

# An Infrastructure-less, Control-Independent Solution to Relative Localisation of a Team of Mobile Robots using Ranging Measurements

Paolo Golinelli *Student Member IEEE*, Tommaso Faraci *Student Member IEEE*,  
Daniele Fontanelli *Fellow Member IEEE*

**Abstract**—The ability to localise teams of robots is essential for applications ranging from robotic fleets in unstructured environments to cooperative control and navigation tasks. In such contexts, fixed infrastructure is often unavailable, deployments must be fast and flexible, and system requirements must be minimal. We present a decentralised cooperative localisation algorithm that addresses all these challenges at once. The method is anchor-less, fully decentralised, and, unlike most existing approaches, does not require controlling the robots motion to ensure team observability. It relies only on local odometry, sparse inter-agent ranging measurements, and short-range communication, all of which are widely available in practice. The algorithm adopts a multi-hypothesis Bayesian framework that maintains the entire set of feasible solutions, ensuring robustness under transient unobservable conditions. Moreover, through information sharing, each agent benefits from the estimates of the entire group, even in partially connected conditions.

## I. INTRODUCTION

Many advanced robotic applications require reliable localisation of large groups of agents [1], [2]. Tracking fleets of robots is not only of industrial interest for achieving high levels of automation, but also plays a key role in unstructured and dynamic environments, as in coordinated exploration and mapping [3], [4]. In such scenarios, fixed infrastructure cannot be used to reduce deployment and management costs, not to mention potentially cumbersome calibration procedures.

The increasing affordability and availability of sensing technologies have made robust localisation a fundamental component in the development of multi-agent systems. In this context, methods that impose few system assumptions and maintain scalability are especially appealing [5], [6], with Ultra-Wideband (UWB) based approaches representing a prominent example [7]. These qualities are essential in most distributed systems to ensure broad applicability across different platforms. Moreover, such properties significantly reduce deployment efforts and simplify setup procedures, which is especially desirable in the development of applications designed for untrained or inexperienced users.

**Related work:** Numerous studies on cooperative localisation using ranging measurements rely on the availability of fixed anchor points [8], [9], commonly referred to as *anchors*, which are fixed reference nodes with known positions used to guarantee system observability [10], [11]. While effective

in controlled environments, the use of anchors is unsuitable in unstructured and dynamic scenarios, where they cannot be deployed or maintained. This is even more evident considering that in the anchor-based systems, when measurements are not retrieved simultaneously, trilateration usually admits multiple solutions [12]. In [13], a complete analysis of the conditions under which a unique trajectory exists when retrieving sparse distance measurements is established. The results show that, for any anchor configuration, there always exist pathological conditions in which no sufficiently high number of measurements and/or anchors can avoid robot trajectory indistinguishability, leaving motion control as the only means to enforce observability. While those analyses focus on anchor-based systems, their results can be applied to multi-agent scenarios by replacing the anchors with instances of positions of a second agent, thus showing that cooperative anchor-less localisation generally admits multiple solutions.

Hence, to guarantee the multi-agent observability, existing solutions rely on a richer measurement system that provides higher degrees of information. For example, several works adopt sensors capable of retrieving both distance and angle measurements [14], [15]. Another family of approaches imposes stringent requirements on agent motion. For instance, those relying on active motion control include strategies that maintain a subset of entities stationary while others are allowed to move [16], [17]. In more general terms, observability-aware formation control has been explored as a means of overcoming these issues [18], [19]. This research field is tailored to the development of control algorithms that generate motion patterns that maximise the information gained from motion-coupled observations, both to ensure observability and to improve the estimate accuracy. While these methods are proven to be effective, they usually degrade the control performance, since part of the control effort must be spent to guarantee the system observability rather than solving the desired task [20].

**Paper contributions:** In this work, we propose a robust and scalable algorithm for multi-agent 2D localisation that operates under minimal requirements. The method is anchor-less, fully decentralised, and explicitly accounts for measurement uncertainty. Unlike most existing approaches, it does not require external infrastructure, nor control over the agents' motion, making it suitable for heterogeneous systems. The algorithm relies only on local odometry, sparse inter-agent distance measurements (e.g., UWB-based), and communication with agents within the network, all of which are realistic and widely available capabilities in distributed

P. Golinelli and D. Fontanelli are with the Department of Industrial Engineering, University of Trento, Trento, Italy. {name.surname}@unitn.it.

T. Faraci is with the Department of Information Engineering and Computer Science, University of Trento, Trento, Italy. {name.surname}@unitn.it.

robotics. As numerous studies have shown, under such minimal requirements, there is no guarantee of system observability. Rather than imposing observability through restrictive control assumptions or additional infrastructure, our approach embraces this limitation by adopting a fundamentally different philosophy. Instead of aiming for uniqueness, the algorithm leverages only the available data and maintains the entire set of feasible solutions. We consider this paradigm shift to be a key novelty compared to existing methods.

The proposed solution is a Multi-Hypothesis Bayesian-based Decentralised Cooperative Localisation (MHDCL) algorithm, inspired by the Interim Master Decentralised Cooperative Localisation (IMDCL) framework [21], [22]. The core idea is to represent the relative poses of agents through multiple hypotheses, sufficient to cover the space of possible configurations, and to update this set collaboratively as measurements arrive. This multi-hypothesis framework offers two main advantages: (i) it enables localisation of partially connected networks of agents, whenever conditions permit, by combining information from the entire group; (ii) it naturally handles the ill-posedness of the problem by maintaining all feasible solutions consistent with the current measurement set, thereby representing and quantifying the lack of information.

Naturally, these benefits come with trade-offs. First, handling a large number of hypotheses causes significant computational and storage effort. Second, since the algorithm may output a set of possible solutions rather than a single estimate, downstream applications must be designed to accommodate this non-uniqueness.

Regarding practical requirements, the algorithm's assumptions are easily satisfied in real-world applications, as robots are typically equipped with odometry or IMU-based feedback loops. Inter-agent distances can be measured with radio-based ranging technologies such as UWB, which often also provide communication capabilities.

The rest of the paper is organised as follows. Section II introduces the problem and presents the necessary models. Section III provides a detailed description of the algorithm, while Section IV discusses practical aspects and the trade-off between computational cost and estimation performance. Section V presents experimental results in realistic case studies, and Section VI concludes the work and outlines future research directions.

Multimedial material illustrating the proposed approach and its experimental validation is available online.<sup>1</sup>

## II. BACKGROUND AND PROBLEM FORMULATION

A fleet of  $n_{\text{ag}}$  agents is deployed in a 2D environment as depicted in Figure 1. Agents are assumed to have access to local odometry information and to obtain occasional inter-agent distance measurements through ranging technologies such as UWB. Moreover, we assume that all agents can communicate with the entire group at any time. The goal for

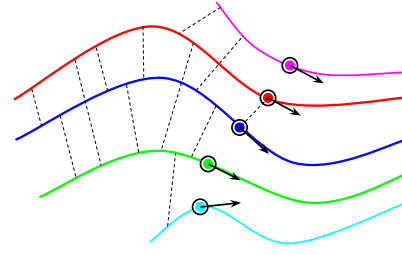


Fig. 1. Illustration of the multi-agent system during operation. The agents traverse the environment while collecting inter-agent distance measurements (dashed lines).

each agent running the decentralised algorithm is to estimate the relative poses of all other agents with respect to its own local reference frame.

### A. Notation

Matrices and vectors are denoted as bold letters (e.g.,  $\mathbf{A}$ ,  $\mathbf{q}$ ), whereas scalar quantities and functions use non-bold notation. The block diagonal matrix of the set  $\mathbf{A}_1, \dots, \mathbf{A}_n$  is written as  $\text{diag}(\mathbf{A}_1, \dots, \mathbf{A}_n)$ . The symbol  $\mathbb{R}^n$  denotes the  $n$ -dimensional Euclidean space.  $\mathbf{0}_{n \times m}$  defines the zero matrix of size  $n \times m$ , and  $\mathbf{I}_n$  is the identity matrix of size  $n$ . A planar rotation is described by the rotation matrix

$$\mathbf{R}(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

Random variables are specified by their distribution. In particular,  $\mathcal{N}(\mu, \sigma^2)$  denotes a Gaussian with mean  $\mu$  and variance  $\sigma^2$ , and  $\mathcal{U}(a, b)$  a continuous uniform distribution defined over the interval  $[a, b]$ . The expectation of a random variable  $\mathbf{x}$  is written as  $\mathbb{E}[\mathbf{x}]$ . Estimates are characterised by a state vector  $\mathbf{q} \in \mathbb{R}^n$  and its corresponding error covariance  $\mathbf{P} \in \mathbb{R}^{n \times n}$ . Superscripts indicate both the type of estimate and the agent relationship:  $p$  and  $c$  denote particle and cluster indices, respectively; the first index refers to the agent being estimated, while an optional second index, placed after a comma, denotes the agent maintaining the estimate. For example,  $\mathbf{q}^{p,i,j}$  represents the  $p$ -th particle estimating the state of agent  $i$ , as maintained by agent  $j$ . A superscript “-” indicates a propagated (predicted) state.

### B. System models

Each agent  $a \in \{1, \dots, n_{\text{ag}}\}$  is modelled using a discrete time model. Without loss of generality, in this paper, agents are modelled as unicycles: their poses at discrete time step  $k$  are fully defined by the set of  $n = 3$  states  $\mathbf{q}_k^a = [x_k^a, y_k^a, \theta_k^a]^\top$ , and their dynamics are described by the following equations of motion

$$\mathbf{q}_{k+1} = f(\mathbf{q}_k, \mathbf{u}_k) = \begin{bmatrix} x_k + \cos(\theta_k)v_k\Delta t \\ y_k + \sin(\theta_k)v_k\Delta t \\ \theta_k + \omega_k\Delta t \end{bmatrix}, \quad (1)$$

where  $\mathbf{u}_k = [v_k \ \omega_k]^\top \in \mathbb{R}^m$ ,  $m = 2$ , are the noisy model inputs and  $\Delta t$  is the time-step duration. Input uncertainty is assumed to be white, Gaussian and zero-mean, with

<sup>1</sup>Video available at: <https://www.youtube.com/watch?v=owNXrBRTyX4>

covariance matrix  $\mathbf{Q}_k \in \mathbb{R}^{m \times m}$ . Since the motion model is non-linear, its Jacobians are computed following the concepts defined by the Extended Kalman Filter (EKF) [23]

$$\begin{aligned} \mathbf{A}_k &= \frac{\partial f(\mathbf{q}_k, \mathbf{u}_k)}{\partial \mathbf{q}_k} = \begin{bmatrix} 1 & 0 & -\sin(\theta_k)v_k\Delta t \\ 0 & 1 & \cos(\theta_k)v_k\Delta t \\ 0 & 0 & 1 \end{bmatrix}, \\ \mathbf{G}_k &= \frac{\partial f(\mathbf{q}_k, \mathbf{u}_k)}{\partial \mathbf{u}_k} = \begin{bmatrix} \cos(\theta_k)\Delta t & 0 \\ \sin(\theta_k)\Delta t & 0 \\ 0 & \Delta t \end{bmatrix}. \end{aligned} \quad (2)$$

Ranging measurements between agent  $i$  and  $j$  at time  $k$  are modelled as  $z_k^{ij} = h(\mathbf{q}_k^i, \mathbf{q}_k^j) + \epsilon_k^{ij}$  with likelihood function

$$h(\mathbf{q}_k^i, \mathbf{q}_k^j) = \sqrt{(x_k^i - x_k^j)^2 + (y_k^i - y_k^j)^2}. \quad (3)$$

Measurements are assumed to be affected by a zero-mean, white, Gaussian noise  $\epsilon_k^{ij} \sim \mathcal{N}(0, \sigma_m^2)$ . The measurement model is also non-linear and is therefore linearised as follows

$$\mathbf{H}^{ij} = \frac{\partial h(\mathbf{q}_k^i, \mathbf{q}_k^j)}{\partial (\mathbf{q}_k^i, \mathbf{q}_k^j)} = \frac{1}{h(\mathbf{q}_k^i, \mathbf{q}_k^j)} [d_{xy} \quad -d_{xy}], \quad (4)$$

where  $d_{xy} = [x_k^i - x_k^j \quad y_k^i - y_k^j \quad 0]$ .

To simplify the notation, when one of the two states is null ( $\mathbf{0}_{1 \times 3}$ ), the null vector is neglected  $h(\mathbf{q}_k^i) = \sqrt{x_k^i{}^2 + y_k^i{}^2}$ , and its corresponding Jacobian is simply

$$\mathbf{H}^i = \frac{1}{h(\mathbf{q}_k^i)} [x_k^i \quad y_k^i \quad 0]. \quad (5)$$

### III. MHDCL: A MULTI-HYPOTHESIS DECENTRALISED COOPERATIVE LOCALISATION ALGORITHM

Each agent  $a \in \{1, 2, \dots, n_{\text{ag}}\}$  performs a decentralised algorithm based on a particle filter to estimate the possible relative poses of the other agents given sparse distance measurements  $z_{k+1}^{ij}$ , obtained between two agents ( $i$  and  $j$ ) at time  $k+1$ . Each agent uses its own reference frame for the poses of the others, so its position is arbitrarily located at the origin, with the heading pointing towards the  $x$  axis. The following section presents a detailed description of the algorithm, combining theoretical explanations with the corresponding mathematical formulations.

#### A. Initialisation

Initially, each agent  $a \in \{1, 2, \dots, n_{\text{ag}}\}$  sets its motion vector to zero, providing an initialisation for propagating future motion estimates

$$d\mathbf{q}_0^a = \mathbf{0}_n, \quad d\mathbf{P}_0^a = \mathbf{0}_{n \times n}, \quad d\Phi_0^a = \mathbf{I}_n \quad (6)$$

These quantities are only known by the agent they represent.

#### B. Prediction step

At each time step, every agent  $a \in \{1, 2, \dots, n_{\text{ag}}\}$  uses the equation of motion (1), and its Jacobians (2) to predict its own motion based on odometry data. This prediction updates each agent's motion vector and the associated uncertainty as follows

$$\begin{aligned} d\mathbf{q}_{k+1}^a &= f(d\mathbf{q}_k^a, \mathbf{u}_k^a), \quad d\Phi_{k+1}^a = \mathbf{A}_k^a d\Phi_k^a, \\ d\mathbf{P}_{k+1}^a &= \mathbf{A}_k^a d\mathbf{P}_k^a \mathbf{A}_k^{a\top} + \mathbf{G}_k^a \mathbf{Q}_k \mathbf{G}_k^{a\top}. \end{aligned} \quad (7)$$

#### C. Measurement Handling

At time step  $k+1$ , if a measurement  $z_{k+1}^{ij}$  is available between agents  $i$  and  $j$ , they either initialise their particle sets (in the case of the first measurement) or update their existing estimates. Without loss of generality, the update procedure is described from the perspective of agent  $i$ , with agent  $j$  considered as the counterpart. The same steps are simultaneously performed by agent  $j$  with reversed roles, assuming that  $z_{k+1}^{ji} = z_{k+1}^{ij}$ .

1) *Initialisation at first measurement:* If  $z_{k+1}^{ij}$  is the first measurement retrieved between agent  $i$  and  $j$ , the receiving agent initialises a set of  $n_p$  particles representing hypotheses for the other agent's pose, distributed according to the measurement uncertainty, as described next:

- 1: **for** each particle  $p, j, i \in \{1, \dots, n_p\}$  **do**
- 2: Draw  $\alpha \sim \mathcal{U}(-\pi, \pi)$ ,  $\delta_p \sim \mathcal{N}(0, \sigma_m^2)$
- 3: Compute position

$$\begin{bmatrix} x^{pj,i} \\ y^{pj,i} \end{bmatrix} = (z_{k+1}^{ij} + \delta_p) \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix}$$

- 4: Draw heading:  $\theta^{pj,i} \sim \mathcal{U}(-\pi, \pi)$
- 5: New particle's pose:  $\mathbf{q}^{pj,i} = [x^{pj,i} \quad y^{pj,i} \quad \theta^{pj,i}]^\top$
- 6: Initialise particle covariance matrix:  $\mathbf{P}^{pj,i} = \mathbf{0}_{n \times n}$
- 7: **end for**

2) *Update Step at subsequent measurements:* If  $z_{k+1}^{ij}$  is not the first measurement between agent  $i$  and  $j$ , then the particles from both agents can undergo an update, which eliminates low-probability hypotheses and redistributes particles in regions where the agent  $j$  is more likely to be located. Before computing the update, agent  $i$  needs to propagate the pose of each particle of  $j$  according to the movements that both agents performed since the previous measurement was taken. Hence, agents communicate to each other their motion vector and the associated uncertainty: for instance,  $(d\mathbf{q}_{k+1}^j, d\Phi_{k+1}^j, d\mathbf{P}_{k+1}^j)$  is sent from  $j$  to  $i$ , and vice-versa.

First, the pose and the covariance matrix of each particle of agent  $j$  held by the agent  $i$  are updated given the  $j$ -th agent motion vector  $d\mathbf{q}_{k+1}^j = [dx^j, dy^j, d\theta^j]^\top$  as

$$\begin{aligned} \tilde{\mathbf{q}}^{pj,i-} &= \mathbf{q}^{pj,i} + \mathbf{R}(\theta^{pj,i}) d\mathbf{q}_{k+1}^j, \\ \tilde{\mathbf{P}}^{pj,i-} &= \overline{d\Phi}^j \mathbf{P}^{pj,i} \overline{d\Phi}^{j\top} + \mathbf{R}(\theta^{pj,i}) d\mathbf{P}^j \mathbf{R}(\theta^{pj,i})^\top, \end{aligned} \quad (8)$$

where  $\overline{d\Phi}^j = \mathbf{R}(\theta^{pj,i}) d\Phi^j \mathbf{R}(\theta^{pj,i})^\top$ .

Then, their pose is roto-translated according to the  $i$ -th agent motion vector  $d\mathbf{q}_{k+1}^i = [dx^i, dy^i, d\theta^i]^\top$ , so that agent  $i$  is always located at the origin of its reference frame, pointing towards the  $x$  axis, i.e.

$$\begin{aligned} \mathbf{q}^{pj,i-} &= \mathbf{R}(-d\theta^i) (\tilde{\mathbf{q}}^{pj,i-} - d\mathbf{q}^i), \\ \mathbf{P}^{pj,i-} &= \mathbf{R}(-d\theta^i) (\overline{d\Phi}^i \tilde{\mathbf{P}}^{pj,i-} \overline{d\Phi}^{i\top} + \overline{d\mathbf{P}}^i) \mathbf{R}(-d\theta^i)^\top, \end{aligned} \quad (9)$$

where  $\overline{\mathbf{M}} = \mathbf{R}(\pi) \mathbf{M} \mathbf{R}(\pi)^\top$ , given a generic matrix  $\mathbf{M}$ . Once the particles have been propagated, an update of each hypothesis can be computed, considering the innovation

$$r^{pj,i} = h(\mathbf{q}^{pj,i-}, \mathbf{q}^i) - z_{k+1}^{ij} = h(\mathbf{q}^{pj,i-}) - z_{k+1}^{ij}, \quad (10)$$

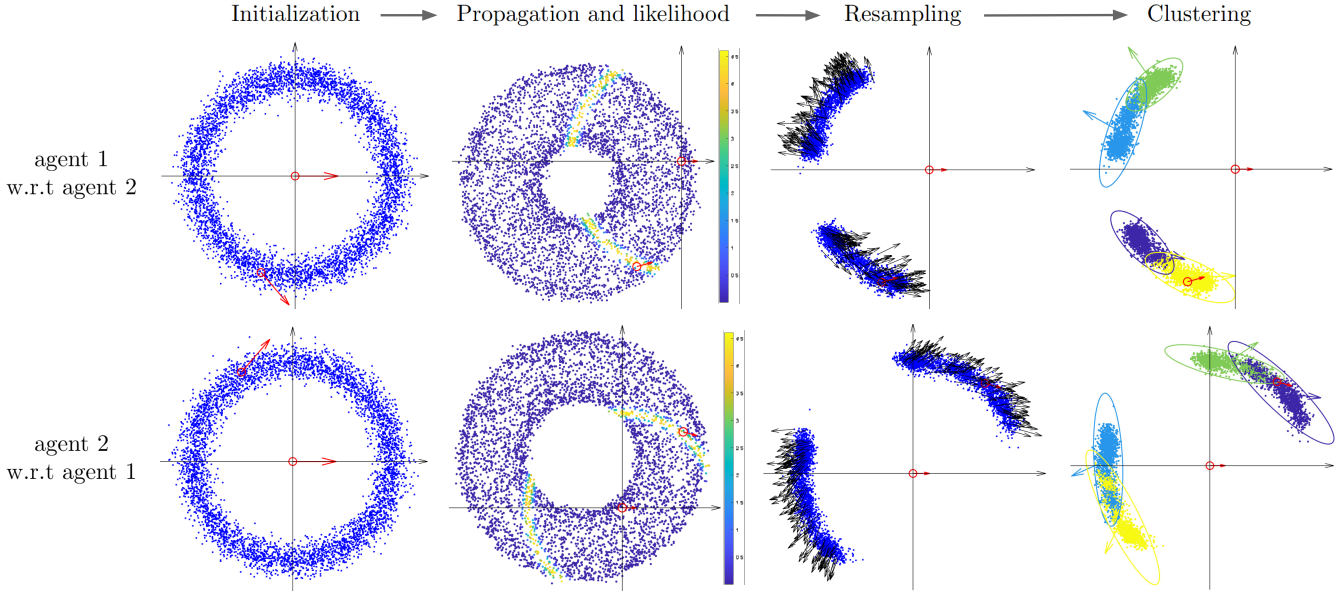


Fig. 2. Example of the algorithm's operation with two distance measurements between a pair of agents, leading to initialisation at first measurement, then update and clustering at the second measurement. Red arrowed circles indicate the true positions of the agents.

where  $h(\mathbf{q}^{pj,i-})$  is the distance measurement model defined in (3). Since  $h(\mathbf{q}^{pj,i-})$  is non-linear, the propagation of the prior state uncertainty  $\mathbf{P}^{pj,i-}$  into the measurement space is approximated by linearising  $h$  around the predicted state. The resulting Jacobian  $\mathbf{H}^{pj,i-}$ , defined in (5), linearly maps pose uncertainty into measurement uncertainty.

The innovation covariance, which accounts for both the propagated pose uncertainty and the sensor uncertainty variance  $\sigma_m^2$ , is given by

$$\begin{aligned} \sigma_r^{pj,i^2} &= \mathbb{E}[r^{pj,i^2}] = \mathbb{E}[(h(\mathbf{q}^{pj,i-}, \mathbf{q}^i) - z_{k+1}^{ij})^2] \\ &\approx \mathbf{H}^{pj,i-} \mathbf{P}^{pj,i-} \mathbf{H}^{pj,i- \top} + \sigma_m^2, \end{aligned} \quad (11)$$

which is then used to evaluate the likelihood of the actual measurement  $z^{ij}$  under each particle hypothesis  $\mathbf{q}^{pj,i}$ , as

$$\begin{aligned} w^{pj,i} &= \mathcal{N}(h(\mathbf{q}^{pj,i-}) | z^{ij}, \sigma_r^{pj,i^2}) \\ &= \frac{1}{\sqrt{2\pi}\sigma_r^{pj,i}} \exp\left(-\frac{(h(\mathbf{q}^{pj,i-}) - z^{ij})^2}{2\sigma_r^{pj,i^2}}\right), \end{aligned} \quad (12)$$

and, as customary for particle filters, the weights are normalised so that  $\sum_{p=0}^{n_p} w^{pj,i} = 1$ .

In order to remove low-probability hypotheses particles are resampled according to their weight, using the systematic approach [24], which is a low-variance method for selecting particles. First, the cumulative sum of the weights  $\mathbf{c} = \{c_1, \dots, c_{n_p}\}$  is computed, where  $c_m = \sum_{p=1}^m w^{pj,i}$ . A single random number is drawn  $u_b \sim \mathcal{U}(0, 1/n_p)$  to generate  $n_p$  equally spaced positions  $u_m = u_b + \frac{m-1}{n_p} \forall m = 1, \dots, n_p$ . For each  $u_m$ , the smallest  $\bar{m}$  is found such that  $c_{\bar{m}} \geq u_m$  and particle  $\mathbf{q}^{\bar{m}j,i}$  is selected.

For reducing particle degeneracy, after resampling, a small amount of Gaussian noise is added to each state component of the resampled particles

$$\mathbf{q}^{pj,i} = \mathbf{q}^{\bar{m}j,i} + [\epsilon_x \quad \epsilon_y \quad \epsilon_\theta]^\top, \quad (13)$$

where  $\epsilon_x, \epsilon_y \sim \mathcal{N}(0, \sigma_{rxy}^2)$  and  $\epsilon_\theta \sim \mathcal{N}(0, \sigma_{r\theta}^2)$ . This regularisation step promotes more uniform coverage of the state space and reduces the risk of particle collapse. The added noise standard deviations  $\sigma_{rxy}, \sigma_{r\theta}$  should be kept small relative to the process dynamics. In fact, excessive noise can overpower the information gained from measurements, vanishing the effect of the update. Resampled particles are assigned with zero uncertainty:  $\mathbf{P}^{pj,i} = \mathbf{0}_{n \times n}$ .

3) *Cluster the particles*: After being resampled, particles are clustered using a specialised Gaussian-von Mises Mixture Model (GVMMM) algorithm, which is a hybrid mixture model combining Gaussian distributions for describing the particle spatial arrangement, and von Mises distributions for describing the heading distribution. Each cluster represents a hypothesis, condensing the agglomerated particles into a single estimate, defined by the mean and covariance, thereby providing a compact representation that facilitates communication to other agents and reduces computational burdens.

The result of the GVMMM algorithm is a set of descriptors  $(\mathbf{q}^{cj,i}, \mathbf{P}^{cj,i}, w^{cj,i}, \kappa^{cj,i})$  for every cluster  $c \in \{1, \dots, n_{cj,i}\}$ . The auxiliary descriptor  $w^{cj,i}$  corresponds to the ratio of the cluster, which is a measure of how many particles it encloses w.r.t. the total number of particles, and  $\kappa^{cj,i}$  is the heading concentration parameter of the von Mises distribution. The number of clusters  $n_{cj,i}$  necessary to cover all the particles is selected using a Bayesian Information Criterion (BIC). Figure 2 illustrates how the steps described in the previous sections unfold for a pair of agents.

4) *Propagation of other agents*: Each time entity  $i$  is involved in a measurement, it applies a propagation of the particles corresponding to all other agents  $a \in \{1, \dots, n_{ag}\} \setminus \{i, j\}$  according to its own motion  $d\mathbf{q}_{k+1}^i = [dx^i, dy^i, d\theta^i]^\top$ , ensuring that estimates  $\mathbf{q}^{pa,i}, \mathbf{P}^{pa,i}$  are referred in the  $i$ -th

agent body-fixed frame, as in (9). To account for motion, the clusters are updated by recomputing their descriptors based on the propagated particles.

#### D. Collaborative updates

If agents  $i$  and  $j$  benefited from a measurement update, they can share their information, allowing other agents to improve their estimates, even if they were not directly involved in a measurement. This also enables sparsely connected agents to reconstruct the poses of the entire fleet. In fact, a well-defined cluster basically consists of a distance and angle measurement between agent  $i$  and  $j$ . Given this knowledge, the other agents can either perform an update of their existing hypotheses on  $i$  and  $j$ , or, if they are lacking an estimate for just one of the two, they can use it as an initialisation.

1) *Cluster sharing*: To let the other agents perform their update, agent  $i$  broadcasts its motion vector ( $\mathbf{d}\mathbf{q}_{k+1}^i$ ,  $\mathbf{d}\mathbf{P}_{k+1}^i$ ) and the descriptors of its clusters ( $\mathbf{q}^{cj,i}$ ,  $\mathbf{P}^{cj,i}$ ,  $w^{cj,i}$ ,  $\kappa^{cj,i}$ ), where  $c \in \{1, \dots, n_{cj,i}\}$ . In a specular manner, agent  $j$  broadcasts its knowledge. Given the received motion vectors, agents  $a \in \{1, \dots, n_{ag}\} \setminus \{i, j\}$  propagate their existing agent  $i$  particles  $\mathbf{q}^{pi,a}$ ,  $\mathbf{P}^{pi,a}$  as in (8). Cluster descriptors are recomputed to account for the propagation, and the procedure is repeated for the agent  $j$  particles and clusters.

2) *Initialisation at first measurement*: If agent  $a$  has never initialised particles either for agent  $i$  or  $j$  (not both), then it can use its preexisting hypotheses and the new  $i$ - $j$  measurement to initialise an estimate for the missing agent. For example, if agent  $a$  has an estimate for agent  $i$ , but not for  $j$ , it can use a Monte-Carlo approach for generating a set of possible poses for agent  $j$ , whose algorithmic steps are reported below.

- 1: **for** each particle  $p_j, a \in \{1, \dots, n_p\}$  **do**
- 2: Randomly select one particle  $pi, a \in \{1, \dots, n_p\}$
- 3: Randomly select one cluster  $c_j, i \in \{1, \dots, n_{cj,i}\}$
- 4: Generate a pose from the cluster distribution

$$\mathbf{q}_{ji} \sim \mathcal{N}(\mathbf{q}^{cj,i^-}, \mathbf{P}^{cj,i^-})$$

- 5: Compute new particle's pose with regularisation

$$\mathbf{q}^{pj,a} = \mathbf{q}^{pi,a^-} + \mathbf{R}(\theta^{pi,a^-})\mathbf{q}_{ji} + [\epsilon_x, \epsilon_y, \epsilon_\theta]^\top$$

$$\text{where } \epsilon_x, \epsilon_y \sim \mathcal{N}(0, \sigma_{rxy}^2) \text{ and } \epsilon_\theta \sim \mathcal{N}(0, \sigma_{r\theta}^2)$$

- 6: Initialise particle covariance matrix:  $\mathbf{P}^{pj,a} = \mathbf{0}_{n \times n}$
- 7: **end for**

For this procedure to yield reliable estimates that cover the entire set of feasible solutions, the prior particles and clusters used in the initialisation procedure must be sufficiently dense and not excessively dispersed.

3) *Update step for other agents*: If agent  $a$  already has an estimate of both agent  $i$  and  $j$ , then it can perform an update on each set of hypotheses. The update is soft, meaning that a likelihood is computed as the sum of probabilities of all the possible combinations. Since the computation of the likelihood for each particle of one agent against all the particles of the other agent would be computationally too demanding, particles are checked against clusters, significantly limiting the number of combinations.

To compute the likelihood between an estimate  $\mathbf{q}^{pi,a^-}$  of agent  $i$  and a cluster  $\mathbf{q}^{cj,a^-}$  of agent  $j$ , given the measured cluster  $\mathbf{q}^{cj,i}$ , two innovations are computed: one for the position and one for the heading, namely

$$r_p = h(\mathbf{q}^{pi,a^-}, \mathbf{q}^{cj,a^-}) - h(\mathbf{q}^{cj,i}), \quad (14)$$

$$r_h = (\theta^{cj,a^-} - \theta^{pi,a^-}) - \theta^{cj,i}. \quad (15)$$

The innovation is computed as a joint probability combining the consistency of the cluster distance with the two estimates and the agreement between particle and cluster headings. For simplicity, we assume the two likelihoods to be uncorrelated.

The uncertainty of the distance innovation is computed by linearising the non-linear function as in (4) and (5)

$$\begin{aligned} \sigma_{rp}^2 &= \mathbb{E}[(r_p)^2] = \mathbb{E}[(h(\mathbf{q}^{pi,a^-}, \mathbf{q}^{cj,a^-}) - h(\mathbf{q}^{cj,i}))^2] \\ &\approx \mathbf{H}^{picj,a^-} \mathbf{P}^{pi,cj} \mathbf{H}^{picj,a^- \top} + \mathbf{H}^{cj,i} \mathbf{P}^{cj,i} \mathbf{H}^{cj,i \top} \end{aligned} \quad (16)$$

where  $\mathbf{P}^{pi,cj} = \text{diag}(\mathbf{P}^{pi,a^-}, k_\sigma \mathbf{P}^{cj,a^-})$ , and  $k_\sigma$  is tunable to scale the cluster covariance to properly enclose the set of hypotheses. Assuming that innovations follow a Gaussian distribution, the likelihood is computed as

$$\begin{aligned} w_{rp} &= \mathcal{N}(h(\mathbf{q}^{pi,a^-}, \mathbf{q}^{cj,a^-}) | h(\mathbf{q}^{cj,i}), \sigma_{rp}^2) \\ &= \frac{1}{\sqrt{2\pi}\sigma_{rp}} \exp\left(-\frac{(h(\mathbf{q}^{pi,a^-}, \mathbf{q}^{cj,a^-}) - h(\mathbf{q}^{cj,i}))^2}{2\sigma_{rp}^2}\right). \end{aligned} \quad (17)$$

Instead, the heading innovation is modelled with a von Mises distribution. The innovation concentration factor is approximated by combining the angular dispersions of the three sources using the small-angle approximation  $\kappa \approx 1/\sigma^2$  [25], hence

$$\kappa_{rh} = \left( \frac{1}{\kappa^{cj,i}} + \sigma_{\theta^{pi,a^-}}^2 + \frac{1}{\kappa^{cj,a^-}} \right)^{-1}. \quad (18)$$

Therefore, the innovation likelihood can be defined as

$$\begin{aligned} w_{rh} &= \text{VM}((\theta^{cj,a^-} - \theta^{pi,a^-}) | \theta^{cj,i}, \kappa_{rh}) \\ &= \frac{1}{2\pi I_0(\kappa_{rh})} \exp(\kappa_{rh} \cos(\theta^{cj,a^-} - \theta^{pi,a^-} - \theta^{cj,i})), \end{aligned} \quad (19)$$

where  $I_0(\kappa)$  is the zero order modified Bessel function [25].

The compound probability is given by the product of the two likelihoods:  $w_r = w_{rp} w_{rh}$ . This computation is carried out for every measurement cluster  $c_j, i \in \{1, \dots, n_{cj,i}\}$  and for each estimate cluster  $ci, a \in \{1, \dots, n_{ci,a}\}$ . The resulting contributions are aggregated by weighted sum with the associated cluster weights  $w^{cj,i}$  and  $w^{ci,a}$ , respectively.

The whole procedure is repeated with the roles of agent  $i$  and  $j$  reversed. Finally, both sets of particles are redrawn using the systematic resampling procedure described in Section III-C.2. Each resampled particle inherits the cluster label of the particle it was drawn from, and the cluster descriptors are recomputed.

#### E. Final procedure

After sharing information with the others, agent  $i$  and  $j$  reset their motion vectors as in (6). Particles and clusters that did not undergo an update are carried forward unchanged.

#### IV. ALGORITHM PRACTICALITIES

Optimisation of computational burden is crucial, as the algorithm is intended for platforms with limited resources. In this section, we discuss heuristics and tunable parameters that help mitigate algorithmic limitations and facilitate deployment on resource-constrained devices.

##### A. Lightweight update for other agents

Due to its combinatorial structure, the update step for other agents described in Sec. III-D.3 is computationally demanding, as it requires evaluating all combinations of particles and clusters, leading to a complexity of  $\mathcal{O}(n_p \cdot n_{c_j,a} \cdot n_{c_j,i})$ . To mitigate this cost, a lightweight approximation of the update is introduced. Instead of evaluating the likelihood for every particle as in (17) and (19), the simplified approach computes the likelihood at the cluster level, resulting in a limited number of combinations  $\mathcal{O}(n_{c_i,a} \cdot n_{c_j,a} \cdot n_{c_j,i})$ , as follows

$$\sigma_{rp}^2 \approx \mathbf{H}^{ci,cj,a-} \mathbf{P}^{ci,cj-} \mathbf{H}^{ci,cj,a- \top} + \mathbf{H}^{cj,i} \mathbf{P}^{cj,i} \mathbf{H}^{cj,i \top},$$

$$w_{rp} = \frac{1}{\sqrt{2\pi}\sigma_{rp}} \exp\left(-\frac{(h(\mathbf{q}^{ci,a-}, \mathbf{q}^{cj,a-}) - h(\mathbf{q}^{cj,i}))^2}{2\sigma_{rp}^2}\right), \quad (20)$$

where  $\mathbf{P}^{ci,cj-} = \text{diag}(k_\sigma \mathbf{P}^{ci,a-}, k_\sigma \mathbf{P}^{cj,a-})$ , and

$$\kappa_{rh} = \left( \frac{1}{\kappa^{cj,i}} + \frac{1}{\kappa^{ci,a-}} + \frac{1}{\kappa^{cj,a-}} \right)^{-1},$$

$$w_{rh} = \frac{1}{2\pi I_0(\kappa_{rh})} \exp(\kappa_{rh} \cos((\theta^{cj,a-} - \theta^{ci,a-}) - \theta^{cj,i})). \quad (21)$$

The resulting cluster-level probabilities are then projected back onto the particles according to the cluster responsibilities obtained from the Mixture-Model algorithm. This simplified procedure significantly reduces computational complexity while preserving the main structure of the update, at the cost of some resolution.

##### B. Tunable parameters and limitations

The performance and scalability of the algorithm are strongly influenced by the choice of the design parameters. The most relevant are discussed below.

a) *Number of particles*: The number of particles  $n_p$  directly controls the granularity of the representation of the state space. Too few particles may cause loss of feasible hypotheses, particularly when agents travel long distances between measurements. This effect is amplified when agents move fast, when measurements are sparse, or when the number of agents is large, reducing the effective measurement frequency per agent.

b) *Timeliness of estimates*: A limitation inherited from [22] is the *staleness* of estimates. Since agents exchange their motion vectors only when participating in a measurement, the hypotheses of other agents are estimates of their state at the time of the last measurement and can quickly become outdated. Increasing measurement frequency or improving network connectivity can partially mitigate this issue.

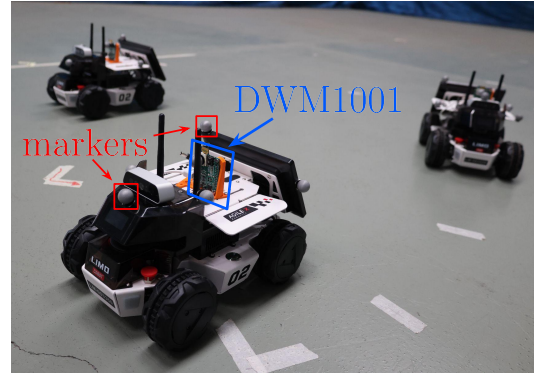


Fig. 3. LIMO mobile robots equipped with a DWM1001 UWB module for inter-agent ranging.

c) *Maximum number of clusters*: The maximum number of clusters  $n_c^{\max}$  controls a trade-off between accuracy and computational cost. The BIC-based selection guarantees the selection of a simple, yet good-fitting model, but as  $n_c^{\max}$  increases, so does computation time. In practice, performance does not significantly improve above a certain number of clusters, so  $n_c^{\max}$  should be tuned to match the signal-to-noise characteristics of the configuration. For example, when distances are large and uncertainty is low, the annulus defined by the likelihood function is thin and can benefit from a larger number of clusters, while when measurement noise is high relative to distance, a smaller number of clusters is sufficient.

#### V. EXPERIMENTAL VALIDATION

We evaluate the proposed algorithm's in three case studies. Each case highlights different conditions of observability and connectivity representative of practical deployments. The experiments are conducted with multiple LIMO differential-drive mobile robots. As shown in Figure 3, each robot is equipped with a DWM1001 UWB module (Qorvo) for inter-agent ranging and mounts reflective markers enabling tracking by an OptiTrack motion capture system, which provides the ground-truth trajectories for validation.

The algorithm's performance is evaluated against this ground truth using three main metrics:

- Probability of the real pose: the likelihood that the estimated clusters assign to the true pose of each agent.
- Error to the most probable estimate: the distance between the real pose and the most likely cluster.
- Cluster area: the total area covered by the clusters at  $3\sigma$ , used as an indicator of estimate uncertainty.

These metrics describe how well the estimates match reality, how precise the reconstruction is, and how the lack of information evolves over time.

##### A. Scenario 1 - Weakly observable motion

Three agents move along curved paths while keeping roughly constant relative poses, resembling the behaviour of three individuals walking together. As can be seen from the

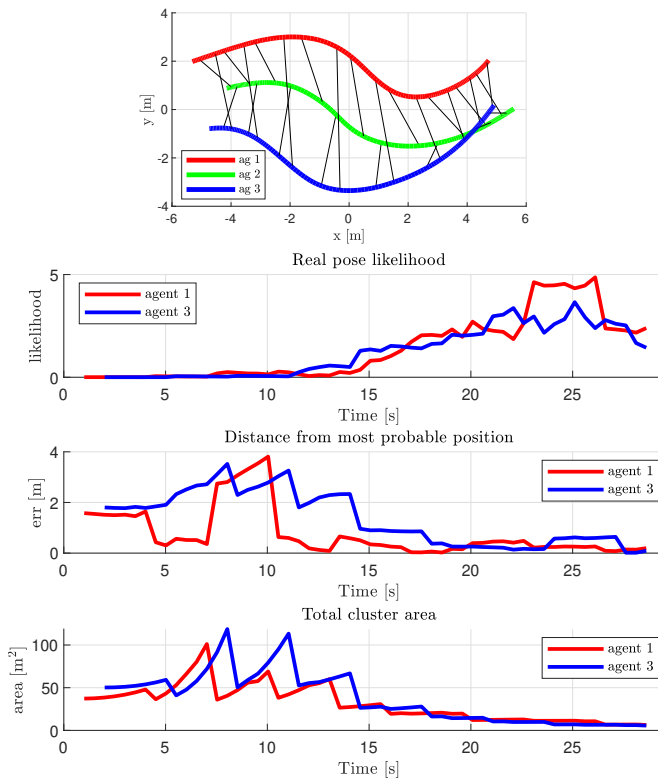


Fig. 4. Experimental results for Scenario 1. The top panel illustrate the ground-truth agent trajectories and inter-agent distance measurements, while the bottom panels present the evaluation metrics obtained from agent 2.

results depicted in Figure 4, the algorithm successfully reconstructs consistent estimates despite the measurements being nearly collinear, which are unobservable configurations [13]. This experiment highlights the ability of the algorithm to handle weakly observable configurations, which are typical of human-robot interaction applications such as coordinated walking or running.

### B. Scenario 2: Information transfer

In the second scenario, two agents move in parallel trajectories while a third performs a sinusoidal motion. The third agent trajectory introduces sufficient diversity in the measurements to resolve the otherwise unobservable configuration of the two parallel agents. As a result, accurate estimates are obtained between all pairs of agents (Figure 5). This demonstrates how informative motion from a subset of agents can propagate through the system, benefiting the localisation of the entire group.

### C. Scenario 3 - Partially connected fleet

The third scenario considers a fleet of five agents with partial connectivity: agents 1, 2 and 3 are fully connected, agent 4 collects only relative ranging from agent 3, and agent 5 only from agent 4. While the first two scenarios are validated on real robotic platforms, this larger-scale configuration is evaluated in simulation due to practical considerations.

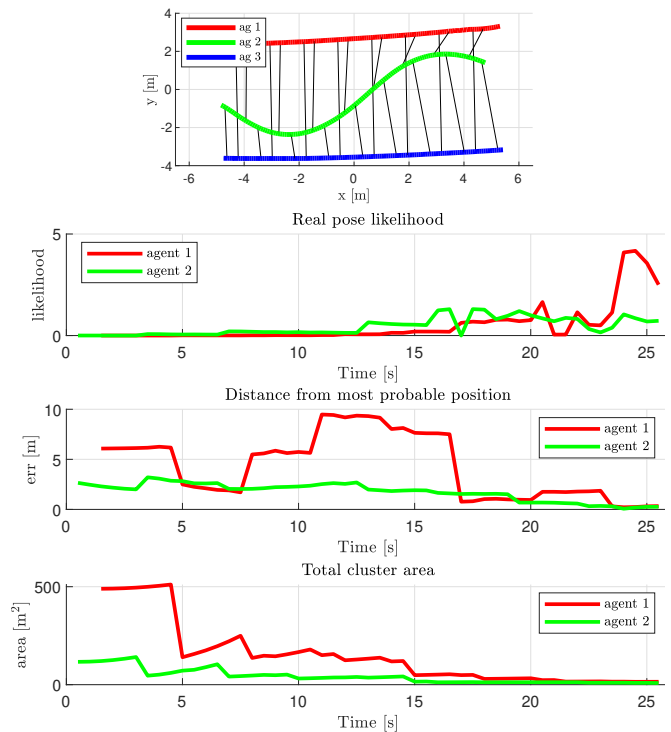


Fig. 5. Experimental results for Scenario 2. The top and bottom panels have the same description as of Figure 4, and the metrics refer to agent 3.

Despite the lack of global connectivity, the information gradually propagates across the network, enabling each agent to maintain reliable estimates with all the others. Notably, as shown in Figure 6, even agent 5, which has no direct measurements to agents 1, 2 and 3, is able to estimate their poses with good accuracy. This result illustrates the scalability of the algorithm to larger, partially connected networks.

## VI. CONCLUSIONS

We presented a decentralised cooperative localisation algorithm that is robust to low-observability conditions by maintaining multiple hypotheses which capture the full set of feasible solutions. Thanks to its design, we were able to remove the requirement, common to previous approaches, of controlling the agent motion to achieve observability, thus enabling precise task execution without constraining robot behaviour for localisation purposes. Moreover, the method relies on a single sensor per agent and, through information sharing, every new measurement benefits the entire group, enabling agents to improve their estimates collectively.

Through a series of experiments and simulations, we demonstrated the algorithm's ability to preserve accurate estimates even in challenging scenarios, such as agent trajectories in nearly collinear configurations. Furthermore, we showed its ability to effectively propagate information across the group, even if agents are not fully connected, which is essential for achieving scalability.

Future work will focus on extending the applicability of the approach to other domains, including multi-rotor

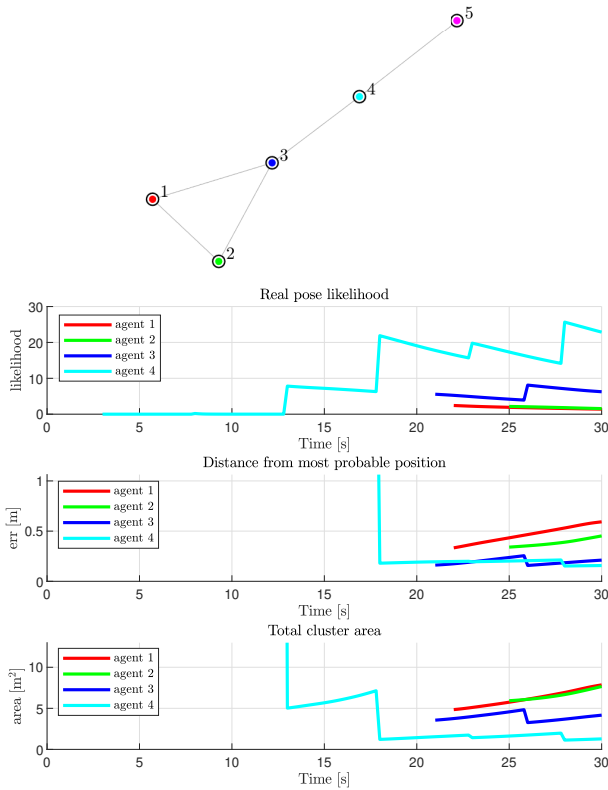


Fig. 6. Simulation results for Scenario 3. The top picture depicts the network connectivity graph. The bottom panels present evaluation metrics obtained from agent 5.

aerial robots, as well as humans equipped with wearable devices. On the algorithmic side, we will explore adaptive particle resampling strategies with noise injection tuned to the estimation uncertainty, as well as optimisations aimed at reducing computational complexity.

## REFERENCES

- [1] T.-M. Nguyen, A. Hanif Zaini, C. Wang, K. Guo, and L. Xie, "Robust target-relative localization with ultra-wideband ranging and communication," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 2312–2319.
- [2] V. Mai, M. Kamel, M. Krebs, A. Schaffner, D. Meier, L. Paull, and R. Siegwart, "Local positioning system using uwb range measurements for an unmanned blimp," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 2971–2978, 2018.
- [3] T. H. Nguyen, T.-M. Nguyen, and L. Xie, "Range-focused fusion of camera-imu-uwb for accurate and drift-reduced localization," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1678–1685, 2021.
- [4] T.-M. Nguyen, M. Cao, S. Yuan, Y. Lyu, T. H. Nguyen, and L. Xie, "Viral-fusion: A visual-inertial-ranging-lidar sensor fusion approach," *IEEE Transactions on Robotics*, vol. 38, no. 2, pp. 958–977, 2022.
- [5] A. Prorok, A. Bahr, and A. Martinoli, "Low-cost collaborative localization for large-scale multi-robot systems," in *2012 IEEE International Conference on Robotics and Automation*. Ieee, 2012, pp. 4236–4241.
- [6] A. Ahmad, G. Lawless, and P. Lima, "An online scalable approach to unified multirobot cooperative localization and object tracking," *IEEE transactions on robotics*, vol. 33, no. 5, pp. 1184–1199, 2017.
- [7] Z. Cao, R. Liu, C. Yuen, A. Athukorala, B. K. Kiat Ng, M. Mathanraj, and U.-X. Tan, "Relative localization of mobile robots with multiple ultra-wideband ranging measurements," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 5857–5863.
- [8] W. Shule, C. M. Almansa, J. P. Queralta, Z. Zou, and T. Westerlund, "Uwb-based localization for multi-uav systems and collaborative heterogeneous multi-robot systems," *Procedia Computer Science*, vol. 175, pp. 357–364, 2020.
- [9] M. Hamer and R. D'Andrea, "Self-calibrating ultra-wideband network supporting multi-robot localization," *Ieee Access*, vol. 6, pp. 22 292–22 304, 2018.
- [10] L. Santoro, D. Brunelli, and D. Fontanelli, "On-line Optimal Ranging Sensor Deployment for Robotic Exploration," *IEEE Sensors Journal*, vol. 22, no. 6, March 2022.
- [11] F. Shamsfakhr, L. Palopoli, and D. Fontanelli, "Minimising the number of ranging sensors verifying target positioning uncertainty," *Measurement*, vol. 211, p. 112666, 2023.
- [12] F. Riz, L. Palopoli, and D. Fontanelli, "Why three measurements are not enough for trilateration-based localisation," in *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2023, pp. 01–06.
- [13] —, "Analysis of indistinguishable trajectories of a nonholonomic vehicle subject to range measurements," *IEEE Transactions on Automatic Control*, vol. 70, no. 2, pp. 708–719, 2025.
- [14] F. Valdeira, C. Soares, and J. Gomes, "Maximum likelihood localization of a network of moving agents from ranges, bearings and velocity measurements," *Signal Processing*, vol. 221, p. 109471, 2024.
- [15] A. Prorok, A. Bahr, and A. Martinoli, "Low-cost collaborative localization for large-scale multi-robot systems," in *2012 IEEE International Conference on Robotics and Automation*, 2012, pp. 4236–4241.
- [16] R. Kurazume, S. Nagata, and S. Hirose, "Cooperative positioning with multiple robots," in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994, pp. 1250–1257 vol.2.
- [17] L. Santoro, M. Nardello, D. Brunelli, and D. Fontanelli, "Where-are-you: an uwb relative tracking system for pedestrian using only ranging information," in *2023 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*. Ieee, 2023, pp. 1–6.
- [18] H. Go, C. L. Chong, L. Qian, and H. H.-T. Liu, "Observability-aware control for cooperatively localizing quadrotor uavs," *arXiv preprint arXiv:2411.03747*, 2024.
- [19] C. Grebe, E. Wise, and J. Kelly, "Observability-aware trajectory optimization: Theory, viability, and state of the art," in *2021 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2021, pp. 1–8.
- [20] D. Coleman, S. D. Bopardikar, and X. Tan, "Observability-aware target tracking with range only measurement," in *2021 American Control Conference (ACC)*, 2021, pp. 4217–4224.
- [21] S. I. Roumeliotis and G. A. Bekey, "Distributed multirobot localization," *IEEE transactions on robotics and automation*, vol. 18, no. 5, pp. 781–795, 2002.
- [22] S. S. Kia, S. Rounds, and S. Martinez, "Cooperative localization for mobile agents: A recursive decentralized algorithm based on kalman-filter decoupling," *IEEE Control Systems Magazine*, vol. 36, no. 2, pp. 86–101, 2016.
- [23] S. Konatowski, A. Pieni *et al.*, "A comparison of estimation accuracy by the use of kf, ekf & ukf filters," *WIT Transactions on modelling and simulation*, vol. 46, 2007.
- [24] M. Bolic, P. Djuric, and S. Hong, "New resampling algorithms for particle filters," in *2003 IEEE International Conference on Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03)*, vol. 2, 2003, pp. II–589.
- [25] K. V. Mardia and P. E. Jupp, *Directional statistics*. John Wiley & Sons, 2009.