Hierarchical Control of a Team of Quadrotors for Cooperative Active Target Tracking

Utku Gürcüoğlu, Gustavo A. Puerto-Souza, Fabio Morbidi, Gian Luca Mariottini

***Abstract*— This paper proposes a novel active target tracking strategy for a team of cooperating quadrotors equipped with 3-D range-finding sensors. The work builds upon previous research of the authors, and adopts a realistic nonlinear dynamic model for the quadrotors. A hierarchical controller is designed for the generation and tracking of the desired optimal trajectories of the aerial vehicles, and a discrete-time Kalman filter is used for fusing their local estimates of the target position. Under suitable conditions, it is shown that the cost function for the D-optimality criterion that the quadrotors aim at collaboratively reduce, possesses a single global minimum and no local minima. Numerical simulations and real-world experiments show the effectiveness of the proposed control strategy.**

I. Introduction

A. Motivation and related work

In the last few years, the number of applications involving *Unmanned Aerial Vehicles* (UAVs) has steadily grown [1], [2]: these applications include mapping of unknown terrains, security enforcement, infrastructure inspection, espionage and entertainment, just to mention the most representative. Several factors have contributed to the present wide interest in UAVs: decreasing costs and increasing miniaturization (cf., AR.Drone Parrot quadricopter and KMeL Robotics' latest products), enhanced sensing and autonomy, as well as the ability to efficiently operate in 3-D environments.

Three categories of (micro) UAVs are currently under study or development: fixed-wing aircraft, avian-style flapping wing aircraft and rotor craft [3]. Two configurations of rotor craft have recently gained large acceptance: coaxial rotor craft and multi-rotor aircraft, such as the popular *quadrotors* which are considered in this paper.

Multi-agent systems research has been lately extremely successful within the robotics and control communities [4]. Extensive work, in particular, has been done on the subject of tracking targets with multiple static sensors [5] or on the optimal placement of fixed sensors [6]. *Mobile sensor networks* are known to offer distinctive advantages over static sensor arrays in terms of quality of sensing and estimation, area coverage, adaptability to changing conditions, and robustness against failures. *Cooperative active sensing* leverages the mobility of a robotic sensor network in order to enhance the target tracking performances [7]–[10]: here, n robots have to fuse their local measurements and move in order to get the best position estimate of a moving target. In practice, this amounts to minimize a certain scalar-valued function of the covariance matrix of the position estimates, which in *optimum experimental-design theory* [6] is typically the determinant (D-optimality criterion).

In [7], a motion-planning algorithm has been presented for tracking a 2-D moving target by using range-bearing measurements. The control law is based on the gradient of the determinant of the covariance matrix of the target's position estimate with respect to each of the robot's coordinates. In [8], by using a distributed Kalman filter based on dynamic average consensus estimators for information fusion, the controller in [7] has been made implementable in a decentralized fashion. In [9], the authors have presented a target tracking algorithm for a team of unicycle robots with bounds on the positive forward velocity, and shown that the associated optimization problem is in general NP-hard. Nontrivial relaxations to this problem are then proposed for determining the set of feasible locations that each robot should move to in order to collect the most informative distance measurements. More recently, in [11], the results in [9] have been adapted to the case of range-bearing measurements. In addition, constraints on the minimum distance between the robots and the target have been incorporated into the optimization problem, in order to avoid collisions.

B. Original contributions and organization

This paper takes its inspiration from [7], [8] and makes several novel contributions. As it is evident from the previous literature review, a large body of research exists on cooperative active target tracking for *wheeled robots*, but few works in the literature (cf., [12], [13]) have dealt with UAVs. In addition, oversimplified models (e.g., first- or second-order integrators) are typically used for the aerial vehicles. A new *hierarchical controller* is designed in this paper for a team of quadrotor UAVs described by *realistic* discrete-time nonlinear dynamic models, and equipped with 3-D range-finding sensors. The hierarchical controller consists of a trajectory planner, which generates optimal paths according to a D-optimality criterion, and a tracker which computes the actual control input for each quadrotor (i.e. the angular velocity for the four propellers). Differently from [13], a *discrete-time* Kalman filter is used here for fusing the local estimates provided by each aerial vehicle. Under suitable conditions, it is shown that the cost function for the D-optimality criterion that the quadrotors aim at collaboratively reduce, has a single global minimum and no local minima. The effectiveness of our control strategy is demonstrated via numerical simulations and preliminary experimental tests.

The rest of the paper is organized as follows. Sect. II deals with the modeling of the mobile target and 3-D range-finding sensors, and the Kalman filter fusion. Sect. III presents the nonlinear dynamic model for the quadrotors and describes our hierarchical control strategy. In Sect. IV, simulation and experimental results are presented and discussed: finally, Sect. V concludes the paper.

U. Gürcüoğlu, G.A Puerto-Souza and G.L. Mariottini are with the Department of Computer Science and Engineering, University of Texas at Arlington, 500 UTA Boulevard, Arlington, TX 76019, USA, {utku.gurcuoglu, gustavo.puerto}@mavs.uta.edu, gianluca@uta.edu

F. Morbidi is with the NeCS team, INRIA Grenoble Rhône-Alpes, France, fabio.morbidi@inria.fr

II. Measurement Model and Information Fusion

Fig. 1 presents a graphical representation of our setup and introduces some useful notation. Consider a team of n quadrotors, let us suppose that quadrotor i has a known absolute position ${}^{\mathcal{W}}\mathbf{p}_i(k) \in \mathbb{R}^3$, $i \in \{1, \ldots, n\}$ in the world reference frame $\{\mathcal{W}\}$ at the discrete time step k (through this paper left superscripts will be used to denote the reference frame in which a vector is expressed). The attitude of the i-th quadrotor, ${}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k) \in \mathbb{R}^{3\times3}$, corresponds to the orientation of its body frame $\{\mathcal{Q}_i\}$ w.r.t. $\{\mathcal{W}\}$, and it is defined in consecutive roll, pitch and yaw angles denoted by $\varphi_i(k)$, $\vartheta_i(k)$ and $\psi_i(k)$ respectively, according to the *Z–X–Y* Euler angles convention [14, Sect. 4.4.3]. Quadrotor i is equipped with a *3-D range-finding sensor*, which measures the position of a target, ${}^{\mathcal{Q}}\mathbf{x}(k) \in \mathbb{R}^3$, moving in a 3-D obstacle-free environment according to,

$$ {}^{\mathcal{Q}}\mathbf{z}_i(k) = {}^{\mathcal{Q}}\mathbf{x}(k) + {}^{\mathcal{Q}}\boldsymbol{\nu}_i(k), $$

where ${}^{\mathcal{Q}}\boldsymbol{\nu}_i(k) \in \mathbb{R}^3$ is a zero-mean white Gaussian measurement noise with covariance matrix ${}^{\mathcal{Q}}\mathbf{P}_i(k) \in \mathbb{R}^{3\times3}$. The position of the target, ${}^{\mathcal{Q}}\mathbf{x}(k)$, can be expressed in the world reference frame as ${}^{\mathcal{W}}\mathbf{x}(k) = {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k){}^{\mathcal{Q}}\mathbf{x}(k) + {}^{\mathcal{W}}\mathbf{p}_i(k)$. Analogously, the position of the target measured by quadrotor i can be expressed in the world reference frame as:

$$ {}^{\mathcal{W}}\mathbf{z}_i(k) = {}^{\mathcal{W}}\mathbf{x}(k) + {}^{\mathcal{W}}\boldsymbol{\nu}_i(k)\,, $$

where ${}^{\mathcal{W}}\boldsymbol{\nu}_i(k) = {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k){}^{\mathcal{Q}}\boldsymbol{\nu}_i(k)$ is the measurement noise expressed in $\{\mathcal{W}\}$, and its covariance matrix is,

$$ {}^{\mathcal{W}}\mathbf{P}_i(k) = {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k)\,{}^{\mathcal{Q}}\mathbf{P}_i(k)\,{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^{\top}(k). $$

The 3-D range sensor on each quadrotor has a so-called "*sweet spot*", i.e., a spatial location where the measurement uncertainty is minimized. At time k, the *Active Target Tracking* module reported in the block diagram in Fig. 2, outputs

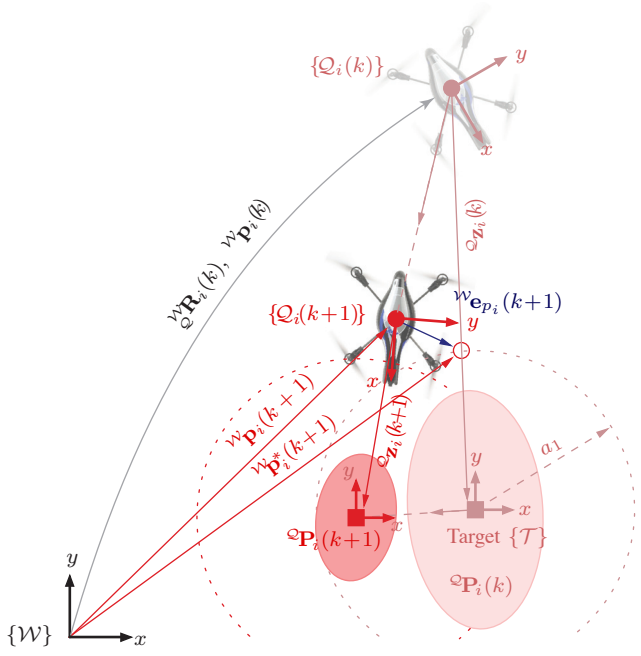


Fig. 1. Graphical description of our active target tracking strategy. From time k to time $k+1$, the quadrotor reduces the uncertainty on the position measurement of the target (red shaded ellipse) by moving towards the sweet spot of the range sensor (dashed circle).

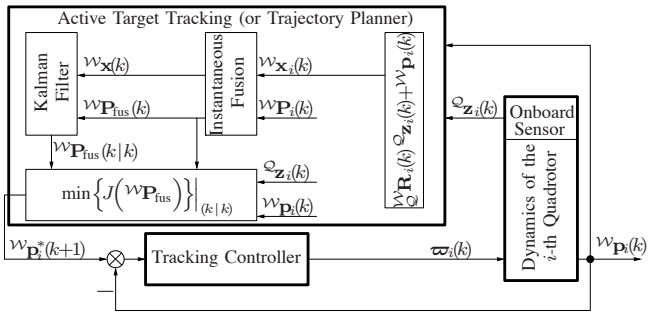


Fig. 2. Block diagram of our cooperative active target tracking strategy. The *Active Target Tracking* module uses the measurements from each quadrotor (and the corresponding covariances) to generate a reference point in the 3-D space, which is used to determine a trajectory-tracking error.

the desired reference position ${}^{\mathcal{W}}\mathbf{p}_i^*(k+1) \in \mathbb{R}^3$ for which a cost function, $J({}^{\mathcal{W}}\hat{\mathbf{P}}\,(k|k))$, is minimized, where ${}^{\mathcal{W}}\hat{\mathbf{P}}\,(k|k)$ denotes the estimate of the overall measurement covariance, which is obtained through the fusion of each quadrotor's measurement by means of a *Kalman filter*. The tracking controller steers each quadrotor towards the desired position by minimizing a *tracking error*, denoted by ${}^{\mathcal{W}}\mathbf{e}_{\mathbf{p}_i} \in \mathbb{R}^3$ and defined as:

$$ {}^{\mathcal{W}}\mathbf{e}_{\mathbf{p}_i}(k+1) \triangleq {}^{\mathcal{W}}\mathbf{p}_i^*(k+1) - {}^{\mathcal{W}}\mathbf{p}_i(k+1). \qquad (1) $$

As a consequence, the uncertainty in the measurement at the next time step, ${}^{\mathcal{Q}}\mathbf{z}_i(k+1)$, will be reduced as qualitatively illustrated in Fig. 1, by the shrinking of the covariance ellipse from ${}^{\mathcal{Q}}\mathbf{P}_i(k)$ to ${}^{\mathcal{Q}}\mathbf{P}_i(k+1)$. The measurement model of each quadrotor is illustrated in Fig. 3 (cf., [13]). A target frame, denoted by $\{\mathcal{T}\}$, is attached to the target and aligned with the world reference frame, $\{\mathcal{W}\}$, at all times.

The covariance matrix of measurement noise of a standard 3-D range-finding sensor in spherical coordinates, can be written as:

$$ {}^{\mathcal{T}}\mathbf{P}_i(k) = \text{diag}(\sigma^2_{\gamma_i}(k),\ \sigma^2_{\beta_i}(k),\ \sigma^2_{r_i}(k)). $$

The variance of the range-measurement noise, $\sigma^2_{r_i}$, is typically represented by a function $f_r(r_i)$ of the distance, r_i, from the i-th quadrotor to the target: the polar and bearing measurement noise variances $\sigma^2_{\gamma_i}$ and $\sigma^2_{\beta_i}$ can also be modeled as dependent on the range r_i through the functions $f_\gamma(r_i)$ and $f_\beta(r_i)$, respectively. A simple but representative form for these functions is,

$$ f_r(r_i) \triangleq a_2(r_i - a_1)^2 + a_0, $$
$$ f_\beta(r_i) \triangleq a_\beta\, f_r(r_i), \quad f_\gamma(r_i) \triangleq a_\gamma\, f_r(r_i), $$

where a_0, a_1, a_2, a_β, a_γ are strictly positive parameters. This model, as aforementioned, assumes the existence of a *sweet spot* located at a distance a_1 from the target, where uncertainty in measurements is minimal.

The covariance matrix ${}^{\mathcal{T}}\mathbf{P}_i(k)$ is rotated to obtain the covariance matrix of measurement noise in Cartesian coordinates, ${}^{\mathcal{W}}\mathbf{P}_i(k)$, as ${}^{\mathcal{W}}\mathbf{P}_i(k) \triangleq \mathbf{T}_i(k)\,{}^{\mathcal{T}}\mathbf{P}_i(k)\,\mathbf{T}_i^{\top}(k)$, where

$$ \mathbf{T}_i = \boldsymbol{\mathcal{R}}_z(\beta_i)\boldsymbol{\mathcal{R}}_y(\gamma_i) = \begin{bmatrix} \cos\beta_i\cos\gamma_i & -\sin\beta_i & \cos\beta_i\sin\gamma_i \\ \sin\beta_i\cos\gamma_i & \cos\beta_i & \sin\beta_i\sin\gamma_i \\ -\sin\gamma_i & 0 & \cos\gamma_i \end{bmatrix}, $$

being $\boldsymbol{\mathcal{R}}_z(\beta_i)$ and $\boldsymbol{\mathcal{R}}_y(\gamma_i)$ the basic 3×3 rotation matrices about the z- and y-axes of an angle β_i and γ_i (see Fig. 3).

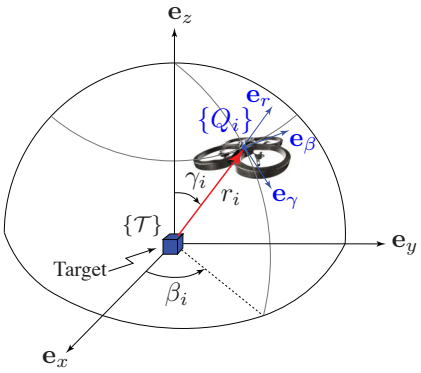


Fig. 3. Measurement model for the 3-D range-finding sensor onboard quadrotor i: The spherical coordinates $(r_i,\, \beta_i,\, \gamma_i)$ are used, where $r_i \in (0,\, +\infty)$ is the distance from quadrotor i to the target, $\beta_i \in (0,\, 2\pi) \setminus \{\pi\}$ is the *bearing angle* and $\gamma_i \in (0,\, \pi]$ is the *polar angle*. $\{\mathbf{e}_x,\, \mathbf{e}_y,\, \mathbf{e}_z\}$, $\{\mathbf{e}_\gamma,\, \mathbf{e}_\beta,\, \mathbf{e}_r\}$ are the Cartesian and spherical orthonormal bases.

A. Information fusion

All the measured target positions, ${}^{\mathcal{W}}\mathbf{z}_i(k)$, and their covariances, ${}^{\mathcal{W}}\mathbf{P}_i(k)$, $i\{1, \ldots, n\}$, are centrally fused into a single weighted least-squares estimate of target position, ${}^{\mathcal{W}}\hat{\mathbf{x}}(k)$, with covariance ${}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k)$:

$$\begin{aligned} {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k) &= \left[\textstyle\sum_{i=1}^{n} {}^{\mathcal{W}}\mathbf{P}_i^{-1}(k)\right]^{-1}, \\ {}^{\mathcal{W}}\hat{\mathbf{x}}(k) &= {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k) \textstyle\sum_{i=1}^{n} \left[{}^{\mathcal{W}}\mathbf{P}_i^{-1}(k)\, {}^{\mathcal{W}}\mathbf{z}_i(k)\right]. \end{aligned}$$

In order to exploit the past measurements as well, a Kalman filter is employed, which provides ${}^{\mathcal{W}}\hat{\mathbf{x}}(k|k)$ and ${}^{\mathcal{W}}\mathbf{P}(k|k)$. To this end, let us consider the following general model for the target motion:

$$ {}^{\mathcal{W}}\mathbf{x}(k) = \mathbf{F}\, {}^{\mathcal{W}}\mathbf{x}(k-1) + \mathbf{B}\, {}^{\mathcal{W}}\mathbf{u}(k-1) + {}^{\mathcal{W}}\boldsymbol{\omega}(k), \quad (2)$$

where $\mathbf{F}, \mathbf{B} \in \mathbb{R}^{3\times 3}$ are given constant matrices, ${}^{\mathcal{W}}\mathbf{x}(k-1) \in \mathbb{R}^3$ denotes the position of the target at time $k-1$, ${}^{\mathcal{W}}\mathbf{u}(k-1) \in \mathbb{R}^3$ is a known exogenous input and ${}^{\mathcal{W}}\boldsymbol{\omega}(k-1) \in \mathbb{R}^3$ is a zero-mean white Gaussian noise with covariance matrix $\mathbf{Q} \in \mathbb{R}^{3\times 3}$. The discrete-time Kalman filter is then designed as:

$$\begin{aligned} {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k-1) &= \mathbf{F}\, {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k-1|k-1)\mathbf{F}^\top + \mathbf{Q}, & \text{(3a)} \\ {}^{\mathcal{W}}\hat{\mathbf{x}}(k|k-1) &= \mathbf{F}\, {}^{\mathcal{W}}\hat{\mathbf{x}}(k-1|k-1) + \mathbf{B}\, {}^{\mathcal{W}}\mathbf{u}(k), & \text{(3b)} \end{aligned}$$

and

$$\begin{aligned} {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k) &= [\mathbf{I}_3 - \mathbf{K}(k)]\, {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k-1), & \text{(3c)} \\ {}^{\mathcal{W}}\hat{\mathbf{x}}(k|k) &= {}^{\mathcal{W}}\hat{\mathbf{x}}(k|k-1) + \mathbf{K}(k) {}^{\mathcal{W}}\boldsymbol{\nu}(k), & \text{(3d)} \end{aligned}$$

where $\mathbf{I}_3$ is the 3×3 identity matrix, and the Kalman gain and innovations are:

$$\begin{aligned} \mathbf{K}(k) &= {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k-1) \left[{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k-1) + {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k)\right]^{-1}, \\ {}^{\mathcal{W}}\boldsymbol{\nu}(k) &= {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k) \left[\mathbf{y}(k) - \mathbf{C}(k) {}^{\mathcal{W}}\hat{\mathbf{x}}(k|k-1)\right], \end{aligned}$$

with $\mathbf{y}(k) \triangleq \sum_{i=1}^{n} {}^{\mathcal{W}}\mathbf{P}_i^{-1}(k)\, {}^{\mathcal{W}}\mathbf{z}_i(k)$ and $\mathbf{C}(k) \triangleq \sum_{i=1}^{n} {}^{\mathcal{W}}\mathbf{P}_i^{-1}(k)$.

III. Hierarchical quadrotor control

A. Active target tracking

Our cooperative active target tracking strategy computes an optimal trajectory for each quadrotor in order to minimize the cost function,

$$J(k) = \ln\det\left({}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k)\right), \quad (4)$$

according to the D-*optimality* criterion [6, p. 16]. Similarly to [13], a *desired control input* ${}^{\mathcal{W}}\mathbf{a}_i^* \in \mathbb{R}^3$ is computed for each quadrotor as:

$${}^{\mathcal{W}}\mathbf{a}_i^*(k) = -\mathbf{D}\, {}^{\mathcal{W}}\mathbf{v}_i^*(k) - \boldsymbol{\Gamma}\mathbf{T}_i \left[\frac{1}{r_i \sin\beta_i}\frac{\partial J}{\partial \gamma_i}, \frac{1}{r_i}\frac{\partial J}{\partial \beta_i}, \frac{\partial J}{\partial r_i}\right]^\top, \quad (5)$$

where $\mathbf{v}_i^*(k) \in \mathbb{R}^3$ is the desired velocity of quadrotor i at time k, $\mathbf{D} \in \mathbb{R}^{3\times 3}$, $\mathbf{D} \succ \mathbf{0}$, is a damping matrix and $\boldsymbol{\Gamma} \in \mathbb{R}^{3\times 3}$, $\boldsymbol{\Gamma} \succ \mathbf{0}$, is a gain matrix. The desired position and velocity trajectories at time $k+1$ are computed by discrete integration of the desired acceleration trajectory at current time k. For an explicit expression of the partial derivatives appearing on the right-hand side of (5) the reader is referred to the Appendix.

Note that in general, the cost function (4) is *not* convex w.r.t. the relative pose of the quadrotors and the target, and they may possess, in general, multiple local minima. However, we show in the following proposition that under suitable conditions, the cost function $J(K)$ has a single global minimum and no local minima.

***Proposition** 1:* Let us suppose that $a_\beta = a_\gamma = 1$ and consider the Kalman filter in (3). Then, the cost function (4) has a single critical point, a global minimum at $\mathbf{r} \triangleq [r_1, \ldots, r_n]^T = [a_1, \ldots, a_1]^T$ (the sweet spot).

Proof: If $a_\beta = a_\gamma = 1$, then,

$$\mathbf{C} \triangleq \sum_{i=1}^{n} {}^{\mathcal{W}}\mathbf{P}_i^{-1} = \left[\sum_{i=1}^{n} \frac{1}{f_r(r_i)}\right] \mathbf{I}_3 \triangleq \eta(\mathbf{r})\, \mathbf{I}_3,$$

i.e., $\mathbf{C} = \mathbf{C}(\mathbf{r})$ depends *only* on the relative distance between the vehicles and the target, and not on the relative orientations β_i, γ_i. It is easy to verify by direct calculation that,

$$\frac{\partial\, \eta(\mathbf{r})}{\partial\, r_i} = \frac{-2\, a_2\, (r_i - a_1) \sum_{j=1, j\neq i}^{n} f_r(r_j)}{f_r^2(r_i) \prod_{j=1, j\neq i}^{n} f_r(r_j)}, \quad (6)$$

$$\nabla^2 \eta(\mathbf{r})\big|_{\mathbf{r}=[a_1,\ldots,a_1]^T} = \begin{cases} -2\, a_2/a_0^2, & \text{if } n = 1, \\ -\frac{2\, a_2\, (n-1)}{a_0^n}\, \mathbf{I}_n, & \text{if } n \geq 2, \end{cases} \quad (7)$$

$$\frac{\partial^2\, \eta(\mathbf{r})}{\partial\, r_1^2} = \frac{2\, a_2\, (3\, a_2\, (r_1 - a_1)^2 - a_0) \sum_{j=2}^{n} f_r(r_j)}{f_r^3(r_1) \prod_{j=2}^{n} f_r(r_j)}. \quad (8)$$

By using the second-order conditions for convexity and Sylvester's criterion, we deduce from (8) that the function $\eta(\mathbf{r})$ is *not* convex on $\mathbf{r}$ (in fact, the sign of $3\, a_2\, (r_1 - a_1)^2 - a_0$ in (8) is not fixed). However, since the gradient of $\eta(\mathbf{r})$ vanishes only at $[r_1, \ldots, r_n]^T = [a_1, \ldots, a_1]^T$ and since the Hessian of $\eta(\mathbf{r})$ (7) is negative definite at this point, we conclude that $\mathbf{r} = [r_1, \ldots, r_n]^T$ is a *global maximum* for $\eta(\mathbf{r})$. Note now that given two points $\mathbf{r}', \mathbf{r}'' \in \mathbb{R}^n$, we have:

$$\begin{aligned} &\eta(\mathbf{r}') \leq \eta(\mathbf{r}'') \Leftrightarrow \mathbf{C}(\mathbf{r}') \preceq \mathbf{C}(\mathbf{r}'') \Rightarrow {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(\mathbf{r}'') \preceq {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(\mathbf{r}') \\ &\Rightarrow \ln\det({}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(\mathbf{r}'')) \leq \ln\det({}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(\mathbf{r}')), \end{aligned} \quad (9)$$

where the second implication follows from the *comparison theorem* [13] applied to the Riccati recursion obtained by combining (3a) with (3c), and the third from Weyl's inequality [15, Th. 8.4.9]. From (9), it then follows that the cost function (4) has a single critical point, a *global minimum* at $\mathbf{r} = [a_1, \ldots, a_1]^T$. ■

B. Modeling of the quadrotors and tracking controller

The dynamic model used for the quadrotor i is [1]:

$$
\begin{aligned}
{}^{\mathcal{W}}\mathbf{a}_i(k) &= {}^{\mathcal{W}}\mathbf{g} - {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k)\,{}^{\mathcal{Q}}\mathbf{F}_i(k)/m_i, && \text{(10a)}\\
{}^{\mathcal{W}}\mathbf{v}_i(k) &= {}^{\mathcal{W}}\mathbf{v}_i(k-1) + {}^{\mathcal{W}}\mathbf{a}_i(k)\Delta T, && \text{(10b)}\\
{}^{\mathcal{Q}}\boldsymbol{\alpha}_i(k) &= {}^{\mathcal{Q}}\mathbb{I}_i^{-1}(-{}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k)\times {}^{\mathcal{Q}}\mathbb{I}_i\,{}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k) + {}^{\mathcal{Q}}\boldsymbol{\tau}_i(k)), && \text{(10c)}\\
{}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k) &= {}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k-1) + {}^{\mathcal{Q}}\boldsymbol{\alpha}_i(k)\Delta T, && \text{(10d)}
\end{aligned}
$$

where ΔT is the length of the discrete time-step, ${}^{\mathcal{W}}\mathbf{v}_i(k) \in \mathbb{R}^3$ and ${}^{\mathcal{W}}\mathbf{a}_i(k) \in \mathbb{R}^3$ are the translational velocity and the acceleration of the quadrotor at time k respectively, ${}^{\mathcal{W}}\mathbf{g} = [0,\, 0,\, -g]^\top$ is the gravity vector, m_i is the mass of the quadrotor, ${}^{\mathcal{Q}}\boldsymbol{\alpha}_i(k) \in \mathbb{R}^3$ and ${}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k) \in \mathbb{R}^3$ are the vectors of angular accelerations and velocities respectively, ${}^{\mathcal{Q}}\mathbb{I}_i \in \mathbb{R}^{3\times 3}$ is the moment of inertia of the quadrotor, and ${}^{\mathcal{Q}}\mathbf{F}_i(k) = [0,\, 0,\, F_i(k)]^\top$ and ${}^{\mathcal{Q}}\boldsymbol{\tau}_i(k) \in \mathbb{R}^3$ represent the exogenous forces and the exogenous moments which are generated by the propellers of the quadrotor and which act on its airframe. In this work, for the quadrotors, we adopt the nonlinear tracking controller introduced in [16], which has been shown to be almost globally convergent (cf., Fig. 2)

The tracking controller consists of an upper-level translational controller and a lower-level attitude controller, which maintains the quadrotor in a stable *hovering mode*. By linearizing (10a) and (10c) around ${}^{\mathcal{W}}\mathbf{p}_i(k) = {}^{\mathcal{W}}\mathbf{p}_i^*(k)$, $\psi_i(k) = \psi_i^*(k)$, $\vartheta_i(k) = \varphi_i = 0$, ${}^{\mathcal{W}}\mathbf{v}_i(k) = \mathbf{0}$, $\dot{\vartheta}_i(k) = \dot{\varphi}_i(k) = \dot{\psi}_i(k) = 0$, the translational controller is obtained [1]:

$$
\begin{aligned}
\begin{bmatrix}\varphi_i^*(k)\\ \vartheta_i^*(k)\end{bmatrix} &= \frac{1}{g}\begin{bmatrix}\sin(\psi_i^*(k)) & -\cos(\psi_i^*(k)) & 0\\ \cos(\psi_i^*(k)) & -\sin(\psi_i^*(k)) & 0\end{bmatrix}{}^{\mathcal{W}}\mathbf{a}_i^{\mathrm{c}}(k),\\
F_i(k) &= m_i\,{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{b}_{i_3}^\top(k)\,{}^{\mathcal{W}}\mathbf{a}_i^{\mathrm{c}}(k),
\end{aligned}
$$

where $\varphi_i^*(k)$ and $\vartheta_i^*(k)$ are the desired roll and pitch angles respectively, which are passed to the lower-level attitude controller as components of the desired attitude matrix, ${}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^*(k+1)$, (see equ. (11) below). Finally, ${}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{b}_{i_3}(k)$ is the third column of ${}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k)$, projecting the control input onto the z-axis of frame $\{\mathcal{Q}_i\}$, and the acceleration command is:

$$
\begin{aligned}
{}^{\mathcal{W}}\mathbf{a}_i^{\mathrm{c}}(k) \triangleq\; & {}^{\mathcal{W}}\mathbf{a}_i^*(k) + \mathbf{K}_d({}^{\mathcal{W}}\mathbf{v}_i^*(k+1) - {}^{\mathcal{W}}\mathbf{v}_i(k))\\
& + \mathbf{K}_p({}^{\mathcal{W}}\mathbf{p}_i^*(k+1) - {}^{\mathcal{W}}\mathbf{p}_i(k)) + {}^{\mathcal{W}}\mathbf{g},
\end{aligned}
$$

where $\mathbf{K}_d,\, \mathbf{K}_p \in \mathbb{R}^{3\times 3}$ are derivative and proportional control gains. The attitude error is defined as [16]:

$$
\mathbf{e}_{\mathbf{R}_{\times i}}(k) = \frac{1}{2}\left[{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^{*\top}(k+1)\,{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i(k) - {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^\top(k)\,{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^*(k+1)\right], \tag{11}
$$

where ${}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^*(k+1)$ is computed using the desired roll, pitch and yaw angles $\varphi_i^*(k)$, $\vartheta_i^*(k)$, and $\psi_i^*(k+1)$, where $\psi_i^*(k+1)$ is the one time-step ahead reference signal to be tracked. The matrix attitude error $\mathbf{e}_{\mathbf{R}_{\times i}}(k)$ is the skew-symmetric representation of the orientation-error vector $\mathbf{e}_{\mathbf{R}_i}(k) = [e_{\varphi_i},\, e_{\vartheta_i},\, e_{\psi_i}]^\top$. Choosing zero desired angular velocities, i.e. $\boldsymbol{\Omega}_i^*(k+1) = \mathbf{0}$, and compensating for the nonlinear inertial terms, the lower-level attitude controller is then given by,

$$
\begin{aligned}
{}^{\mathcal{Q}}\boldsymbol{\tau}_i(k) =\; & -{}^{\mathcal{Q}}\mathbb{I}_i(\mathbf{K}_{\mathbf{R}}\,\mathbf{e}_{\mathbf{R}_i}(k) + \mathbf{K}_{\boldsymbol{\Omega}}\,\mathbf{e}_{\boldsymbol{\Omega}_i}(k)) + {}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k)\times\\
& {}^{\mathcal{Q}}\mathbb{I}_i\,{}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k) - {}^{\mathcal{Q}}\mathbb{I}_i({}^{\mathcal{Q}}\boldsymbol{\Omega}_i(k)\times {}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^\top(k)\,{}^{\mathcal{W}}_{\mathcal{Q}}\mathbf{R}_i^*(k)\,{}^{\mathcal{Q}}\boldsymbol{\alpha}_i^*(k)),
\end{aligned}
$$

where $\mathbf{K}_{\mathbf{R}} \succ \mathbf{0}, \mathbf{K}_{\boldsymbol{\Omega}} \succ \mathbf{0}$ are gain matrices and $\mathbf{e}_{\boldsymbol{\Omega}_i}(k)$ is the error on the angular velocities. The control input, $\boldsymbol{\varpi}_i \in \mathbb{R}^4$ (i.e. the vector of angular velocities of the four rotors), is then computed as a solution to,

$$
\begin{bmatrix}F_i(k)\\ {}^{\mathcal{Q}}\boldsymbol{\tau}_i(k)\end{bmatrix} = \begin{bmatrix}c_T & c_T & c_T & c_T\\ 0 & d\,c_T & 0 & -d\,c_T\\ -d\,c_T & 0 & c_T & 0\\ -c_Q & c_Q & -c_Q & c_Q\end{bmatrix}\boldsymbol{\varpi}_i^2(k),
$$

where c_T and c_Q are the aerodynamic thrust and reaction coefficients related to the quadrotor airframe and rotors, and d is the rotors' arm length (i.e., the distance between the center of mass of the vehicle and one of the rotors [1]).

IV. Experimental validation

A. Simulation results

Our cooperative active target tracking strategy has been tested through numerical simulations. The spatial, temporal and inertial dimensions considered in our simulations are meters, seconds and kilograms, respectively: the discrete time step is $\Delta T = 0.01$ sec. For the sake of simplicity, the parameters governing the target motion (see equ. (2)) are $\mathbf{F} = \mathbf{B} = \mathbf{I}_3$, $\mathbf{Q} = \sigma_Q^2\mathbf{I}_3$, where $\sigma_Q = 3\times 10^{-3}$, and the velocity input is ${}^{\mathcal{W}}\mathbf{u}(k) = [0,\, 1,\, 0]^\top$. The model parameters for the quadrotors were selected according to [17]: $m_i = 4$, ${}^{\mathcal{Q}}\mathbb{I}_i = \mathrm{diag}(0.082,\, 0.082,\, 0.149)$, $i \in \{1,\, 2\}$, $c_T = 1.3\times 10^{-5}$, $c_Q = 1.1\times 10^{-7}$ and $d = 0.315$.

Fig. 4(a) illustrates the reference trajectories and the actual trajectories of two quadrotors cooperatively tracking the target (black). Some of the points on the reference trajectories are marked with crossed circles in the figure. The initial position of the quadrotors and the target are ${}^{\mathcal{W}}\mathbf{p}_1(0) = [2.5,\, 5,\, 0]^\top$, ${}^{\mathcal{W}}\mathbf{p}_2(0) = [-1.5,\, 5,\, 0]^\top$, and ${}^{\mathcal{W}}\mathbf{x}(0) = [1,\, 5,\, 2]^\top$ and they are marked by circles and a square, respectively. The parameters of the measurement model are $a_0 = 2.5\times 10^{-3}$, $a_1 = 0.5$, $a_2 = 1.3\times 10^{-4}$, $a_\beta = \alpha_\gamma = 1$, which simulate the real depth-measurement uncertainty for a commercial (Kinect) range-finding sensor [18]. The gain and damping matrices are $\boldsymbol{\Gamma} = 6\,\mathbf{I}_3$ and $\mathbf{D} = 0.5\,\mathbf{I}_3$, respectively. The control gains of the tracking controller are $\mathbf{K}_p = \mathrm{diag}(250,\, 250,\, 10)$, $\mathbf{K}_d = \mathrm{diag}(100,\, 100,\, 8)$, $\mathbf{K}_{\mathbf{R}} = \mathrm{diag}(200,\, 200,\, 10)$, $\mathbf{K}_{\boldsymbol{\Omega}} = 10\,\mathbf{I}_3$, and the desired yaw angles are $\psi_1^* = \psi_2^* = 0$. Fig. 4(b) displays the time evolution of r_1 and r_2 which reach the sweet spot located at a distance $a_1 = 0.5$ m from the target (black, dashed line) after about 10 sec., and Fig. 4(c) shows the time history of the cost function $J = \ln\det({}^{\mathcal{W}}\mathbf{P}_{\mathrm{fus}})$. Finally, Fig. 4(d) reports

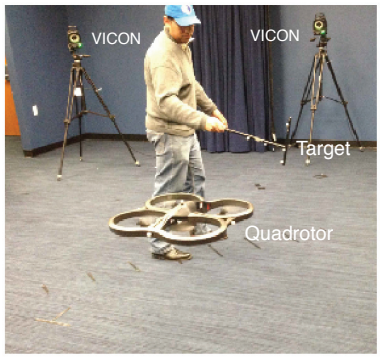


Fig. 5. *Experimental setup*: reflective markers were used to track the target and locate the Parrot quadricopter with the VICON motion capture system.

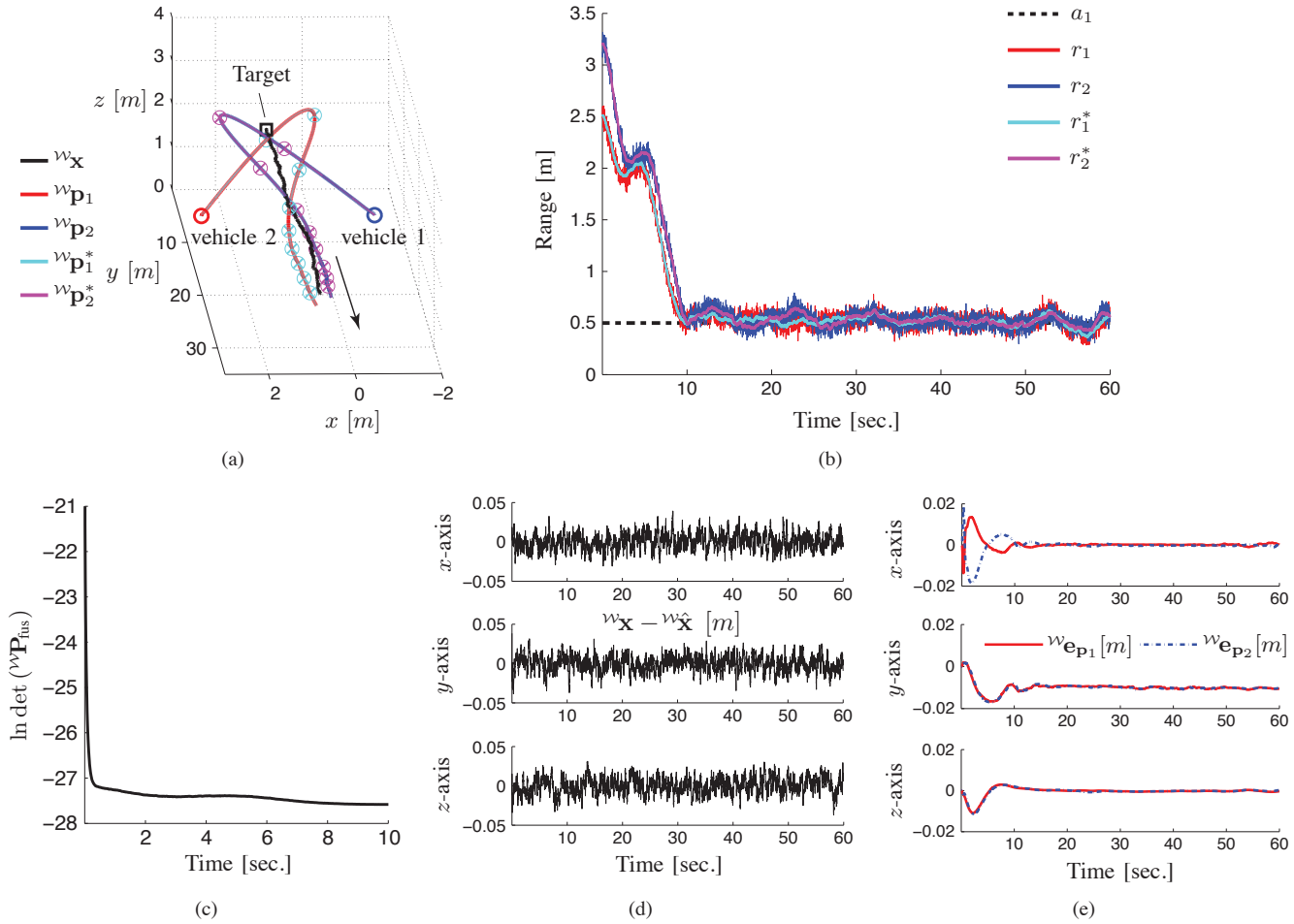


Fig. 4. *Simulation results:* (a) Trajectory of the target and quadrotors: the initial positions are respectively marked with a square and a circle, respectively: some points on the reference trajectories are marked with crossed circles; (b) Convergence of the quadrotors to the sweet spot located at $a_1 = 0.5$ m from the target. (c) Time evolution of $\ln\det(^{\mathcal{W}}\mathbf{P})$ in the first 10 sec. of the simulation; (d), (e) Time history of the estimation and tracking errors, respectively.

the target's position estimation error $^{\mathcal{W}}\mathbf{x} - ^{\mathcal{W}}\hat{\mathbf{x}}$ and Fig. 4(e) presents the control performance by displaying the tracking errors $^{\mathcal{W}}\mathbf{e}_{\mathbf{p}_1}$ and $^{\mathcal{W}}\mathbf{e}_{\mathbf{p}_2}$ (recall equ. (1)).

B. Experimental results

Two preliminary tests of our control strategy have conducted using a Parrot AR-Drone quadrotor (see Fig. 5). In the first experiment the quadrotor minimizes the uncertainty on the measured position of a stationary target by moving towards the sweet spot located at $a_1 = 1$ m from it. In the second experiment, instead, the quadrotor tracks a hand-moved translating target. The parameters that we used in the tests are the same as in Sect. IV-A, except for the gain and damping matrices which are now $\mathbf{\Gamma} = \text{diag}(2,\ 1,\ 1)$, $\mathbf{D} = \mathbf{0}$ and $m_i = 0.42$, $d = 0.285$. For the real-time implementation of our control strategy we used the CVDrone library which allows to interface with the Parrot AR-Drone 2.0 and to control it in velocity, and a VICON motion capture system to precisely locate the target and the quadrotor. Both experiments were carried out on a laptop with an Intel i7-2.20 GHz CPU and 8GB of RAM. Fig. 6(a) shows the trajectory of the quadrotor reaching the sweet spot of the stationary target (black squared point) in the first experiment, and Fig. 6(b) the time evolution of r_1 (solid line). Moreover, Fig. 6(c) shows the time history of the cost function $J = \ln\det(^{\mathcal{W}}\mathbf{P}_{\text{fus}})$. Fig. 7(a) shows the trajectory of the quadrotor (red) tracking the moving target (black) in the second experiment. Note that the target does not start moving until the quadrotor reaches the sweet spot. Fig. 7(b) displays the time evolution of r_1 (solid line), and Fig. 7(c) the time history of the cost function J.

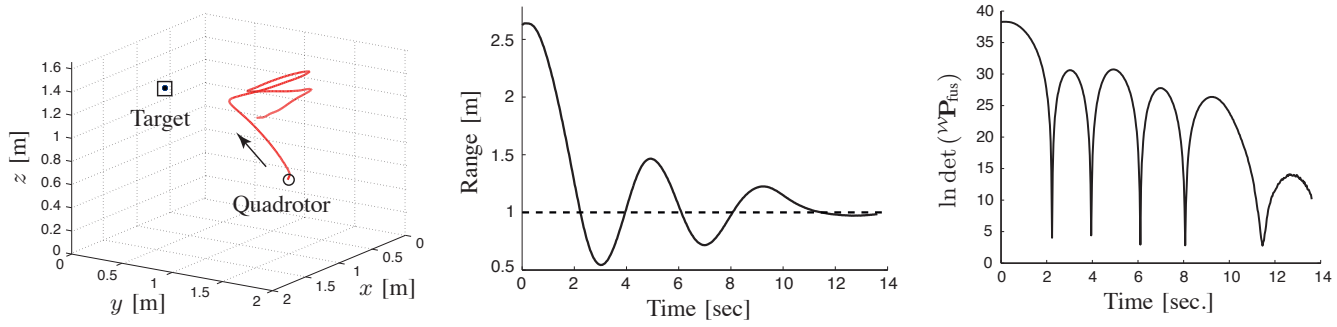
V. Conclusions and Future Work

In this paper, we have presented a new active target tracking strategy for a team of cooperating quadrotors equipped with 3-D range-finding sensors. A hierarchical controller is synthesized for the generation and tracking of desired optimal trajectories for the aerial vehicles, whose motion is described by a realistic nonlinear dynamic model. A discrete-time Kalman filter is used for fusing the local estimates of the quadrotors. Numerical simulations and preliminary experiments have demonstrated the effectiveness of the proposed cooperative active target tracking strategy. Work is in progress to test our controller on multiple quadrotors and to design suitable collision-avoidance modules.

Appendix

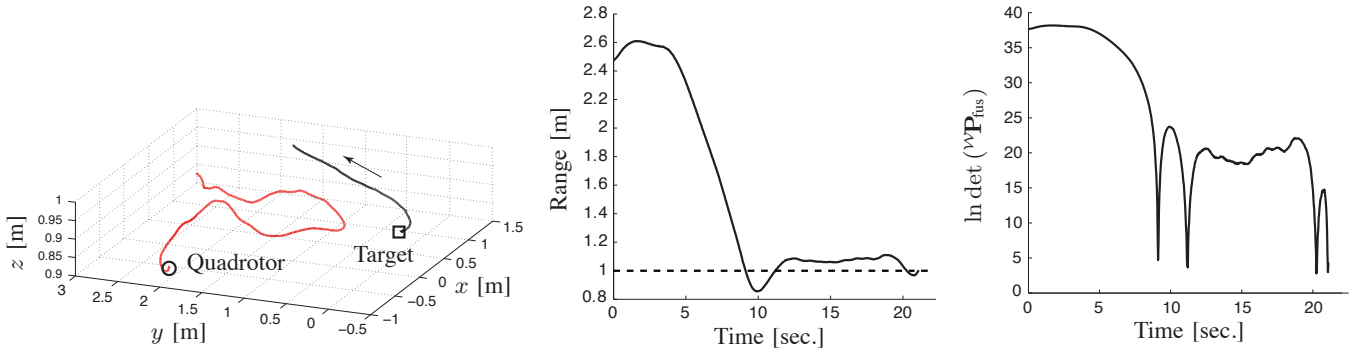
Using the formula in [6, Th. B.17], the partial derivatives on the right-hand side of (5) can be computed as:

$$\frac{\partial J(k)}{\partial r_i} = \text{tr}\big[{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^{-1}(k|k)\,\hat{\mathbf{P}}_k^{r_i}\big],\quad \frac{\partial J(k)}{\partial \beta_i} = \text{tr}\big[{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^{-1}(k|k)\,\hat{\mathbf{P}}_k^{\beta_i}\big]$$
$$\frac{\partial J(k)}{\partial \gamma_i} = \text{tr}\big[{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^{-1}(k|k)\,\hat{\mathbf{P}}_k^{\gamma_i}\big],$$



(a) (b) (c)

Fig. 6. *Experiment with a stationary target*: (a) Trajectory of the quadrotor (red); (b) Time evolution of r_1, reaching the sweet spot located at $a_1 = 1.0$ m from the target (dashed line); (c) Time history of $\ln\det({}^{\mathcal{W}}\mathbf{P}_1)$.



(a) (b) (c)

Fig. 7. *Experiment with a moving target*: (a) Trajectory of the target (black) and quadrotor (red): the initial positions are respectively marked with a square and a circle; (b) Time evolution of r_1, reaching the sweet spot located at $a_1 = 1.0$ m from the target (dashed line); (c) Time history of $\ln\det({}^{\mathcal{W}}\mathbf{P}_1)$.

where

$$\hat{\mathbf{P}}_k^{r_i} \triangleq \frac{\partial {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k)}{\partial r_i},\ \hat{\mathbf{P}}_k^{\beta_i} \triangleq \frac{\partial {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k)}{\partial \beta_i},\ \hat{\mathbf{P}}_k^{\gamma_i} \triangleq \frac{\partial {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k|k)}{\partial \gamma_i}, \quad (12)$$

and $\text{tr}[\cdot]$ denotes the trace of a matrix. By taking the partial derivatives of both sides of the Riccati recursion (obtained by combining (3a) with (3c)) w.r.t. r_i, β_i and γ_i, we find the following *Lyapunov difference equations*, which can be solved by recursion to compute the matrices in (12):

$$\begin{aligned}
\hat{\mathbf{P}}_k^{r_i} &= (\mathbf{F}-\mathbf{K}_k\mathbf{F})\,\hat{\mathbf{P}}_{k-1}^{r_i}(\mathbf{F}-\mathbf{K}_k\mathbf{F})^\top \\
&+ 2a_2(r_i-a_1)\mathbf{K}_k\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k\mathbf{T}_i\,{}^{\mathcal{T}}\mathbf{P}_i^{-2}\text{diag}(a_\gamma,a_\beta,1)\mathbf{T}_i^\top\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k\mathbf{K}_k^\top, \\
\hat{\mathbf{P}}_k^{\beta_i} &= (\mathbf{F}-\mathbf{K}_k\mathbf{F})\,\hat{\mathbf{P}}_{k-1}^{\beta_i}(\mathbf{F}-\mathbf{K}_k\mathbf{F})^\top \\
&- \mathbf{K}_k\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k(\mathbf{A}_{\beta_i}+\mathbf{A}_{\beta_i}^\top)\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k\,\mathbf{K}_k^\top, \\
\hat{\mathbf{P}}_k^{\gamma_i} &= (\mathbf{F}-\mathbf{K}_k\mathbf{F})\,\hat{\mathbf{P}}_{k-1}^{\gamma_i}(\mathbf{F}-\mathbf{K}_k\mathbf{F})^\top \\
&- \mathbf{K}_k\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k(\mathbf{A}_{\gamma_i}+\mathbf{A}_{\gamma_i}^\top)\,{}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k\,\mathbf{K}_k^\top,
\end{aligned}$$

where $\mathbf{K}_k = \mathbf{K}(k)$, ${}^{\mathcal{W}}\mathbf{P}_{\text{fus}}^k = {}^{\mathcal{W}}\mathbf{P}_{\text{fus}}(k)$, and $\mathbf{A}_{\gamma_i} = \mathbf{T}_i\boldsymbol{\Upsilon}\,{}^{\mathcal{T}}\mathbf{P}_i^{-1}\mathbf{T}_i^\top$, $\mathbf{A}_{\beta_i} = \boldsymbol{\Theta}\mathbf{T}_i\,{}^{\mathcal{T}}\mathbf{P}_i^{-1}\mathbf{T}_i^\top$ with

$$\boldsymbol{\Theta} = \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \boldsymbol{\Upsilon} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{bmatrix}.$$

REFERENCES

[1] R.E. Mahony, V. Kumar, and P.I. Corke. Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor. *IEEE Rob. Autom. Mag.*, 19(3):20–32, 2012.

[2] L. Doitsidis, S. Weiss, A. Renzaglia, M.W. Achtelik, E. Kosmatopoulos, R. Siegwart, and D. Scaramuzza. Optimal surveillance coverage for teams of micro aerial vehicles in GPS-denied environments using onboard vision. *Auton. Robot.*, 33(1–2):173–188, 2012.

[3] V. Kumar and N. Michael. Opportunities and challenges with autonomous micro aerial vehicles. *Int. J. Robot. Res.*, 31(11):1279–1291, 2012.

[4] F. Bullo, J. Cortés, and S. Martínez. *Distributed Control of Robotic Networks*. Princeton University Press, 2009.

[5] Y. Bar-Shalom, X.R. Li, and T. Kirubarajan. *Estimation with applications to tracking and navigation*. John Wiley & Sons, 2001.

[6] D. Uciński. *Optimal Measurement Methods for Distributed Parameter System Identification*. CRC Press, 2005.

[7] T.H. Chung, J.W. Burdick, and R.M. Murray. A decentralized motion coordination strategy for dynamic target tracking. In *Proc. IEEE Int. Conf. Robot. Automat*, pages 2416–2422, 2006.

[8] P. Yang, R.A. Freeman, and K.M. Lynch. Distributed cooperative active sensing using consensus filters. In *Proc. IEEE Int. Conf. Robot. Automat*, pages 405–410, 2007.

[9] K. Zhou and S.I. Roumeliotis. Optimal motion strategies for range-only constrained multisensor target tracking. *IEEE Trans. Robot.*, 24(5):1168–1185, 2008.

[10] E. Stump, V. Kumar, B. Grocholsky, and P.M. Shiroma. Control for Localization of Targets using Range-only Sensors. *Int. J. Robot. Res.*, 28(6):743–757, 2009.

[11] K. Zhou and S.I. Roumeliotis. Multirobot Active Target Tracking with Combinations of Relative Observations. *IEEE Trans. Robot.*, 27(4):678–695, 2011.

[12] B. Grocholsky, J. Keller, V. Kumar, and G. Pappas. Cooperative air and ground surveillance. *IEEE Rob. Autom. Mag.*, 13(3):16–25, 2006.

[13] F. Morbidi and G.L. Mariottini. On Active Target Tracking and Cooperative Localization for Multiple Aerial Vehicles. In *Proc. IEEE/RSJ Int. Conf. Intel. Robots Syst*, pages 2229–2234, 2011.

[14] P. Corke. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*, volume 73. Springer, 2011.

[15] D.S. Bernstein. *Matrix Mathematics: Theory, Facts, and Formulas*. Princeton University Press, 2009.

[16] T. Lee, M. Leoky, and N.H. McClamroch. Geometric tracking control of a quadrotor UAV on SE(3). In *Proc. 49th IEEE Conf. Dec. Contr*, pages 5420–5425, 2010.

[17] P. Corke. Robotics Toolbox for MATLAB Release 9. `http://www.petercorke.com/robot/`.

[18] K. Khoshelham and S.O. Elberink. Accuracy and Resolution of Kinect Depth Data for Indoor Mapping Applications. *Sensors*, 12(2):1437–1454, 2012.