Interval-based cooperative UAVs pose domain characterization from images and ranges

Ide-Flore Kenmogne, Vincent Drevelle, Eric Marchand

HAL Id: hal-01853958
https://inria.hal.science/hal-01853958

Submitted on 6 Aug 2018

Interval-based cooperative UAVs pose domain characterization from images and ranges

Ide-Flore Kenmogne, Vincent Drevelle and Eric Marchand

Abstract— An interval-based approach to cooperative localization for a group of unmanned aerial vehicles (UAVs) is proposed. It computes a pose uncertainty domain for each robot, i.e., a set that contains the true robot pose, assuming bounded error measurements. The algorithm combines distances measurements to the ground station and between UAVs, with the tracking of known landmarks in camera images, and provides a guaranteed enclosure of the robots pose domains. Pose uncertainty domains are computed using interval constraint propagation techniques, thanks to a branch and bound algorithm. We show that the proposed method also provides a good point estimate, that can be further refined using non-linear iterative weighted least squares. Results are presented for simulated two-robots configurations, for experimental data, and compared with a classical Extended Kalman Filter.

I. INTRODUCTION

In the last few years, unmanned aerial vehicles (UAVs) have significantly attracted attention and interest in several fields of applications involving multi-robot tasks, like searching and coverage missions [1], cooperative manipulation [2], and camera based formation control [3]. It appears that a team of robots can perform more efficiently tedious tasks like exploration [4].

One of the problems to address with group of robots is cooperative localization [5], i.e., improvement of each robot positioning ability thanks to information exchange (e.g sensor measurements) with other robots of the group. Several methods are available for Cooperative Localization (CL), like the Extended Kalman Filter (EKF) for a centralized system [6], or, if computation is decentralized and communication is unreliable, other techniques like Covariance Intersection [7] or Interleaved Update [8]. Bounded error approaches using polytopes and linear programming have also been proposed [9], [10].

In a bounded error framework, interval methods [11] enable rigorous propagation of errors in non-linear problems. Applications are found in localization of mobile robots [12] or cars [13], and in the context of cooperative localization for underwater robots with sonar ranging [14], and vehicle networks with GNSS [15]. The main characteristic of these approaches is to provide a set a feasible positions, instead of a point estimate.

In a previous work [16], we proposed an interval-based set-membership approach to single robot pose domain characterization. In this paper, we extend this method to multi-robots cooperative localization. We consider a group of

Ide-Flore Kenmogne, Vincent Drevelle and Eric Marchand are with Univ Rennes, Inria, CNRS, IRISA {ide-flore.kenmogne, vincent.drevelle, eric.marchand}@irisa.fr.

UAVs equipped with cameras able to track landmarks and a communication system with ranging capabilities. We aim at computing an enclosure for the pose of each robot based on set-inversion and constraints propagation techniques. The method computes a set that contains all the feasible poses of each robot, by taking image measurements, distance measurements and landmark positions error bounds into account. The true pose is guaranteed to be inside the computed solution set, as long as the actual measurement errors stay within their bounds. The main contribution of this paper resides in:

- using interval methods for cooperative fusion of camera and range measurements for the localization of two UAVs
- combining the interval approach with non-linear weighted least squares for accurate initial localization
- providing a comparison with other existing classical approaches

The paper is organized as follows. Section II presents the localization problem with camera and ranges. Interval analysis and contractors are introduced in Section III. The proposed approach to compute pose domains is presented in Section IV. Finally, the approach is validated in Section V.

II. PROBLEM STATEMENT

This paper addresses the problem of localizing two UAVs in the world reference frame (Fig. 1), in a cooperative way, and with a focus on characterizing pose uncertainty domains. Given measurement uncertainties, we want to determine an enclosing domain that contains the pose of the robots.

Both robots are equipped with an altimeter, an inertial measurement unit (IMU) and a camera, so that each robot R_k, $k \in \{1, 2\}$ can measure:

- its altitude z_k
- its accurate roll ψ_k and pitch θ_k angles
- a rough estimate of the heading ϕ_k angle
- image points $\{\mathbf{x}_{i,k}, ..., \mathbf{x}_{N,k}\}$ corresponding to observable landmarks $\{{}^w\mathbf{X}_i, ..., {}^w\mathbf{X}_N\}$ in the environment

In addition to these embedded sensors measurements, each robot is capable of:

- measuring its distance $d_{k,j}$ to the other robot R_j
- measuring its distance d_k to the base station (the origin of the world frame in which the robots move)
- communicating its measurements to the other robot.

This can be done using an IEEE 802.15.4.a ultra-wide band (UWB) communication system [17] which enables ranging via time of flight measurements, or others external sensors.
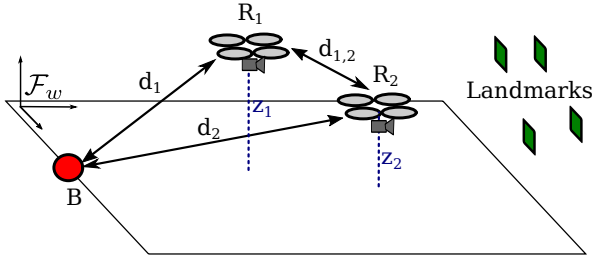
Fig. 1: Cooperative localization of two UAVs R_1 and R_2, seeing landmarks and measuring ranges to the base station B.

We aim at computing the domains of the feasible 6-DOF poses of each robot in the world reference frame $\mathcal{F}_w$, i.e., positions $\mathbf{p}_k = (x_k, y_k, z_k)$ and attitudes (roll, pitch and yaw) $\mathbf{q}_k = (\phi_k, \theta_k, \psi_k)$. As in [18], since we are operating in near-hovering regime, we assume that the roll and pitch (ϕ_k, θ_k) are measured with an onboard IMU. Moreover, it is possible to retrieve the altitude of each robot through an altimeter. Thus, most of the effort in this article will be focused on estimating the horizontal position x_k, y_k and the heading ψ_k of each robot.

A. Camera based pose estimation

Let $\mathcal{F}_w$ the world frame attached to the origin of a local coordinate system, $\mathcal{F}_{r_k}$ the robot frame attached to the UAV R_k center of gravity, and $\mathcal{F}_c$ the camera frame attached to the camera optical center. We aim at determining the pose $\mathbf{r}_k = (x_k, y_k, z_k, \phi_k, \theta_k, \psi_k)$ of each robot in the world frame. Considering a single robot, the indices k are omitted in the sequel of this subsection for clarity. Pose estimation amounts to estimating the transformation matrix ${}^r\mathbf{T}_w$ between the world reference frame $\mathcal{F}_w$ and each robot frame $\mathcal{F}_r$.

$$
{}^r\mathbf{T}_w = \begin{pmatrix} {}^r\mathbf{R}_w & {}^r\mathbf{t}_w \\ \mathbf{0}_{3\times1} & 1 \end{pmatrix} \tag{1}
$$

where ${}^r\mathbf{R}_w$ and ${}^r\mathbf{t}_w$ are respectively the rotation matrix depending on the UAV's orientation and the translation vector that represents the position of the UAV in the world reference frame.

Knowing that ${}^c\mathbf{T}_w = {}^c\mathbf{T}_r\,{}^r\mathbf{T}_w$, if we suppose that the rigid transformation ${}^c\mathbf{T}_r$ between the robot and the camera frame is known from calibration [19] (for each UAV), the pose can be obtained indirectly by estimating the transformation between the camera and the world frames.

To determine the pose of a camera in the world (as depicted in Fig. 2), the perspective projection equation of a 3-D point
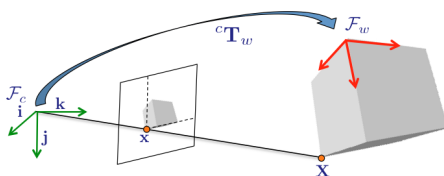

Fig. 2: Perspective projection of an object in the image plane

(in the world frame) on the image frame (a 2-D point) is used. This is the pinhole camera model, expressed by:

$$
\bar{\mathbf{x}} = \mathbf{K}\,\mathbf{\Pi}\,{}^c\mathbf{T}_w\,{}^w\mathbf{X} \tag{2}
$$

where:
- ${}^w\mathbf{X} = (X, Y, Z, 1)^\top$ are the homogeneous coordinates of a 3-D point in the world frame;
- $\bar{\mathbf{x}} = (u, v, 1)^T$ are the homogeneous coordinates (in pixel) of the projection of ${}^w\mathbf{X}$ in the image;
- $\mathbf{K}$ is the camera intrinsic parameters matrix expressed using p_x and p_y, the meter/pixel scale factor and u_0 and v_0 the image principal point's coordinates [20];
- $\mathbf{\Pi}$ is the perspective projection matrix.

The intrinsic parameters can be easily obtained through an off-line calibration step; we therefore consider them as known. If we have N points ${}^w\mathbf{X}_i$, $i = 1..N$ whose coordinates expressed in $\mathcal{F}_w$ are given by ${}^w\mathbf{X}_i = (X_i, Y_i, Z_i, 1)^\top$, the projection $\bar{\mathbf{x}}_i$ of these points in the image plane is then given by: $\bar{\mathbf{x}}_i = \mathbf{K}\mathbf{\Pi}\,{}^c\mathbf{T}_w\,{}^w\mathbf{X}_i$. Knowing 2D-3D point correspondences, $\bar{\mathbf{x}}_i$ and ${}^w\mathbf{X}_i$, pose estimation consists in solving the system of equations for ${}^c\mathbf{T}_w$. This is an inverse problem that is known as the Perspective from N Points problem or PnP. Many solutions exist to solve the PnP problem [21], [22]. The "gold-standard" consists in estimating the six parameters of the transformation ${}^c\mathbf{T}_w$ by minimizing the norm of the reprojection error, by using a non-linear minimization approach such as the Levenberg Marquardt technique. Without prior knowledge of the pose, 3 unaligned points lead to four ambiguous solutions. A minimum number $N \geq 4$ of points is thus needed to determine the pose.

B. Range measurements

Assuming the antenna coincides with the center of the UAV, range measurements are independent of the robot attitude. They only provide information about the robot position $\mathbf{p}_k = (x_k, y_k, z_k)$. Indeed, range measurement of robot R_k to the base station B is defined by

$$
d_k = \|\mathbf{p}_k - \mathbf{b}\|_2 \quad \text{with } k \in \{1, 2\},
$$

where $\mathbf{b} = (x_B, y_B, z_B)$ is the base station position in the world reference frame. Inter-robot range measurements are defined by

$$
d_{k,j} = \|\mathbf{p}_k - \mathbf{p}_j\|_2 \quad \text{with } (k, j) \in \{1, 2\}^2 \text{ and } k \neq j.
$$

Since the base station position $\mathbf{b}$ is known, the knowledge of d_k constrains the position of R_k on the sphere $S^2(\mathbf{b}, d_k)$. The additional knowledge of $d_{1,2}$ restricts the feasible configurations: for a given candidate position of R_1, the admissible positions of R_2 are restricted to a circle (intersection of the two spheres $S^2(\mathbf{b}, d_2)$ and $S^2(\mathbf{p}_1, d_{1,2})$).

Range measurements d_1, d_2 and $d_{1,2}$ provide constraints on the positions of the two robots, but there are 3 degrees of freedom left (out of 6) on the position components of the two-robot group (corresponding to all rotations of the formation around the base station).

Now let us consider that the altitude z_k of each robot is known (provided by the altimeter). This provides two

additional constraints. Distances d_k to the ground station now restrict the position of each robot on an horizontal circle (intersection of $S^2(\mathbf{b}, d_k)$ and the $z_k = z_k^{\text{meas}}$ plane). The inter-distance measurement $d_{1,2}$ constrains the feasible configurations such that for a given position of R_1, there are only two admissible positions of R_2. These corresponds to the intersections of the R_2 position circle with the sphere $S^2(\mathbf{p}_1, d_{1,2})$. In addition to this ambiguity, there is still one degree of freedom left for the two-robot position configuration, corresponding to rotations around the vertical axis of the base station.

C. Cooperative pose-domain characterization

As previously explained, in the proposed configuration (two robots and a fixed base station), ranging alone does not enable to do perform a position fix. It has to be combined with camera measurements. A robot is able to compute its pose by combining pitch and roll knowledge from the IMU, altitude measurement, distance measured to the base station, and vision of at least two landmarks. Seeing more landmarks enables to refine the pose thanks to data redundancy.

Cooperative positioning is enabled by inter-UAV ranging, and measurements exchange. It provides a means for position refining by using additional measurements. Cooperation also allows to relax the landmark visibility constraint, enabling one of the robots to see only one landmark and still estimate its pose (up to an ambiguity that can be resolved with a rough heading estimate).

In the sequel, we consider that measurements are uncertain, and are represented by intervals (bounded-error model). The solution to the localization problem is not a point pose estimate anymore, but a *pose domain*, i.e., a set. We now show how to compute a guaranteed enclosure of the two robots pose domains, by using interval analysis and constraint propagation methods.

III. INTERVAL ANALYSIS

This section presents a short overview of *Interval Analysis* as a non-probabilistic tool for bounded error estimation. It puts an emphasis on the use of Contractor Programming [23] and Set Inversion via Interval Analysis (SIVIA) [24] to perform set-membership estimation, i.e., computing the set of solutions that are compatible with the measurements and their error bounds.

A. Set Inversion

Let us denote $\mathbb{IR}$ the set of real intervals, and $\mathbb{IR}^n$ the set of n-dimensional *boxes* (interval vectors). Intervals and boxes will be denoted between brackets. The interval $[x] = [\underline{x}, \overline{x}]$ is defined as the set $\{x \in \mathbb{R}, \ \underline{x} \le x \le \overline{x}\}$, where $\underline{x}$ and $\overline{x}$ are respectively the lower and upper bounds of $[x]$. The width of an interval is $w([x]) = \overline{x} - \underline{x}$, and its midpoint (or center) is $mid([x]) = \frac{\overline{x} + \underline{x}}{2}$. The width of a box is the largest of the widths of its components, and the center of a box is the vector of the midpoints of its components.

With $\mathbf{f}$ a function from $\mathbb{R}^n$ to $\mathbb{R}^m$ and $\mathbb{Y}$ a subset of $\mathbb{R}^m$, *set-inversion* consists in characterizing the preimage of $\mathbb{Y}$, i.e., the set $\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n | \mathbf{f}(\mathbf{x}) \in \mathbb{Y}\} = \mathbf{f}^{-1}(\mathbb{Y})$.

Since computing with arbitrary sets is intractable in the general case, interval analysis treats problems by operating on interval and boxes. *Inclusion functions* are an important tool for interval computations. An inclusion function for $\mathbf{f}$ is a function from $\mathbb{IR}^n$ to $\mathbb{IR}^m$ such that

$$\forall [\mathbf{x}] \in \mathbb{IR}^n, \mathbf{f}([\mathbf{x}]) \subset [\mathbf{f}]([\mathbf{x}]).$$

If $\mathbf{f}$ admits a convergent inclusion function $[\mathbf{f}]$, then two subpavings (sets of non-overlapping boxes) $\mathbb{X}^-$ and $\mathbb{X}^+$ that characterize the set $\mathbb{X}$ such that $\mathbb{X}^- \subset \mathbb{X} \subset \mathbb{X}^+$ can be obtained with the SIVIA branch and bound algorithm.

B. Contractors

To characterize a set $\mathbb{X}$, SIVIA needs to bisect in all dimensions of the search space, which make the complexity grow exponentially with the dimension of the problem. Problems in higher dimensions can however still be addressed, by using *contractors* to reduce the search space.

A contractor is a mapping from $\mathbb{IR}^n$ to $\mathbb{IR}^n$ such that
- $\forall [\mathbf{x}] \in \mathbb{IR}^n, C([\mathbf{x}]) \subseteq [\mathbf{x}]$ (**contraction**)
- $(\mathbf{x} \in [\mathbf{x}], C(\mathbf{x}) = \mathbf{x}) \Rightarrow \mathbf{x} \in C([\mathbf{x}])$ (**consistency**)
- $C(\mathbf{x}) = \emptyset \Leftrightarrow (\exists \varepsilon > 0, \forall [\mathbf{x}] \subseteq B(\mathbf{x}, \varepsilon), C([\mathbf{x}]) = \emptyset)$ (**continuity**), where $B(\mathbf{x}, \varepsilon)$ is the ball centered on $\mathbf{x}$ with radius ε.

A contraction operation returns a sub-domain of the input domain $[\mathbf{x}]$, without losing any solution. For example if we have the equation $a = 2c + b$. Let $a \in [-2, 3]$, $b \in [1, 1]$ and $c \in [0, 2]$. The most straightforward way to contract with respect to numerical constraints is by using the *forward-backward* algorithm. The forward operation will give $a = [-2, 3] \cap 2 * [0, 2] + [1, 1] = [1, 3]$. The backward operation, based on applying the derived constraints $c = (a - b)/2$ and $b = a - 2c$, gives $c = [0, 1]$ and $b = [1, 1]$. If the box resulting from the contraction operation is equal to the smallest interval that contains the solution, the contractor is said to be *minimal*. The *forward-backward* contractor, also known as HC4 [25], is the classical constraint propagation loop found in the literature. It allows to contract with respect to a system of constraints, by propagating constraints in an optimal order, as shown in the example above. In this article, we use the contractor to refine our domain by rejecting the part that is non consistent with the measurements.

C. Set Inversion with Contractor

An outer subpaving of the solution set $\mathbb{X}$ contained in the initial domain $[\mathbf{x}_0]$ can be computed with the SIVIA algorithm (Alg. 1). Taking the initial box $[\mathbf{x}_0]$, SIVIA successively refines the box by applying contractions and bisections. The list $\mathfrak{L}$, which is a stack (LIFO) for depth first space exploration, is used as a working list in which bisected boxes are temporary stored. For each box, the contractor $\mathcal{C}_f$ is used to discard all parts of the box that are not consistent with the measurements (this is equivalent to applying the constraint $f(\mathbf{x}) \in [\mathbf{y}]$. The parameter ε is used to control the sharpness of $\mathbb{X}^+$. If the width of a contracted box is greater than ϵ, then the box is bisected and the two resulting boxes are enqueued in $\mathfrak{L}$.

Result: $\mathbb{X}^+$
$\mathbb{X}^+ := \emptyset$ *outer subpaving of pose domain*
$\mathfrak{L} \leftarrow \{[\mathbf{x}_0]\}$
while $\mathfrak{L} \neq \emptyset$ **do**
    $[\mathbf{x}] \leftarrow$ get the first box of $\mathfrak{L}$
    $[\mathbf{x}] \leftarrow \mathcal{C}_f([\mathbf{x}])$ *contract the box*
    if width$([\mathbf{x}]) < \epsilon$ **then**
        $\mathbb{X}^+ \leftarrow \mathbb{X}^+ \cup [\mathbf{x}]$
    else if $[\mathbf{x}] \neq \emptyset$ **then**
        $([\mathbf{x}_1], [\mathbf{x}_2]) \leftarrow$ bisect$([\mathbf{x}])$
        put $[\mathbf{x}_1]$ in $\mathfrak{L}$
        put $[\mathbf{x}_2]$ in $\mathfrak{L}$
    end
end

Algorithm 1: SIVIA($[\mathbf{x}_0], \mathcal{C}_f, \epsilon$)

IV. PROPOSED APPROACH

To solve the problem described in Section II, we define our problem as a Constraint Satisfaction Problem (CSP), and then use SIVIA with contractors (Alg. 1) to characterize the feasible poses domain for each robot. So in this section, we present:

- the constraints related to the projection of points in the camera frame
- the constraints related to the measured ranges (UAV to UAV and UAV to base ranges)
- the proposed strategy used to seek the domain of all the feasible poses.

The width of this domain will quantify the uncertainty on the pose with respect to that of the measurements.

A. Camera constraints

For each landmark seen by the camera of a robot, the projection equation (Eq. 2) and the fact that the camera cannot see from the back give the following constraints:

- Two constraints C_{proj} w.r.t. the 2D-3D correspondences. With each landmark point represented in homogeneous coordinates, the landmarks coordinates in the camera frame are given by

$$(^cX, {^c}Y {^c}Z, 1)^T = {^c}\mathbf{T}_r {^r}\mathbf{T}_w(\mathbf{r})(X, Y, Z, 1)^T$$

and we have their pixel coordinates in the image plane

$$(u, v, 1)^T = \mathbf{K}\,\mathbf{\Pi}(^cX, {^c}Y {^c}Z, 1)^T.$$

The image projection constraint C_{proj} is given by:

$$C_{proj} = \{u = p_x x_i + u_0,\ v = p_y y_i + v_0\}$$

$$\text{with } x_i = \frac{^cX_i}{^cZ_i}, y_i = \frac{^cY_i}{^cZ_i}.$$

- C_{front} expresses the constraint w.r.t. the fact that we have a front-looking camera. For all points in camera frame, $^cZ_i > 0$, i.e., observed points are in the front half-space:

$$C_{front} = \{^cZ_i > 0\}$$

Therefore $C_{cam,i} = \{C_{proj,i}, C_{front,i}\}$ is the set of all these constraints for all 2D-3D pairs $(\mathbf{x}_i, {^w}\mathbf{X}_i)$, with $i \in 1...N$.

B. Range constraints

Considering the three known ranges, we have 3 constraints derived from this.

- C_{base} expresses the constraint w.r.t. the distance to the base station for each UAV.

Considering the base $\mathbf{b}$ as defined in Section II-B in the world frame, we define d_k the distance constraint from $\mathbf{b}$ to UAV k (with $k \in 1, 2$) by

$$C_{r_k,base} : \{d_k = \|\mathbf{p}_k - \mathbf{b}\|_2\}$$

- C_{r_1,r_2} expresses the constraint w.r.t. the inter-distance between UAVs.

Let d_{12} be the measured distance between both UAVs. The constraint can be expressed as follow

$$C_{r_1,r_2} = \{d_{12} = \|\mathbf{r}_1 - \mathbf{r}_2\|_2\}$$

C. Approach

Taking into account the fact that all UAV exchange their image measurements, their distance to the base station, and the distance between each other, we can formulate the problem as a centralized CSP $\mathcal{H}$ defined by:

$$\mathcal{H} : \begin{pmatrix} \mathbf{r}_1 \in [\mathbf{r}_1], \mathbf{r}_2 \in [\mathbf{r}_2], \\ \bar{\mathbf{x}}_{i,k} \in [\bar{\mathbf{x}}_{i,k}], {^w}\mathbf{X}_i \in [{^w}\mathbf{X}_i], \\ \{C_{cam,i,k},\ i \in 1...N, k \in \{1,2\}, \\ C_{r_1,base}, C_{r_2,base}, C_{r_1,r_2}\} \end{pmatrix} \quad (3)$$

where the boxes $[\mathbf{r}_1]$ and $[\mathbf{r}_2]$ represent prior knowledge of the UAV poses, $[\bar{\mathbf{x}}_{i,k}]$ represents bounded-error measurements of the landmarks in the image seen by UAV number k, and the width of $[{^w}\mathbf{X}_i]$ represents the uncertainty about world coordinates of the landmarks.

To characterize the pose solution-set of each UAV, we use the SIVIA algorithm presented in section III-C to compute the subpaving of all feasible poses in $[\mathbf{r}_1], [\mathbf{r}_2]$ that are consistent with the measurements using $\mathcal{H}$ (CSP in Eq. 3).

As the components $\phi_1, \theta_1, \phi_2, \theta_2, z_1$ and z_2 of the poses $\mathbf{r}_1$ and $\mathbf{r}_2$ are directly measured, the initial domains of theses components of the state are known with a small uncertainty related to the sensors precision. Let $\epsilon_\phi, \epsilon_\theta$ and ϵ_z, these measurements uncertainties. The domains are therefore set to intervals $[\phi_1^{\text{meas.}} \pm \epsilon_\phi], [\theta_1^{\text{meas.}} \pm \epsilon_\theta], [\phi_2^{\text{meas.}} \pm \epsilon_\phi], [\theta_2^{\text{meas.}} \pm \epsilon_\theta], [z_1^{\text{meas.}} \pm \epsilon_z]$ and $[z_2^{\text{meas.}} \pm \epsilon_z]$. For the unknown parameters $(x_1, y_2, \psi_2, x_2, y_2$ and $\psi_2)$, their initial domains in $[\mathbf{r}_1], [\mathbf{r}_2]$ are set w.r.t. the prior knowledge of the environment (in this article, the experimental room dimensions)

After setting the initial domain for each robot, a forward-backward contractor (HC4) is built and applied in the SIVIA algorithm for all unknown parameters (x, y, ψ). This enables to recursively reduce the initial domain by rejecting the parts that are not consistent with the measurements when applying the constraints of $\mathcal{H}$).

At convergence of SIVIA, bounded domains for the poses $\mathbf{r}_1$ and $\mathbf{r}_2$ are obtained in the form of subpavings $\mathbb{X}_{\mathbf{r}_1}^+$ and
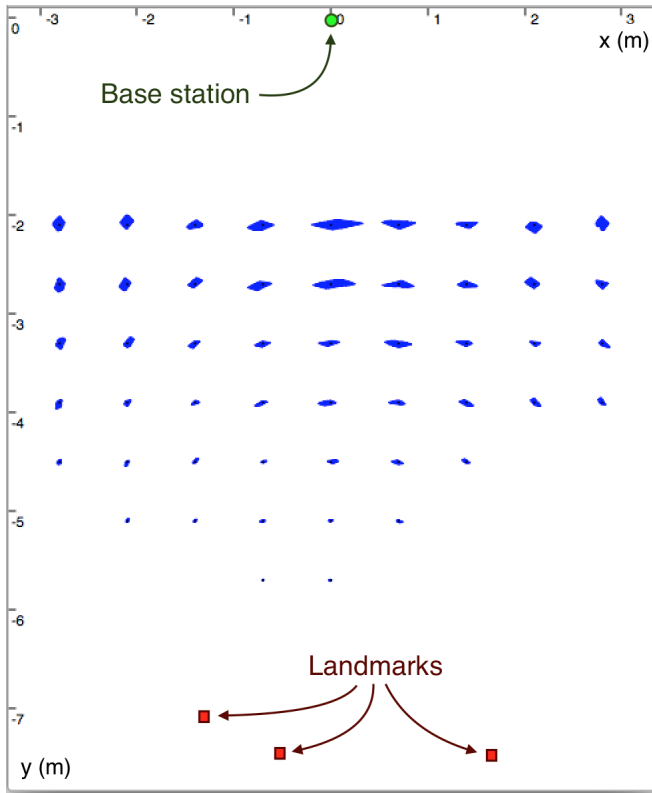
Fig. 3: Position domains for a single robot, at 50 different positions, with range measurement to the base station (± 5 cm) and 3 landmarks in view (± 0.5 px). Top view.

$\mathbb{X}_{\mathbf{r}_2}^+$. Assuming the measurements error bounds have not been violated (i.e., the bounded error model is respected), the computed domains are guaranteed to contain the true robot poses $\mathbf{r}_1^*$ and $\mathbf{r}_2^*$, i.e., $\mathbf{r}_1^* \in \mathbb{X}_{\mathbf{r}_1}^+$ and $\mathbf{r}_2^* \in \mathbb{X}_{\mathbf{r}_2}^+$.

V. COOPERATIVE LOCALIZATION

In this section, we present results obtained from simulations, and an experimental trial using cameras onboard UAVs.

A. Simulation results

The simulation environment consists in a room, with the base station at coordinates $\mathbf{b} = (0, 0, 0)$, and three landmarks of known positions on the other side of the room (see Fig. 3 for a top view). A pattern of 50 test positions has been generated to validate our approach. This leads to a set of 2450 two-robot positions combinations (for every fixed position $i \in 1..50$ of robot R_1, all positions $j \neq i$, $j \in 1..50$ of R_2 are considered).

Altitude, pitch angle and roll angle are assumed to be known. Assuming unreliable compass measurements, the robot heading is assumed to be known only very coarsely (within a ± 20 degrees interval). Ranging errors (from base station and inter-UAV) are assumed to be within ± 5 cm. Landmarks tracking in the image is assumed to be ± 0.5 px accurate.
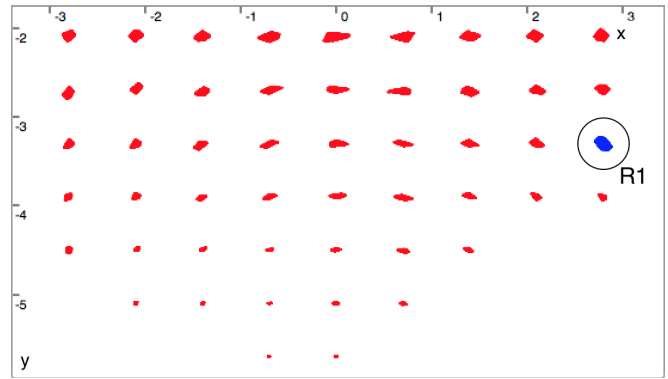

Fig. 4: Position domains for cooperative localization. R_1 is fixed (in blue, circled), R_2 at 49 different positions (in red).

1) Single robot: The results of single robot positioning, using ranging from the base station and camera measurements of 3 landmarks are shown in Fig. 3. The figure represents in blue the projection on the (x, y) plane of the subpavings computed for the 50 simulated positions. Statistics of the width of the computed domains are reported in left part of Table I. We clearly observe in the figure that pose domains get larger as the robot gets further from the landmarks. It is also interesting to notice bigger uncertainties near the $x = 0$ axis. This is due to the poorly conditioned intersection of the sphere from the base station distance and the landmarks projection constraints.

2) Cooperative localization with full visibility: Position domains have been computed using the proposed cooperative localization method for the two-robot combinations of the simulated positions patterns, assuming both robots are identical and able to see the three landmarks. Overall improvement on the position uncertainty (w.r.t. the single robot case) is mainly visible on the x coordinate (better min and mean values in Table I). This improvement is not very large because the position uncertainty domains of both robots without measurements exchange are often of comparable sizes, and the geometric configuration does not always allow domain contraction. Moreover, some of the computed domains are larger that in the single robot case, due to the ϵ parameter of the set inversion process which prevents perfect approximation of the domain (this is visible in the "max" column of Table I, and also affects the means).

Figure 4 shows a configuration where cooperative localization sensibly reduces position uncertainty, thanks to a good geometric configuration (compare the $x = 0$ column with Fig. 3).

On a 2014 2.8GHz Core i7, with a single threaded implementation, the average computation time is of 200 ms for $\epsilon = 30$ mm (presented results), and of 32 ms for $\epsilon = 60$ mm. Increasing ϵ reduces the computation time, but provides a rougher estimate of the feasible poses domain. When using a fixed ϵ value, the computation time varies with the size of the computed domain and is linear with the number of landmarks seen by a robot. Real-time implementation is possible using, either a fixed number of SIVIA iterations, or an any-time
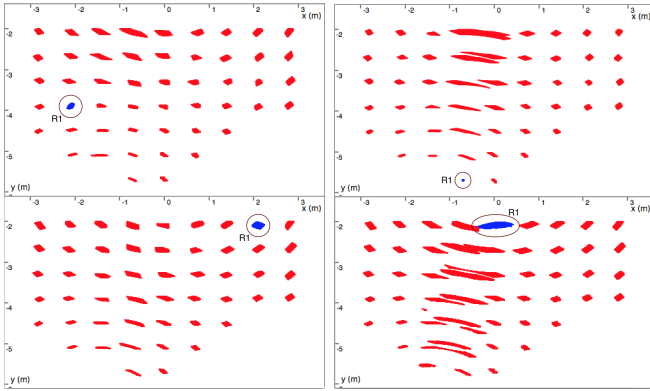
Fig. 5: Cooperative localization results when R_1 sees 3 landmarks and R_2 sees only one. Each figure corresponds to a different fixed position of R_1 (circled blue domain). The (x_2, y_2) domains at 49 positions of R_2 are shown in red.

implementation like in [26].

Domain dimension	Single robot			2-robot coop.		
	min	mean	max	min	mean	max
x (cm)	3.62	15.8	53.8	2.8	15.4	60
y (cm)	2.0	8.0	16.53	1.1	8.1	18.6
yaw (deg)	0.75	2.1	5.6	0.67	2.1	6.2

TABLE I: Pose domains width in the single robot case and in the full visibility two-robot cooperative case.

3) Cooperative localization with reduced visibility: To emphasize the benefits of the proposed cooperative localization, we now suppose that the robot R_1 sees all 3 landmarks, but the robot R_2 only sees one landmark. This situation could arise when an operator wants to rotate an UAV in order to observe another part of the environment. Localization with a single landmark is impossible for a single UAV (horizontal position is only constrained by the distance to the base station, which, given uncertainties and badly known heading, leads to an arc-shaped solution set). Knowing the distance to the other (well localized) UAV enables to compute an usable position domain. Simulations have been done for the 2450 possible configurations. Obtained uncertainty domains are, of course, larger than in the previous case, due to the lack of camera measurements (see Table II). The very large maximum values obtained in Table II correspond to a few cases where the heading measurement uncertainty prevents from resolving the position ambiguity of R_2, leaving a set of two disjoint pose domains. Figure 5 shows the pose domain subpavings obtained for the robot R_2 (in red), for 4 different positions of the robot R_1 (whose position domain is painted in blue and circled). It clearly shows the influence of the uncertainty

	min	mean	max
x (cm)	11.09	32.5	357.3
y (cm)	2.9	14.45	83.39
yaw (deg)	2.09	5.62	48

TABLE II: Pose domain width of R_2 in the reduced visibility cooperative case. R_1 sees 3 landmarks and R_2 only sees one.

Method	Horiz. pos. error (cm)			Heading error (deg)		
	mean	95 %	max	mean	95 %	max
Interval (CoB)	1.51	3.52	17.5	0.16	0.41	1.99
Interval + LM	1.43	3.36	10.1	0.14	0.36	1.06
LM (GT guess)	1.43	3.36	10.1	0.14	0.36	1.06
LM (bad guess)	351.7	645.2	721.9	39.1	105.7	179.3

TABLE III: Position (2-D horizontal) and heading error statistics in the 2-UAV cooperative case with full visibility

Method	Horiz. pos. error (cm)			Heading error (deg)		
	mean	95 %	max	mean	95 %	max
Interval (CoB)	5.39	13.61	167.7	0.80	2.59	19.4
Interval + LM	4.11	7.00	328.4	0.56	1.19	39.4
LM (GT guess)	2.90	6.68	34.1	0.36	1.08	5.79
LM (bad guess)	391.9	760.8	1074	140.5	179.0	180.0

TABLE IV: Position and heading error of R_2 in the cooperative case with reduced visibility. R_1 sees all landmarks and R_2 only sees one.

of R_1, but also of the geometric configuration: large domains obtained in the center of the two right figures are due to the fact that distance to the base and distance to the robot R_1 constrain the position of R_2 at the intersection of almost tangent circles. Bottom right figure of Fig. 5 even shows a few ambiguous cases, where the solution set is made of two disjoint subsets.

B. Point estimate and comparison with weighted least squares

By computing the set of all feasible poses, our approach is interesting for solving the initial localization problem. When point estimate is needed, a straightforward solution consists in taking the center of the bounding-box (CoB) of the solution subpaving.

The horizontal position and heading errors of the CoB estimate are reported in the first row of Table III (full visibilty case) and of Table IV (robot R_2 in the reduced visibility case). Cumulative distributions of the errors for the reduced visibility case are shown in Fig. 6 (dotted blue lines).

These estimates are compared to non-linear weighted least squares (WLS) estimates computed with the Levenberg-Marquardt (LM) algorithm, using all observations (images, distances, altitude and IMU angles). The measurements weights are initially set w.r.t. the standard deviations corresponding to the widths of the interval error bounds and then fine-tuned. The ground truth is first used as an initial guess (*LM GT guess* in the tables, dashed red lines in Fig. 6). It shows that the center of the bounding box is a good point estimate, with errors figures very close to the WLS estimate when the domains are not too large. In the full visibility case, the CoB average errors are 1.51 cm horizontal and 0.16 degrees in heading, vs 1.51 cm and 0.14 degrees with the WLS. However, when the computed domains gets bigger, the CoB becomes less precise. This can be seen with robot R_2 in the reduced visibility case (Table IV).

An idea is to combine the domain computed with the interval method (as an uncertainty indicator) with LM to provide a precise point estimate. When initialized far from the solution (here, the center of the simulation area), LM
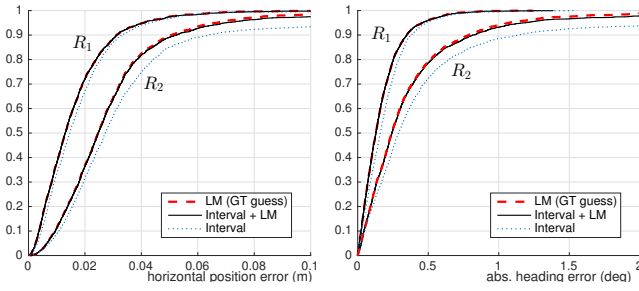
Fig. 6: Cumulative distribution of horizontal position error (left) and absolute heading error (right), for robot R_1 seeing 3 landmarks and robot R_2 seeing only one.
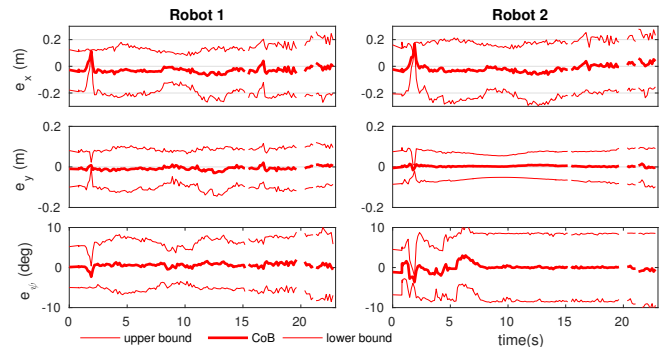

Fig. 7: Onboard camera view of the two UAVs

Fig. 8: Errors bounds and punctual estimate in the experimental cooperative case. Zero corresponds to ground truth.

Fig. 9: Position domains of the two UAVs at the beginning of the experiment: R_1 in blue (right), R_2 in red (left).

often fails to reach the true pose; this can be seen in the fourth row of Tables III and IV (*LM bad guess*).

We thus use the CoB as initial guess for LM, in order to get a better point estimate (*Interval+LM*). In the full visibility case, starting at the CoB make the LM result similar to the one initialized by the ground truth (second row of Table IV). This shows the CoB is a good initial point for efficient convergence to the solution and helps to avoid local minimums.

In the reduced visibility test case, the *Interval+LM* method significantly improves point estimates, especially in the case of robot R_2 (see how plain black curves in Fig. 6 are almost superimposed with ground-truth-initialized-LM curves).

C. Experimental validation

The experimental environment is made of a Vicon motion capture system that tracks the pose of quadcopter UAVs (*MikroKopter MK-Quadro* equipped with a camera) to provide ground truth and the landmarks positions. The landmarks are cubes for which the positions are known in the reference frame and have a very visible face for the purpose of image tracking (Fig. 7).

In addition to the uncertainties described in the previous simulation section, this experiment also considers small uncertainties on the roll and pitch angles (± 0.6 deg), on the altitude measurements (± 1 cm) and on the landmark positions (± 1 mm).

The experiment mainly consists in moving the UAVs along the x axis. The bounds of the computed pose domains for the two robots are reported in Fig. 8. Similarly to the simulation, uncertainty is larger on the x component of the position (36 cm mean width) than on y (17 cm mean width), which is due to the geometry of the problem. This can be observed on Fig. 9, where the position domains computed for the two
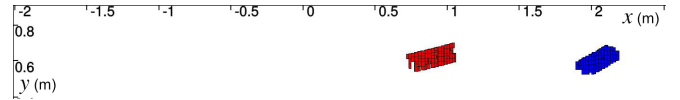
robots R_1 (in blue, right) and R_2 (in red, left) at the first epoch are represented.

The first row of Table V reports the position and heading errors obtained during the experiment, when using the center of the computed domain as a point estimate. The mean horizontal error is of 3.03 cm, and the mean heading error is less than 0.6 degree. An Extended Kalman Filter (EKF) has been applied on the same data, with the observations gains set identically to the WLS of the previous subsection. The results of the EKF (second row of Table V) shows the proposed interval method outperforms the EKF in terms of position precision. The EKF initial state has been set to the true pose, to avoid initial localization problem and keep the comparison fair. The poor results of the EKF on position estimate are mostly due to the assumption of centered, white and independent noise on measurements, which is not the case in practice due to correlation in landmark tracking errors (e.g., due to motion blur) and biases. The EKF, however, yields a better heading estimate than the interval method. An explanation comes from the ϵ setting of the interval method, which limits the precision of the pose domain approximation.

Camera tracking of the landmarks has failed at several epochs of the trial, leading to measurements error bounds violation. At times $t \approx 16$ s and around $t = 20$ s, the result of the interval method is the empty set (blanks in Fig. 8). It is the consequence of measurements inconsistency, i.e., there is no pose that satisfies all the constraints of Eq. 3.

Method	Horiz. pos. error (cm)			Heading error (deg)		
	mean	95 %	max	mean	95 %	max
Interval	3.03	6.01	7.22	0.58	1.63	3.02
EKF	4.2	9.47	20.6	0.26	0.66	5.25

TABLE V: Position and heading error in the cooperative experiment. Statistics of EKF and interval errors

This inconsistency enables to prove that the bounded error assumption is violated for these epochs. Around time $t \approx 2$ s, we observe that the computed pose domains are very small and do not contain the true robot poses. This still correspond to error bounds violation, but not enough to yield empty set. It can be noted that, during these tracking failure epochs, the EKF precision is also negatively impacted, with average errors of 6.4 cm in position and 1.0 deg in heading (vs 4.1 cm and 0.2 deg average over the other epochs).

VI. CONCLUSION

This paper has presented a set-membership cooperative localization approach for two UAVs. Assuming that measurement errors are bounded and can be represented by intervals, a set-inversion method based on constraint propagation techniques is employed to compute a guaranteed outer approximation of the set of feasible poses for each robot. Both robots measure distances to a base station, the distance between them, and can see known landmarks with their cameras. They share their measurements to enable cooperative localization. We show that cooperation enables to tighten the pose uncertainty domain, especially when the geometrical configuration is favorable to inter-distance measurement. The center of the computed pose domain is a good point estimate that can be refined using the Levenberg-Marquardt algorithm. Another strong aspect of cooperative localization is that it gives positioning ability to a robot that otherwise would have not been able to compute a position fix due to the lack of measurements. An example is given, where one of the robot can only see a single landmark, while the other can see all of them. These aspects have been demonstrated by simulating multiple two-robot configurations. Validation with experimental data shows the usability of the method with real image measurements, and better position accuracy than an Extended Kalman Filter.

ACKNOWLEDGMENT

REFERENCES

[1] I. Maza and A. Ollero, "Multiple UAV cooperative searching operation using polygon area decomposition and efficient coverage algorithms," in *Distributed Autonomous Robotic Systems 6*. Tokyo: Springer Japan, 2007, pp. 221–230.

[2] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, Jan 2011.

[3] F. Schiano and P. R. Giordano, "Bearing rigidity maintenance for formations of quadrotor UAVs," in *ICRA 2017 - IEEE Int. Conf. on Robotics and Automation*, Singapore, Singapore, May 2017, pp. 1467 – 1474.

[4] D. Fox, J. Ko, K. Konolige, B. Limketkai, D. Schulz, and B. Stewart, "Distributed multirobot exploration and mapping," *Proceedings of the IEEE*, vol. 94, no. 7, pp. 1325–1339, July 2006.

[5] S. Roumeliotis, G. A. Bekey, *et al.*, "Distributed multirobot localization," *Robotics and Automation, IEEE Trans. on*, vol. 18, no. 5, pp. 781–795, 2002.

[6] G. P. Huang, N. Trawny, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based consistent EKF estimators for multi-robot cooperative localization," *Autonomous Robots*, vol. 30, no. 1, pp. 99–122, 2011.

[7] S. J. Julier and J. K. Uhlmann, "A non-divergent estimation algorithm in the presence of unknown correlations," in *In Proc. of the American Control Conference*, 1997, pp. 2369–2373.

[8] A. Bahr, J. J. Leonard, and M. F. Fallon, "Cooperative localization for autonomous underwater vehicles," *The Int. Journal of Robotics Research*, vol. 28, no. 6, pp. 714–728, 2009.

[9] J. Spletzer and C. J. Taylor, "A bounded uncertainty approach to multi-robot localization," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2003, pp. 1258–1264.

[10] J. S. Camillo Jose Taylor, "A bounded uncertainty approach to cooperative localization using relative bearing constraints," in *IEEE/RSJ Int. Conf.on Intelligent Robots and Systems (IROS)*, October 2007, pp. 2500–2506.

[11] R. Moore, *Interval analysis*. Prentice Hall, 1966.

[12] E. Colle and S. Galerne, "Mobile robot localization by multiangulation using set inversion," *Robotics and Autonomous Systems*, vol. 61, no. 1, pp. 39–48, Jan. 2013.

[13] V. Drevelle and P. Bonnifait, "Localization confidence domains via set inversion on short-term trajectory," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1244–1256, October 2013.

[14] A. Bethencourt and L. Jaulin, "Cooperative localization of underwater robots with unsynchronized clocks," *Paladyn, Journal of Behavioral Robotics*, vol. 4, no. 4, pp. 233–244, 2013.

[15] K. Lassoued, P. Bonnifait, and I. Fantoni, "Cooperative localization with reliable confidence domains between vehicles sharing GNSS pseudoranges errors with no base station," *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 1, pp. 22–34, Spring 2017.

[16] I.-F. Kenmogne, V. Drevelle, and E. Marchand, "Image-based UAV localization using interval methods," in *IROS 2017 - IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Vancouver, Canada, Sept. 2017, pp. 5285–5291.

[17] Z. Sahinoglu and S. Gezici, "Ranging in the IEEE 802.15.4a standard," in *2006 IEEE Annual Wireless and Microwave Technology Conference*, Dec 2006, pp. 1–5.

[18] F. Schiano and P. R. Giordano, "Bearing rigidity maintenance for formations of quadrotor uavs," in *ICRA 2017-IEEE International Conference on Robotics and Automation*, 2017, pp. 1467–1474.

[19] R. Tsai, "A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf tv cameras and lenses," *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 323–344, August 1987.

[20] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2001.

[21] V. Lepetit, F. Moreno-Noguer, and P. Fua, "EPnP: An accurate O(n) solution to the PnP problem," *Int. Journal of Computer Vision*, vol. 81, no. 2, pp. 155–166, 2009.

[22] E. Marchand, H. Uchiyama, and F. Spindler, "Pose estimation for augmented reality: a hands-on survey," *IEEE Trans. on Visualization and Computer Graphics*, vol. 22, no. 12, pp. 2633–2651, Dec. 2016.

[23] G. Chabert and L. Jaulin, "Contractor programming," *Artificial Intelligence*, vol. 173, no. 11, pp. 1079 – 1100, 2009.

[24] L. Jaulin and E. Walter, "Set inversion via interval analysis for nonlinear bounded-error estimation," *Automatica*, vol. 29, no. 4, pp. 1053 – 1064, 1993.

[25] F. Benhamou, F. Goualard, L. Granvilliers, and J.-F. Puget, "Revising hull and box consistency," in *Int. Conf. on Logic Programming*. MIT press, 1999, pp. 230–244.

[26] V. Drevelle and P. Bonnifait, "Interval-based fast fault detection and identification applied to radio-navigation multipath," *Int. Journal of Adaptive Control and Signal Processing*, vol. 30, no. 2, pp. 154–172, 2016.