag{2}$$

with $\varepsilon_{ij}^{m} \sim \mathcal{N}(0, (\sigma^m)^2)$ and $\varepsilon_{ij}^{a} \sim \mathcal{N}(0, (\sigma^a)^2)$ assumed independent[12]. In practice, with the advancement of wireless ranging modules, the additive errors of measurements obtained by UWB and similar sensors are often below 10 cm, making them negligible relative to the typical sensing distances[3]. Therefore, the additive noise can be safely ignored, i.e., $\varepsilon_{ij}^{a} = 0$. The measured distance $d_{ij} \in \mathscr{E}$ can finally be expressed as:

$$d_{ij} = (1 + \varepsilon_{ij}^{m}) \|\mathbf{p}_i - \mathbf{p}_j\|, \; \varepsilon_{ij}^{m} \sim \mathcal{N}(0, (\sigma^m)^2). \tag{3}$$

Together, $\mathscr{V}$ and $\mathscr{E}$ define the measurement graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$. Ranging-based localization aims to infer the node coordinates using the pairwise constraints encoded in $\mathscr{E}$. This problem is typically formulated as minimizing a cost function that accumulates the residuals between the current geometric configuration and the measured distances:

$$P1 : \min \sum (||p_i - p_j|| - d_{ij})^2 \tag{4}$$

$$s.t. d_{ij} = (1 + \varepsilon_{ij}^{m}) \|\mathbf{p}_i - \mathbf{p}_j\| \tag{4a}$$

$$\varepsilon_{ij}^{m} \sim \mathcal{N}(0, (\sigma^m)^2). \tag{4b}$$

$$\mathbf{p_i}, \mathbf{p_j} \in \mathscr{V}, i \neq j \tag{4c}$$

$$d_{ij} \in \mathscr{E} \tag{4d}$$

2.2 System Characteristics

The measurement graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ induces a probabilistic model over node positions. Each distance observation contributes a likelihood term

$$p(d_{ij} \mid \mathbf{p}_i, \mathbf{p}_j) \propto \exp\left(-\frac{(\|\mathbf{p}_i - \mathbf{p}_j\| - d_{ij})^2}{2(\sigma^m d_{ij})^2} \right), \tag{5}$$

leading to the posterior

$$p(\mathbf{P} \mid \mathscr{E}) \propto \prod_{(i,j) \in \mathscr{E}} p(d_{ij} \mid \mathbf{p}_i, \mathbf{p}_j). \tag{6}$$

Under standard linearization, the corresponding Fisher information matrix (FIM) takes the form $\mathbf{J} = \mathbf{H}^{\mathsf{T}} \blacksquare \mathbf{H}$, where $\mathbf{H}$ reflects the sparsity of $\mathscr{G}$ and $\blacksquare$ is a diagonal matrix weighting each edge by $(\sigma^m d_{ij})^{-2}$. Because sensing is range-limited, $\mathscr{G}$ often contains several locally dense regions connected only by a few long-range edges. Thus $\mathbf{J}$ is approximately block-structured:

$$\mathbf{J} \approx \begin{bmatrix} J_1 & J_{12} & 0 & \cdots \\ J_{21} & J_2 & J_{23} & \cdots \\ 0 & J_{32} & J_3 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \tag{7}$$

Within each dense region, the diagonal blocks J_k aggregate many strong constraints and are typically well conditioned. In contrast, the cross-region blocks J_{ij} contain far fewer edges, and those edges usually have larger true distances, resulting in weaker statistical weight in ■. Hence the global information $\mathbf{J}_{\text{inter}} = \sum_{(i,j)\in\mathscr{E}_{\text{inter}}} \mathbf{H}_{ij}^{\mathsf{T}}(\sigma^m d_{ij})^{-2}\mathbf{H}_{ij}$, may contribute only a small fraction of the overall Fisher information. When $\lambda_{\min}(\mathbf{J})$ becomes small, the global configuration is only weakly identifiable: the posterior admits families of solutions differing mainly in the relative placement of regions (e.g., flips or rotations). Standard optimization methods then tend to drift along these low-curvature directions, potentially causing large-scale geometric distortions despite locally consistent measurements.

3 CLOVER Algorithm

This chapter introduces the Clustered Localization via Variance-weighted Edge Reinforcement (CLOVER) method. The method performs local cluster formation, inter-cluster structural cue extraction and stitching, and global pose refinement via G2O, enabling robust and efficient localization in unmanned sensing system.

3.1 Cluster Aggregation

Given the locally dense yet globally sparse structure described in the last section, the first step of the proposed method is to aggregate spatially coherent node groups into a set of local clusters. This procedure aims to (i) suppress the adverse effects of noisy long edges and (ii) reduce the complexity of subsequent global optimization.

3.1.1 Edges Cutting

As established in the system model, range measurements become increasingly unreliable as the distance approaches the sensing radius. Moreover, long edges often function either as weak inter-region bridges or as redundant constraints within dense neighborhoods. To avoid propagating these unstable relations into the clustering stage, we remove edges longer than a fraction of the sensing radius. For a threshold $\beta \in (0,1)$, an edge (i,j) is discarded if

$$d_{ij} > \beta R_{\text{det}} \tag{8}$$

The resulting pruned graph retains only short, structurally reliable edges, which capture the intrinsic local connectivity pattern of the network.

3.1.2 Community Detection Method for Cluster Aggregation

After edge pruning, we apply the weighted Louvain algorithm to partition the graph into local clusters. Unlike hierarchical use in multi-level abstraction, we employ Louvain only once to identify densely connected node groups, producing clusters $\mathscr{C} = \{\mathscr{C}_1, \mathscr{C}_2, \ldots, \mathscr{C}_K\}$. The algorithm

maximizes the standard modularity function

$$Q = \frac{1}{2m} \sum_{i,j} \Big(W_{ij} - \frac{k_i k_j}{2m} \Big) \delta(c_i, c_j), \tag{9}$$

where W_{ij} is the pruned edge weight, $k_i = \sum_j W_{ij}$ is the weighted degree, and $m = \frac{1}{2} \sum_{i,j} W_{ij}$ is the total edge weight. To emphasize short and stable constraints, we use

$$W_{ij} = \frac{1}{d_{ij}}, \tag{10}$$

which naturally strengthens dense local neighborhoods and weakens residual long edges. The output clusters form a set of compact local subgraphs that serve as the basic units for subsequent coarse alignment and global optimization.

3.2 Hop Extraction and Cluster Stitching

Let the pairwise measurements be arranged in a distance matrix $\mathbf{D}_m \in \mathbb{R}^{N \times N}$ with $[\mathbf{D}_m]_{ij} = d_{ij}$, and let $\mathbf{B} \in \{0,1\}^{N \times N}$ denote sensing connectivity ($[\mathbf{B}]_{ij} = 1$ if d_{ij} is observed). We first recover multi-hop relations by Boolean layer expansion: $\mathbf{C}^{(1)} = \mathbf{B}$, $\mathbf{C}^{(\ell)} = (\mathbf{C}^{(\ell-1)}\mathbf{B} > 0) \vee \mathbf{C}^{(\ell-1)}$, $\ell \geq 2$, where $(\cdot) > 0$ maps positive entries to 1 and "$\vee$" is elementwise OR. Accumulate hop counts by

$$\mathbf{H}_{\text{hop}} = \sum_{\ell=1}^{L_{\max}} \ell \, \mathbf{C}^{(\ell)}. \tag{11}$$

To prevent inconsistent growth, maintain the reachability mask $\mathbf{R}^{(\ell)} = \bigvee_{k=1}^{\ell} \mathbf{C}^{(k)}$ and accept layer ℓ only when

$$\sum_{i,j} \mathbf{C}^{(\ell)}_{ij} = \frac{\ell(\ell-1)}{2}. \tag{12}$$

After local clustering, cluster stitching uses three representative nodes per cluster to form robust inter-cluster anchors. For adjacent clusters C_i and C_j we select three node pairs $\{(u_t, v_t)\}_{t=1}^{3}$ by the following compact criteria:

$$\begin{cases} (u_1, v_1) = \arg \min\limits_{u \in C_i, v \in C_j} d(u,v), \\ (u_3, v_3) = \arg \max\limits_{u \in C_i, v \in C_j} d(u,v), \\ (u_2, u_2) = \arg\min \operatorname{std}\{d(x,y) : x, y \in C_i \text{ or } C_j\}, \end{cases} \tag{13}$$

where the third rule picks internally stable nodes with low intra-cluster distance variance. These estimated distances yield the inter-cluster constraints added to the optimizer:

$$\mathscr{E}_{ij} = \sum_{t=1}^{3} w_t \big(\|x_{u_t} - x_{v_t}\| - \hat{d}_{ij}(t) \big)^2, \tag{14}$$

Input: $\mathscr{G} = (\mathscr{V}, \mathscr{E})$;
Output: Cluster set: $\mathscr{C}$;
Obtaining $\mathscr{G}'$ from cutting G by Eq. (8).;
Set weights for edges of $\mathscr{G}'$ by Eq. (10).;
Get the cluster $\mathscr{C}$ by Eq. (9);
return $\mathscr{C}$;

Algorithm 1: Cluster Aggregation Algorithm

with weights w_t set inversely proportional to hop-count variance. In summary, we (i) extract a topologically consistent hop-count matrix, (ii) select three complementary representative pairs per adjacent cluster, and (iii) convert hop statistics into a small set of weighted inter-cluster distance constraints. This compact backbone suppresses noisy long edges and supplies stable coarse constraints for the subsequent local and global G2O refinement.

3.3 CLOVER Algorithm

3.3.1 Cluster Aggregation Algorithm

Cluster formation aggregates nodes into locally dense blocks using the pruned, weighted graph (edge pruning threshold Eq. (8) and weights Eq. (10). We apply a community detection method with Eq. (9) on the pruned graph to yield clusters $\{\mathscr{C}_k\}$ that serve as local reconstruction units.

3.3.2 Residual-based G2O formulation

To begin, we formulate the residual-based optimization problem from $P1$. For an unmanned sensing system network represented by $\mathscr{G}$, define the pairwise residual $\mathbf{e}(\mathbf{p_i}, \mathbf{p_j}, d_{ij}) = \left(\|\mathbf{p_i} - \mathbf{p_j}\| - d_{ij}\right)^2$. Thus $P1$ can be written as

$$\begin{aligned} P2 : F(\mathbf{P}) &= \sum_{(i,j)\in\mathscr{E}} \mathbf{e}(\mathbf{p_i}, \mathbf{p_j}, d_{ij})^\top \Omega_{ij}\, \mathbf{e}(\mathbf{p_i}, \mathbf{p_j}, d_{ij}), \\ \mathbf{P}^* &= \arg\min_{\mathbf{P}} F(\mathbf{P}), \end{aligned} \tag{15}$$

where Ω_{ij} is the per-edge weight matrix. For an edge $(i, j) \in \mathscr{E}$, let current estimates be $\hat{\mathbf{p}}_i, \hat{\mathbf{p}}_j$ and increments be $\Delta\hat{\mathbf{p}}_i, \Delta\hat{\mathbf{p}}_j$. A first-order Taylor expansion yields $\mathbf{e}_{ij}(\hat{\mathbf{p}} + \Delta\hat{\mathbf{p}}) \approx \mathbf{e}_{ij} + \mathbf{J}_{ij}\Delta\hat{\mathbf{p}}$, where $\mathbf{J}_{ij}$ denotes the Jacobian of the residual w.r.t. the stacked pose increment $\Delta\hat{\mathbf{p}}$. The local cost contribution becomes

$$\begin{aligned} F_{ij} &= \mathbf{e}_{ij}(\hat{\mathbf{p}} + \Delta\hat{\mathbf{p}})^\top \Omega_{ij}\, \mathbf{e}_{ij}(\hat{\mathbf{p}} + \Delta\hat{\mathbf{p}}) \\ &\approx (\mathbf{e}_{ij} + \mathbf{J}_{ij}\Delta\hat{\mathbf{p}})^\top \Omega_{ij}\, (\mathbf{e}_{ij} + \mathbf{J}_{ij}\Delta\hat{\mathbf{p}}) \\ &= \mathbf{e}_{ij}^\top \Omega_{ij} \mathbf{e}_{ij} + 2\,\mathbf{e}_{ij}^\top \Omega_{ij} \mathbf{J}_{ij}\Delta\hat{\mathbf{p}} + \Delta\hat{\mathbf{p}}^\top \mathbf{J}_{ij}^\top \Omega_{ij} \mathbf{J}_{ij}\Delta\hat{\mathbf{p}}. \end{aligned} \tag{16}$$

Input: graph $\mathscr{G}^{in} = (\mathscr{V}^{in}, \mathscr{E}^{in})$
Output: updated poses $\mathbf{P}$
initialize $\mathbf{H} \leftarrow \mathbf{0}$, $\mathbf{b} \leftarrow \mathbf{0}$;
forall *edges* $(i, j) \in \mathscr{E}^{in}$ **do**
  compute residual $\mathbf{e}_{ij}$ and Jacobia $\mathbf{J}_{ij}$;
  $\mathbf{H} \leftarrow \mathbf{H} + \mathbf{J}_{ij}^{\top} \Omega_{ij} \mathbf{J}_{ij}$;
  $\mathbf{b} \leftarrow \mathbf{b} + \mathbf{J}_{ij}^{\top} \Omega_{ij} r_{ij}$;
end
solve for the increment: $\mathbf{H}\Delta\mathbf{P} = -\mathbf{b}$;
update poses: $\mathbf{P} \leftarrow \mathbf{P} + \Delta\mathbf{P}$;
return $\mathbf{P}$;

Algorithm 2: Position Update by G2O

Define $\mathbf{b}_{ij} = \mathbf{J}_{ij}^{\top} \Omega_{ij} e_{ij}$, and $\mathbf{H}_{ij} = \mathbf{J}_{ij}^{\top} \Omega_{ij} \mathbf{J}_{ij}$. The variation of F_{ij} is $\Delta F_{ij} = 2\,\mathbf{b}_{ij}^{\top} \Delta\hat{p} + \Delta\hat{p}^{\top} \mathbf{H}_{ij} \Delta\hat{p}$, and setting the derivative to zero yields the normal equation $\mathbf{H}_{ij} \Delta\hat{p} = -\mathbf{b}_{ij}$. Following the aggregation rationale, we set weights inversely proportional to measured distance $\Omega_{ij} \propto \frac{1}{d_{ij}}$.

3.3.3 CLover Instant

The CLover Instant module performs hierarchical graph localization through three stages. It first partitions the raw graph into local clusters and solves each cluster independently. It then enters an iterative stitching process, where inter-cluster constraints are constructed to estimate the optimal transformations between clusters, progressively integrating them into a unified structure. After all clusters are merged, a final global refinement is applied to the entire graph to produce the consolidated localization result.

3.4 Time Complexity Analysis

The CLOVER algorithm contain X parts: (i)cluster aggregation, (ii) G2O Optimizer and (iii) edge completions of cluster. Given a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ with average degree $\overline{D}$. The iteration of G2O optimization is τ_{max} and the cluster number is K, which both are much smaller than number of nodes and edges. The cluster aggregation is $O(n \log \overline{D})$[17]. The G2O Optimizer for all optimization is $O(\tau_{max} n \overline{D}^2)$. The edge completions of cluster is $O(K)$. Thus, the total time complexity is $O(n\overline{D}^2)$.

4 Experimental Evaluation

This section evaluates the proposed method in terms of localization accuracy and computational efficiency across varying network densities and noise levels. Network density is characterized by the number of nodes N and the average degree $\overline{D}$, and noise is controlled by the standard deviation σ^m.

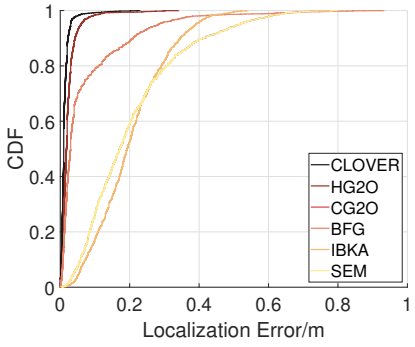
Input: graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$
Output: localization $\mathbf{P}$
/* Cluster Generation */
run Algorithm 1 on $\mathscr{G}$ to obtain $\{\mathscr{G}_1, \ldots, \mathscr{G}_K\}$;
forall *clusters* $\mathscr{G}_k$ **do**
  run Algorithm 2 on $\mathscr{G}_k$ to obtain $\mathbf{P}_k$;
end
/* Cluster Stitching */
initialize merged graph $\mathscr{G}^{\text{merge}}$ with one cluster;
while *there exist unmerged clusters* **do**
  select pair $(\mathscr{G}_a, \mathscr{G}_b)$ with inter-cluster edges;
  construct stitching constraints by Eq. (11) (12) (13) (14);
  run Algorithm 2 on $(\mathscr{G}_a, \mathscr{G}_b)$ to estimate transform and merge them into $\mathscr{G}^{\text{merge}}$;
end while
/* Global Refinement */
run Algorithm 1 on $\mathscr{G}^{\text{merge}}$ to obtain $\mathbf{P}$;
return $\mathbf{P}$;

Algorithm 3: CLover Instant Pipeline

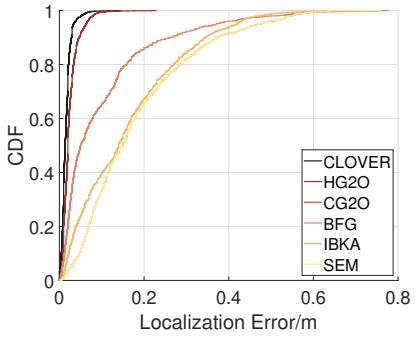
We first present a visualization of the localization process. Then, the method is compared with state-of-the-art algorithms using the average localization error $\frac{1}{n}\sum_{i=1}^{n}\|\hat{\mathbf{p}}_i - \mathbf{p}_i\|$. Efficiency is analyzed via runtime comparison.

To evaluate the performance of the CLOVER, several representative approaches were selected for comparison. These methodologies cover distinct theoretical pathways for addressing the localization. Specifically, the compared approaches include:

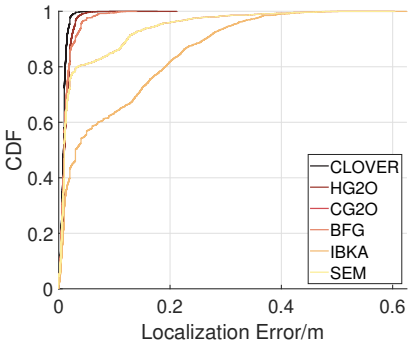
- Hop–Radius Constrained G2O (HG2O) introduces hop-derived upper bounds to augment sparse graphs with virtual constraints for global optimization.
- Cluster-Based G2O (CG2O) views the graph as a set of local clusters and refines their global arrangement through G2O optimization.[16].
- Blurred-Fragment Graph methods (BFG) enhance localization reliability by removing ambiguous configurations in fragment graphs [12, 18].
- IBKA, as an intelligent algorithm, leverages data-driven learning strategies to suppress noise and improve localization performance [19].
- Statistical Estimation methods (SEM) mitigate measurement bias through likelihood modeling, residual analysis, and Bayesian inference [9].



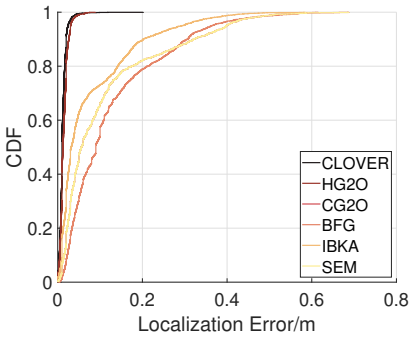
(a) CDF, $\overline{D} = 11$, $\sigma^m = 0.10$



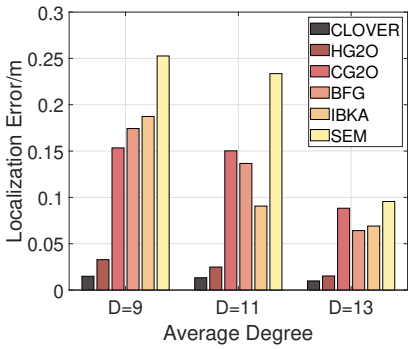
(b) CDF, $\overline{D} = 11$, $\sigma^m = 0.20$



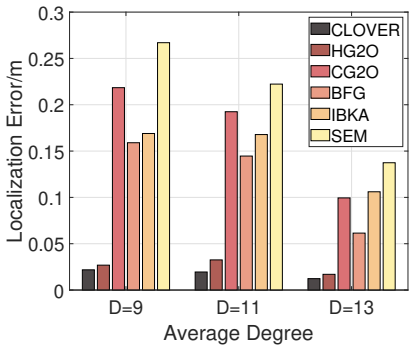
(c) CDF, $\overline{D} = 13$, $\sigma^m = 0.10$



(d) CDF, $\overline{D} = 13$, $\sigma^m = 0.20$



(e) $\overline{D} \in \{9, 11, 13\}$, $\sigma^m = 0.10$



(f) $\overline{D} \in \{9, 11, 13\}$, $\sigma^m = 0.20$

Fig. 2. Comparing the localization performances of different algorithms in different ranging radius, and noise level settings.

4.1 Accuracy Comparison and Ablation Verification

Fig. 2 provides a detailed evaluation using the Cumulative Distribution Function (CDF) of localization errors. Under the two noise conditions shown in the first two rows, the CDF of the proposed algorithm consistently lies leftmost, indicating lower errors and higher accuracy than HG2O, particularly in high-noise scenarios. The third row shows the evolution of average error with respect to the average node degree. As expected, all methods benefit from increased network density or reduced noise. A closer examination further shows that under extreme conditions with sparse connectivity and substantial noise—where limited measurement redundancy becomes the dominant bottleneck—the proposed method maintains the most stable and reliable localization performance.

The comparison with CG2O baseline reveals the effectiveness of the three technical components in CLOVER. The performance degradation of CG2O in sparse or noisy regions reflects the absence of edge screening, structured weighting, and rigid three-anchor stitching. CLOVER addresses these weaknesses by filtering unreliable edges, stabilizing optimization through informed weights, and enforcing consistent inter-module alignment. The resulting and consistently higher accuracy demonstrates the contribution of each component and the advantage of their combined design.

4.2 Efficiency Comparison

Table 1 summarizes the overall time consumption of several representative baseline algorithms under different parameter settings. The proposed CLOVER algorithm reduces computation time by nearly 70% compared with the HG2O baseline, representing a substantial improvement in efficiency. Relative to convex optimization methods that rely on noise-free correction models, CLOVER requires roughly two to three times more processing time. Even so, it remains considerably faster than learning-based approaches, and this advantage becomes increasingly pronounced as the node density and average degree grow. When the network is sparse, its runtime is comparable to statistical estimation methods, whereas in denser settings CLOVER demonstrates superior scalability. Moreover, its computational cost is on the same order as Intelligent algorithm.

5 Conclusion

This paper addressed the localization problem for unmanned sensing systems under sparse and noisy measurements, where conventional algebraic formulations may become inconsistent with the true geometry of the network. From a unified modeling perspective, we showed that such systems often exhibit a characteristic structure that is locally dense yet globally sparse, which motivates a hierarchical treatment of the estimation problem. Based on this observation, we proposed the CLOVER algorithm, which first groups nodes into effective clusters, then builds inter-cluster adjacency constraints, and finally performs cluster stitching followed by global refinement through an efficient optimization procedure. The proposed framework improves both robustness and consistency by reducing the difficulty of direct global estimation while still preserving the essential geometric relationships across the network. Experimental results on networks with different densities and noise levels demonstrated that CLOVER achieves stable and reliable localization performance under

Table 1: The Algorithm Time Consumption (s)

Para Setting	Algorithm					
$\{N, \overline{D}, \sigma^m\}$	CLOVER	HG2O	CG2O	BFG	IBKA	SEM
50, 9, 0.1	**0.71**	0.77	0.32	0.45	0.67	1.30
50, 9, 0.2	**0.65**	0.85	0.36	0.38	0.94	1.23
50, 11, 0.1	**0.75**	0.81	0.39	0.48	1.00	1.78
50, 11, 0.2	**0.72**	0.81	0.39	0.51	1.06	1.76
50, 13, 0.1	**0.81**	0.88	0.51	0.59	0.83	3.16
50, 13, 0.2	**0.81**	0.93	0.52	0.64	0.93	3.01
450, 9, 0.1	**22.92**	56.86	6.00	17.17	24.33	14.72
450, 9, 0.2	**19.73**	56.90	6.04	18.81	20.63	14.18
450, 11, 0.1	**21.89**	59.32	6.62	19.69	35.04	18.18
450, 11, 0.2	**19.49**	59.94	6.76	21.98	25.48	18.48
450, 13, 0.1	**20.03**	57.94	7.09	19.76	59.18	24.59
450, 13, 0.2	**17.68**	58.74	7.31	20.23	58.90	25.09

challenging sensing conditions. These results indicate that the proposed hierarchical strategy provides a practical and scalable solution for localization in large, irregular, and measurement-limited unmanned sensing systems. Future work will consider extensions to dynamic network settings, heterogeneous sensing modalities, and learning-assisted estimation and refinement schemes.

Acknowledgments

This work was supported in part by the Beijing Natural Science Foundation under Grant 4244076, and in part by the National Natural Science Foundation of China (No.62372017) and Beijing Natural Science Foundation (No. L241049).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] He Y, Wang W, Mottola L, Li S, Sun Y, Li J, et al. Acoustic Localization System for Precise Drone Landing. IEEE Transactions on Mobile Computing. 2024;23(5):4126-44.

[2] Tong F, Ding B, Zhang Y, He S, Peng Y. A Single-Anchor Mobile Localization Scheme. IEEE Transactions on Mobile Computing. 2024;23(1):56-69.

[3] Shan F, Zeng J, Li Z, Luo J, Wu W. Ultra-wideband swarm ranging. In: IEEE INFOCOM 2021-IEEE Conference on Computer Communications. IEEE; 2021. p. 1-10.

[4] Ma C, Wu B, Poslad S, Selviah DR. Wi-Fi RTT ranging performance characterization and positioning system design. IEEE Transactions on Mobile Computing. 2020;21(2):740-56.

[5] Liu Q, Bai X, Gan X, Yang S. Lora RTT ranging characterization and indoor positioning system. Wireless Communications and Mobile Computing. 2021;2021(1):5529329.

[6] Morselli F, Razavi SM, Win MZ, Conti A. Soft Information-Based Localization for 5G Networks and Beyond. IEEE Transactions on Wireless Communications. 2023;22(12):9923-38.

[7] Conti A, Mazuelas S, Bartoletti S, Lindsey WC, Win MZ. Soft information for localization-of-things. Proceedings of the IEEE. 2019;107(11):2240-64.

[8] Yang Y, Chen M, Blankenship Y, Lee J, Ghassemlooy Z, Cheng J, et al. Positioning using wireless networks: Applications, recent progress and future challenges. IEEE Journal on Selected Areas in Communications. 2024.

[9] Zeng G, Mu B, Shi L, Chen J, Wu J. Consistent and asymptotically efficient localization from range-difference measurements. IEEE Transactions on Information Theory. 2023;70(4):3032-45.

[10] Win MZ, Shen Y, Dai W. A theoretical foundation of network localization and navigation. Proceedings of the IEEE. 2018;106(7):1136-65.

[11] Ping H, Wang Y, Li D, Chen W. Understanding node localizability in barycentric linear localization. IEEE/ACM Transactions on Networking. 2022;31(3):1353-68.

[12] Ping H, Wang Y, Li D, Sun T. Flipping free conditions and their application in sparse network localization. IEEE Transactions on Mobile Computing. 2020;21(3):986-1003.

[13] Fang X, Wang C, Nguyen TM, Xie L. Graph optimization approach to range-based localization. IEEE Transactions on Systems, Man, and Cybernetics: Systems. 2020;51(11):6830-41.

[14] So AMC, Ye Y. Theory of semidefinite programming for sensor network localization. Mathematical Programming. 2007;109(2):367-84.

[15] Korkmaz S, van der Veen AJ. Robust localization in sensor networkswith iterative majorization techniques. In: 2009 IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE; 2009. p. 2049-52.

[16] Zhang L, Liu L, Gotsman C, Gortler SJ. An as-rigid-as-possible approach to sensor network localization. ACM Transactions on Sensor Networks (TOSN). 2010;6(4):1-21.

[17] Traag VA. Faster unfolding of communities: Speeding up the Louvain algorithm. Physical Review E. 2015;92(3):032801.

[18] Guo Q, Zhang Y, Lloret J, Kantarci B, Seah WK. A localization method avoiding flip ambiguities for micro-UAVs with bounded distance measurement errors. IEEE Transactions on Mobile Computing. 2018;18(8):1718-30.

[19] Sun H, Yang S. Range-free localization algorithm based on modified distance and improved black-winged kite algorithm. Computer Networks. 2025;259:111091.