2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Trajectory Optimization for Cooperatively Localizing Quadrotor UAVs

H S Helson Go[1] Hugh H.-T. Liu[1]

Abstract— **In this paper, an Active Cooperative Localization system for Quadrotor Unmanned Aerial Vehicles is developed. The optimal trajectories are determined by minimizing the uncertainty in position estimation by Extended Kalman Filter. In this system, a piecewise polynomial parameterization of trajectories is adopted for the optimizer, and the underlying state estimator is updated with appropriate models of sensors and quadrotor dynamics. This system is verified in extensive simulations in the scenario of a team of quadrotors with heterogeneous GNSS capabilities. These simulations answer an open question, showing that solving for trajectories by minimizing Kalman covariance computed in a noiseless environment is reasonable and that the optimized trajectories offer visible reductions in positioning uncertainty in the presence of noise.**

I. Introduction

Cooperative Localization (CL) is a promising strategy for navigating multi-robot systems (MRS). CL entails letting robots track each other and share sensor data to jointly estimate their states [1]–[3]. In the Unmanned Aerial Vehicle (UAV) context, CL has been successfully employed in scenarios such as target tracking [4], [5], mapping and exploration [6], and navigation in GNSS-degraded environments [7]–[11]. Within CL, an actively studied topic is Active Cooperative Localization, which refers to seeking the best trajectories or motion strategies for the robot team that maximize the quality of localization, variously embodied by position estimation certainty, likelihood of inter-vehicle sensing and communication, and amount of information gained from sensors [12]–[16].

A well-known approach towards active CL is minimizing the uncertainty of position estimation by an Extended Kalman Filter (EKF). Trawny and Barfoot [12] chose the logarithm of the determinant of the Kalman covariance matrix as the objective and used a numerical solver to compute the optimal trajectory. Two key issues were identified in this approach. The first issue, that an objective function based on Kalman covariance is nonconvex and potentially discontinuous, has been extensively researched. Hidaka *et. al.* [13] chose the trace of the steady-state Kalman covariance matrix as the objective in computing the optimal formation of a robot team and employed a Genetic Algorithm to solve this problem. Zhang *et. al.* [14] considered a leader-follower configuration and optimized the desired position of the follower over one single timestep, allowing the critical points of the EKF-based objective function to be evaluated analytically. More recently, alternative metrics such as generalized dilution of precision (gDOP) [10], [11] and local observability [16] have also been brought into consideration.

The second issue, that noise is neglected in the evaluation of the Kalman covariance-based objective function, is not as well studied. This practice is motivated by the desire to evaluate the Kalman covariance deterministically given a certain trajectory but leads to the question of whether the optimization objective accurately represents the estimation uncertainty once noise is factored in. To address this issue, trajectories generated by the active CL algorithms should be validated in comprehensive simulations with noise and a tracking controller in the navigation loop. Furthermore, to the authors' best knowledge, no simulation verification has been done in this manner in the area of active CL, leaving this question open.

To answer this question in simulations, an accurate physical model of the mobile robot and its sensors must be available. For UAVs, this means assumptions that physical models are linear [15], or that vehicles move on a horizontal plane [16] are untenable. Fortunately, research on CL for UAV teams has focused on developing state estimators [17]–[20], providing some solutions to the problem of modeling UAV dynamics and range/bearing sensors for CL. Nevertheless, CL systems for UAVs are rarely integrated with path planning and trajectory generation: active CL of UAV teams is primarily demonstrated on a specific configuration [8]–[11] where *father* UAVs enjoying accurate positioning play a role similar to GNSS satellites by guiding other *son* UAVs in a GNSS-degraded environment. As such, previous works' lack of experience in integrating UAV-specific motion and sensor models with Active CL poses another challenge on the implementational level.

The main contribution of this paper is a trajectory optimization algorithm for cooperative localizing quadrotor UAVs, based on the concept of minimizing EKF covariance, and incorporating two important features: 1) In the cost function, quadrotor trajectories are parameterized by piecewise polynomials per [21], reducing the dimension of the optimization problem and bypassing the need for costly numerical integration to propagate the motion model forward in time 2) In the underlying EKF, more detailed quadrotor dynamics and sensor models are implemented

These developments enable extensive simulations to be done to answer the open question about the validity of neglecting noise during the evaluation of Kalman-based objective functions for active CL. Results confirm that optimized trajectories are tracked faithfully and theoretical values of positioning accuracy can be largely achieved even when noise and external disturbances are accounted for.

[1]H S Helson Go and Hugh H.-T. Liu are with the Institute for Aerospace Studies, University of Toronto (UTIAS), 4925 Dufferin Street, Toronto, Canada hei.go@mail.utoronto.ca, hugh.liu@utoronto.ca

II. Problem Formulation

A. Quadrotor Dynamics Model

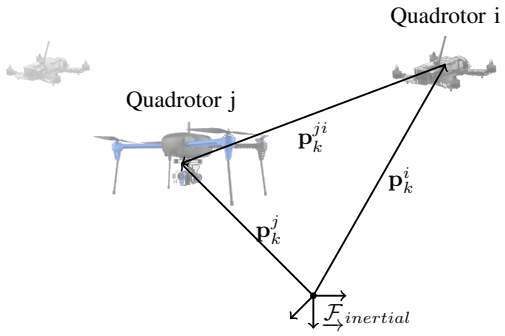


Fig. 1. Visualizing the Cooperative Localization System

The state of quadrotor i in a system of N_{uav} vehicles at time step k is given by $\mathbf{x}_k^i = \left(\mathbf{p}_k^i, \mathbf{q}_k^i, \mathbf{v}_k^i\right) \in \mathcal{X}^i \subset \mathbb{R}^{10}$, respectively the quadrotor's absolute position, the rotation of the inertial frame relative to the body frame expressed as a quaternion, and its absolute velocity. The control input is $\mathbf{u}_k^i = \left(f_k^i, \boldsymbol{\omega}_k^i\right) \in \mathcal{U}^i \subset \mathbb{R}^4$, respectively the quadrotor's thrust delivered along its body z-axis and its angular velocity.

The discrete-time motion model is given by

$$\begin{aligned}\mathbf{x}_k^i &= \mathbf{f}^i(\mathbf{x}_{k-1}^i, \mathbf{u}_k^i) \\ &\triangleq \begin{bmatrix} \mathbf{p}_{k-1}^i + T_k \mathbf{v}_k^i \\ \mathbf{q}_{k-1}^i \otimes \mathbf{q}\left(T_k \boldsymbol{\omega}_k^i\right) \\ \mathbf{v}_{k-1}^i + T_k \left(\frac{f_k^i}{m^i} \mathbf{R}\left(\mathbf{q}_{k-1}^i\right) \mathbf{1}_3 + \mathbf{g}\right) \end{bmatrix}, \end{aligned} \tag{1}$$

where m^i is the quadrotor mass, $\mathbf{g}$ the gravity vector, T_k the time interval, and $\mathbf{1}_3 = (0, 0, 1)$. Also, $\otimes : \mathbb{H} \times \mathbb{H} \to \mathbb{H}$ denotes the quaternion product and $\mathbf{R} : \mathbb{H} \to SO(3)$ the rotation matrix equivalent to a quaternion. Furthermore, $\mathbf{q}\left(T_k \boldsymbol{\omega}_k^i\right) = \exp\left(\frac{T_k \boldsymbol{\omega}_k^i}{2}\right)$ represents the infinitesimal rotation under the effect of $\boldsymbol{\omega}_k^i$ per Sola's [22] notation for integrating quaternions aided by the *quaternion exponential map*. Similarly, when dynamics or observation models are linearized, an approach inspired by Sola's means of linearizing w.r.t. a rotational error 3-vector is adopted.

The Jacobian of the motion model w.r.t. the state, evaluated at the current state estimate $\hat{\mathbf{x}}_k^i$ with control inputs $\mathbf{u}_k^i$, is

$$\mathbf{F}_k^i = \begin{bmatrix} \mathbf{1} & \mathbf{0} & T_k \mathbf{1} \\ \mathbf{0} & \mathbf{R}\big(\mathbf{q}\left(T_k \boldsymbol{\omega}_k^i\right)\big)^\top & \mathbf{0} \\ \mathbf{0} & -\frac{T_k f_k^i}{m^i} \mathbf{R}\left(\mathbf{q}_{k-1}^i\right) \mathbf{1}_3^\times & \mathbf{1} \end{bmatrix}. \tag{2}$$

The Jacobian of the motion model w.r.t. the inputs, similarly evaluated at $\left(\hat{\mathbf{x}}_k, \mathbf{u}_k^i\right)$ is given by

$$\mathbf{G}_k^i = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & T_k \mathbf{1} \\ \frac{T_k}{m^i} \mathbf{R}\left(\mathbf{q}_{k-1}^i\right) \mathbf{1}_3 & \mathbf{0} \end{bmatrix}. \tag{3}$$

The inputs $\mathbf{u}_k^i$ are affected by noise, assumed to be drawn from a zero-mean multivariate Gaussian distribution, for a diagonal covariance matrix $\mathbf{Q}_k^i$.

The full system has state $\mathbf{x}_k = \left(\mathbf{x}_k^1, \ldots, \mathbf{x}_k^N\right)$ and inputs $\mathbf{u}_k = \left(\mathbf{u}_k^1, \ldots, \mathbf{u}_k^N\right)$, defined in the state space $\mathcal{X} \subset \mathbb{R}^{10 N_{\text{uav}}}$ and input space $\mathcal{U} \subset \mathbb{R}^{4 N_{\text{uav}}}$ respectively.

Therefore, the motion model for the system comprises the following block matrices:

$$\mathbf{f}\left(\mathbf{x}_k, \mathbf{u}_k\right) = \begin{bmatrix} \mathbf{f}^1\left(\mathbf{x}_k^1, \mathbf{u}_k^1\right) \\ \vdots \\ \mathbf{f}^N\left(\mathbf{x}_k^N, \mathbf{u}_k^N\right) \end{bmatrix}, \tag{4}$$

$$\mathbf{F}_k = \operatorname{diag}\left(\mathbf{F}_k^1, \ldots, \mathbf{F}_k^N\right), \tag{5}$$

$$\mathbf{G}_k = \operatorname{diag}\left(\mathbf{G}_k^1, \ldots, \mathbf{G}_k^N\right), \tag{6}$$

$$\mathbf{Q}_k = \operatorname{diag}\left(\mathbf{Q}_k^1, \ldots, \mathbf{Q}_k^N\right). \tag{7}$$

Based on these definitions, the covariance matrix is updated during the EKF prediction step by

$$\check{\mathbf{P}}_k = \mathbf{F}_k \hat{\mathbf{P}}_{k-1} \mathbf{F}_k^\top + \mathbf{G}_k \mathbf{Q}_k \mathbf{G}_k^\top. \tag{8}$$

B. Observation Model

Three sources of exteroceptive measurements are considered. Firstly, each quadrotor is provided with measurements of its position by GNSS, though of heterogeneous capability and quality. The position observation equation is

$$\mathbf{h}^{\text{pos}}(\mathbf{x}_k) = \begin{bmatrix} {\mathbf{p}_k^1}^\top & \cdots & {\mathbf{p}_k^{N_{\text{uav}}}}^\top \end{bmatrix}^\top. \tag{9}$$

The Jacobian of (9) w.r.t. the state, denoted by $\mathbf{H}_k^{\text{pos}}$, consists of 3-by-3 identity matrices at locations corresponding to each quadrotor's position coordinates.

Secondly, each quadrotor is assumed to directly measure its attitude. The attitude observation equation is

$$\mathbf{h}^{\text{att}}\left(\mathbf{x}_k\right) = \begin{bmatrix} {\mathbf{q}_k^1}^\top & \cdots & {\mathbf{q}_k^{N_{\text{uav}}}}^\top \end{bmatrix}. \tag{10}$$

The Jacobian of (10) w.r.t. the state is denoted by $\mathbf{H}_k^{\text{att}}$ and likewise consists of 3-by-3 identity matrices at locations corresponding to the attitude coordinate of each quadrotor.

The internal mechanics of IMU and GNSS are not modelled in (9) and (10) in what could be called a *loosely-coupled* approach. This highlights the impact on CL performance by interrobot sensing apart from other sensors.

Thirdly, each quadrotor tracks its teammates and measures their positions relative to itself. Both sensor *extrinsics* and *intrinsics* are modeled.

In the context of CL, the extrinsics describe how the target quadrotor's position is expressed inside the sensor's frame of reference given extrinsic parameters $\{\mathbf{R}_s, \mathbf{r}_s\}$ that describe the pose of the sensor in the body frame, through

$$\boldsymbol{\rho}_k^{ji} = \boldsymbol{\rho}\left(\mathbf{p}_k^i, \mathbf{p}_k^j\right) \triangleq \mathbf{R}_s \mathbf{p}_k^{ji} + \mathbf{p}_s, \tag{11}$$

where $\mathbf{p}_k^{ji} = \mathbf{R}_k^i \left(\mathbf{p}_k^j - \mathbf{p}_k^i\right)$.

The intrinsics describe how $\boldsymbol{\rho}_k^{ji}$ is expressed in the sensor's internal coordinates. Here, these coordinates are azimuth and altitude angles $\left(\theta_k^{ji}, \varphi_k^{ji}\right)$, giving

$$\begin{bmatrix} \theta_k^{ji} \\ \varphi_k^{ji} \end{bmatrix} = \mathbf{h}^{ji}\left(\boldsymbol{\rho}_k^{ji}\right) \triangleq \begin{bmatrix} \operatorname{atan2}\left(\boldsymbol{\rho}_{k\ 2}^{ji}, \boldsymbol{\rho}_{k\ 1}^{ji}\right) \\ \operatorname{atan2}\left(\boldsymbol{\rho}_{k\ 3}^{ji}, \|\boldsymbol{\rho}_{k\ 1:2}^{ji}\|\right) \end{bmatrix}. \tag{12}$$

Note that (12) is a generic bearings-only model inspired by [20] that can be implemented in terms of monocular vision or scanning rangefinders. Limitations in sensor range or field-of-view are not modeled. Said limitations' impact on CL performance is orthogonal to this paper's focus and treated in a separate class of works on connectivity optimization [15]. Thus, the complete interrobot tracking model is

$$\mathbf{h}^{\mathrm{rel}}(\mathbf{x}_k) = \begin{bmatrix} \cdots & \theta_k^{ji} & \varphi_k^{ji} & \cdots \end{bmatrix}^\top,\ (i,j) \in \binom{N_{\mathrm{uav}}}{2}. \tag{13}$$

The Jacobian of (13) w.r.t. the state is

$$\mathbf{H}_k^{\mathrm{rel}} = \begin{bmatrix} \mathbf{H}_+^{21} & \mathbf{H}_-^{21} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{H}_+^{31} & \mathbf{0} & \mathbf{H}_-^{31} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathbf{H}_+^{N1} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} & \mathbf{H}_-^{N1} \end{bmatrix}, \tag{14}$$

where

$$\mathbf{H}_+^{ji} = \frac{\partial \boldsymbol{\rho}_k^{ji}}{\partial \mathbf{x}_k^i} = \boldsymbol{\Phi}_k^{ji} \begin{bmatrix} -\mathbf{R}_k^{i\top} & \mathbf{0} & \boldsymbol{\rho}_k^{ji\times} \end{bmatrix},$$

$$\mathbf{H}_-^{ji} = \frac{\partial \boldsymbol{\rho}_k^{ji}}{\partial \mathbf{x}_k^j} = \boldsymbol{\Phi}_k^{ji} \begin{bmatrix} \mathbf{R}_k^{i\top} & \mathbf{0} & \mathbf{0} \end{bmatrix},$$

$$\boldsymbol{\Phi}_k^{ji} = \frac{\partial \mathbf{h}^{ji}}{\partial \boldsymbol{\rho}_k^{ji}} = \begin{bmatrix} \frac{-\boldsymbol{\rho}_2^{ji}}{\|\boldsymbol{\rho}_{1:2}^{ji}\|^2} & \frac{\boldsymbol{\rho}_1^{ji}}{\|\boldsymbol{\rho}_{1:2}^{ji}\|^2} & 0 \\ -\frac{\boldsymbol{\rho}_1^{ji}\boldsymbol{\rho}_3^{ji}}{\|\boldsymbol{\rho}^{ji}\|^2\|\boldsymbol{\rho}_{1:2}^{ji}\|} & -\frac{\boldsymbol{\rho}_1^{ji}\boldsymbol{\rho}_2^{ji}}{\|\boldsymbol{\rho}^{ji}\|^2\|\boldsymbol{\rho}_{1:2}^{ji}\|} & \frac{\|\boldsymbol{\rho}_{1:2}^{ji}\|}{\|\boldsymbol{\rho}^{ji}\|^2} \end{bmatrix}.$$

The observation model for the full system comprises the following block matrices

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) \triangleq \begin{bmatrix} \mathbf{h}^{\mathrm{pos}}(\mathbf{x}_k) \\ \mathbf{h}^{\mathrm{att}}(\mathbf{x}_k) \\ \mathbf{h}^{\mathrm{rel}}(\mathbf{x}_k) \end{bmatrix},\ \mathbf{H}_k = \begin{bmatrix} \mathbf{H}_k^{\mathrm{pos}} \\ \mathbf{H}_k^{\mathrm{att}} \\ \mathbf{H}_k^{\mathrm{rel}} \end{bmatrix}. \tag{15}$$

where the monolithic nature of $\mathbf{h}(\mathbf{x}_k)$ reflects that all sensors are assumed to be sampled synchronously. Since low-level sensor mechanics are not modeled, individual sensor latencies are disregarded.

The observations $\mathbf{z}_k$ are affected by noise, assumed to be drawn from a zero-mean multivariate Gaussian distribution, for a diagonal covariance matrix $\mathbf{R}_k$.

Based on these definitions, the covariance matrix is updated during the EKF update step by

$$\hat{\mathbf{P}}_k = (\mathbf{1} - \mathbf{K}_k\mathbf{H}_k)\,\check{\mathbf{P}}_k. \tag{16}$$

where the Kalman gain is $\mathbf{K}_k = \check{\mathbf{P}}_k\mathbf{H}_k^\top(\mathbf{H}_k\check{\mathbf{P}}_k\mathbf{H}_k^\top + \mathbf{R})^{-1}$.

If state correction is required, the state update vector is computed by $\mathbf{K}_k\check{\mathbf{x}}_k$. Simple addition is used to apply corrections to vector quantities. As for quaternion corrections, refer to [22] for details.

III. The Optimization Problem

A. Differentially Flat Outputs

Directly using control commands, e.g. desired linear velocity, as trajectory parameters are not appropriate for quadrotors since it may result in dynamically infeasible trajectories that cannot be accurately or even safely followed. To address this issue, trajectory points are represented in a differentially flat form [21], such that a flat output $\mathbf{y}_k \in \mathbb{R}^{m_f}$ and its derivatives fully parameterize the state and inputs of a system by

$$\mathbf{x}_k = \boldsymbol{\zeta}\left(\mathbf{y}_k, \dot{\mathbf{y}}_k, \ldots, \mathbf{y}_k^{(r-1)}\right), \tag{17}$$

$$\mathbf{u}_k = \boldsymbol{\psi}\left(\mathbf{y}_k, \dot{\mathbf{y}}_k, \ldots, \mathbf{y}_k^{(r)}\right), \tag{18}$$

where $\boldsymbol{\zeta} : (\mathbb{R}^{m_f})^{r-1} \to \mathbb{R}^m$ and $\boldsymbol{\psi} : (\mathbb{R}^{m_f})^r \to \mathbb{R}^n$ are termed *flatness maps*.

For a quadrotor, the flat outputs comprise its absolute position and yaw angle

$$\mathbf{y}_k = \begin{bmatrix} \mathbf{p}_k^\top & \psi_k \end{bmatrix}^\top, \tag{19}$$

and aerodynamic disturbance-aware flatness maps from the GCOPTER project [23] are used to convert a sequence of flat outputs to real states and inputs.

B. Piecewise-Polynomial Trajectory Representation

To represent a continuous trajectory with a finite number of parameters, piecewise polynomials are used following common practice in quadrotor trajectory generation [21], [24].

$$\mathbf{y}(t, \mathcal{P}) = \{\boldsymbol{\mathcal{P}}_1\mathbf{t}(t), \ldots, \boldsymbol{\mathcal{P}}_M\mathbf{t}(t)\}, \tag{20}$$

where each piece out of M pieces, corresponding to the time interval $[T_k, T_{k+1})$, is a n-dimensional polynomial of degree d comprising coefficient matrix $\boldsymbol{\mathcal{P}}_k \in \mathbb{R}^{n\times d+1}$ and time vector $\mathbf{t}(t) \triangleq \begin{bmatrix} 1 & t & \cdots t^{d-1} \end{bmatrix}^\top$. The complete set of polynomial coefficients is $\mathcal{P} = \{\boldsymbol{\mathcal{P}}_1, \ldots, \boldsymbol{\mathcal{P}}_M\}$.

In a discrete-time context, Equation (20) is evaluated (*c.f. sampled*) at arbitrary times for a sequence of flat outputs.

The position, velocity and acceleration at the initial and final points of the trajectory are fixed by linear constraints given by

$$\begin{aligned} \boldsymbol{\mathcal{P}}_0\mathbf{t}(T_0) &= \mathbf{p}_0, & \boldsymbol{\mathcal{P}}_M\mathbf{t}(T_M) &= \mathbf{p}_M, \\ \boldsymbol{\mathcal{P}}_0\dot{\mathbf{t}}(T_0) &= \mathbf{v}_0, & \boldsymbol{\mathcal{P}}_M\dot{\mathbf{t}}(T_M) &= \mathbf{v}_M, \\ \boldsymbol{\mathcal{P}}_0\ddot{\mathbf{t}}(T_0) &= \mathbf{a}_0, & \boldsymbol{\mathcal{P}}_M\ddot{\mathbf{t}}(T_M) &= \mathbf{a}_M. \end{aligned} \tag{21}$$

If a quadrotor is required to pass through intermediate waypoints, said waypoints are specified as the intersection of trajectory pieces whose locations are fixed in space by

$$\boldsymbol{\mathcal{P}}_k\mathbf{t}(T_k) = \mathbf{p}_{\mathrm{waypoint,k}}. \tag{22}$$

Furthermore, the trajectory is constrained to be continuous up to the 2nd derivative by

$$\begin{aligned} \boldsymbol{\mathcal{P}}_k\mathbf{t}(T_k) &= \boldsymbol{\mathcal{P}}_{k+1}\mathbf{t}(T_k), \\ \boldsymbol{\mathcal{P}}_k\dot{\mathbf{t}}(T_k) &= \boldsymbol{\mathcal{P}}_{k+1}\dot{\mathbf{t}}(T_k), \\ \boldsymbol{\mathcal{P}}_k\ddot{\mathbf{t}}(T_k) &= \boldsymbol{\mathcal{P}}_{k+1}\ddot{\mathbf{t}}(T_k), \end{aligned} \tag{23}$$

which ensures that the position, velocity and acceleration at the intersection of trajectory pieces are matching.

The means to express Equations (21) through (23) as linear mappings affine in the polynomial coefficients are given in [24] and not repeated here for brevity.

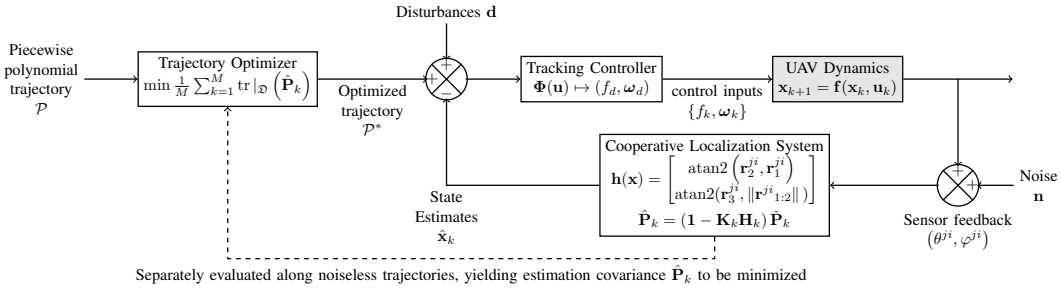


Fig. 2. Block diagram showing the configuration of the simulation where the Cooperative Navigation System is inside the navigation loop together with the vehicle dynamics model, a tracking controller, and injection of disturbances and noise

C. The Optimization Objective

The cost function for trajectory optimization is built around a *quality-of-localization* score

$$J_{\mathrm{QoLo}}\left(\mathcal{P}\right) = \frac{1}{M}\sum_{k=1}^{M} \mathrm{tr}\,|_{\mathfrak{D}}\left(\hat{\mathbf{P}}_k\right), \tag{24}$$

where $\mathrm{tr}\,|_{\mathfrak{D}}$ denotes taking trace in a subspace $\mathfrak{D}$ of the state space $\mathcal{X}$. Essentially, only specific diagonal elements of $\hat{\mathbf{P}}_k$ corresponding to coordinates in this subspace are summed by the trace. In this paper, $\mathfrak{D}$ is the space of quadrotor positions since only the position estimation covariance is considered.

When J_{QoLo} is evaluated, the piecewise-polynomial $\mathcal{P}$ is sampled at M points in time yielding a sequence of flat outputs $\{\mathbf{y}_k\}_1^M$. Next, the flat outputs are translated to a noiseless trajectory of states and inputs by the flatness maps $\boldsymbol{\psi}$ and $\boldsymbol{\zeta}$. The quadrotors are specified to not yaw at all in the absence of a suitable approach towards CL-aware yaw planning. Then, the covariance matrix $\hat{\mathbf{P}}_k$ is updated recursively along this trajectory by the EKF equations (8) and (16). By the nature of the EKF, they are sufficient to describe the development of estimation covariance along a trajectory; The state propagation equations and real sensor feedback are unnecessary. Evaluated in the absence of noise, the value of J_{QoLo} is deterministic for a given $\mathcal{P}$.

Lastly, J_{QoLo} computes the mean position estimation variance over the quadrotor team and every time step. Minimizing it gives the problem

$$\min_{\mathcal{P}} J_{\mathrm{QoLo}}(\mathcal{P}), \tag{25}$$

$$\text{subject to} \begin{cases} \mathbf{u}_{\min} \le \mathbf{u}_k^i \le \mathbf{u}_{\max}, & \text{(actuator limits)} \\ z_{\min} \le z_k^i \le z_{\max}, & \text{(altitude bounds)} \\ r_{\min} \le \left\|\mathbf{p}_k^{ji}\right\| \le r_{\min}, & \text{(interrobot distance)} \\ \mathbf{p}_0^i = \mathbf{p}_{\mathrm{home}}^i, & \\ \mathbf{v}_0^i = \mathbf{a}_0^i = \mathbf{0}, & \text{(Initial Conditions)} \\ \mathbf{p}_M^i = \mathbf{p}_{\mathrm{destination}}^i, & \\ \mathbf{v}_M^i = \mathbf{a}_M^i = \mathbf{0}, & \text{(Terminal Conditions)} \end{cases} \tag{26}$$

In addition to the constraints expressed above, constraint (23) ensures continuity of the trajectory; Intermediate position constraints enforced by (22) are specified on a permission basis.

This optimization problem is solved by `fmincon` in MATLAB and the `GlobalSearch` global optimization solver is used to refine the solution.

IV. Tracking Control

Once Problem (25) is solved, the resultant trajectories are tracked by a nonlinear Model Predictive Controller (NMPC), which solves an optimal control problem by minimizing

$$L\left(\mathbf{x}, \mathbf{u}\right) = \tilde{\mathbf{x}}_N^\top \boldsymbol{\mathcal{Q}}_N \tilde{\mathbf{x}}_N + \sum_{k=0}^{N-1} \tilde{\mathbf{x}}_k^\top \boldsymbol{\mathcal{Q}} \tilde{\mathbf{x}}_k + \tilde{\mathbf{u}}_k^\top \boldsymbol{\mathcal{R}}_k \tilde{\mathbf{u}}_k, \tag{27}$$

$$\text{where } \tilde{\mathbf{x}}_k = \mathbf{x}_k - \mathbf{x}_{\mathrm{ref},k} \quad \tilde{\mathbf{u}}_k = \mathbf{u}_k - \mathbf{u}_{\mathrm{ref},k},$$

subject to the system dynamics (1) The NMPC is solved under the *acados* [25] framework.

V. Results

The performance of this Active CL framework is evaluated in two simulation test cases on a team of three quadrotor UAVs (*i.e.* $N_{\mathrm{uav}} = 3$) with heterogeneous GNSS capabilities.

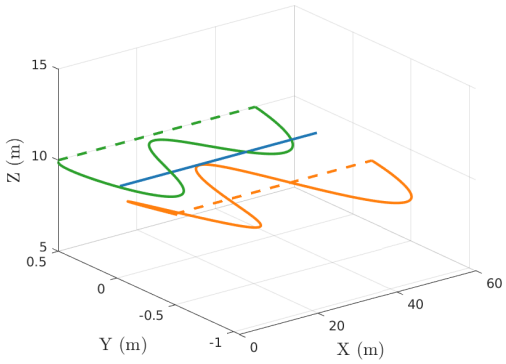
In all cases, the quadrotors are initially arranged in an equilateral triangle with 1m sides. As configured in Table I, the central quadrotor is equipped with significantly more accurate GNSS (e.g. RTK); Its trajectory is fixed *a-priori* for each test case and not optimized. Its counterparts on the left and right wings are equipped with baseline GNSS systems. CL plays a primary role in their localization and their trajectories are freely optimized.

TABLE I
Variances for underlying EKF of active cooperative localization system

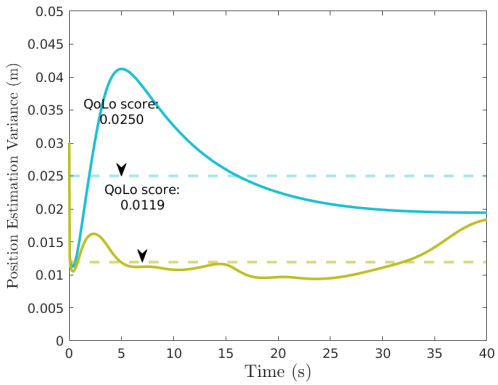
	Parameters	Values
Process noise	$\sigma^2\left(f_k^i\right)$	0.1
	$\sigma^2\left(\boldsymbol{\omega}_k^i\right)$	$(0.001, 0.001, 0.002)$
Sensor noise	$\sigma^2\left(\mathbf{p}_k^{\mathrm{HiPrecGNSS}}\right)$	$(0.05, 0.05, 0.1)$
	$\sigma^2\left(\mathbf{p}_k^i\right)$	$(5, 5, 10)$
	$\sigma^2\left(\boldsymbol{\phi}_k^i\right)$	$(0.01, 0.01, 0.01)$[1]
	$\sigma^2\left(\theta_k^{ji}, \varphi_k^{ji}\right)$	$\left(10^{-6}, 10^{-6}\right)$[2]

[1]For each rotational degree of freedom (out of 3)

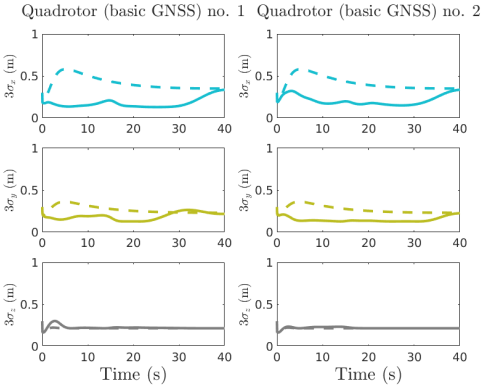
[2]Smallest figure (most precise sensor) found in the literature [26] ($0.001rad$ std. dev.) to emphasize the trust placed in interrobot tracking



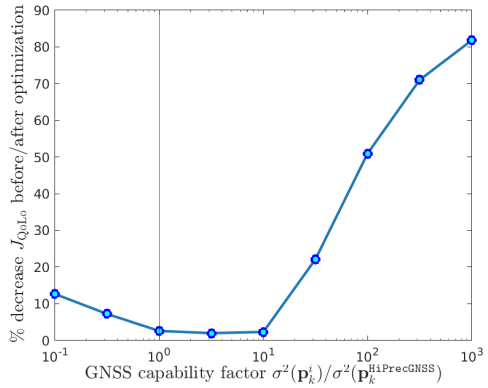
(a) Trajectories of the quadrotor team. Dashed lines are initial trajectories (straight lines and uniform velocity); Solid lines are optimized trajectories. The central (zeroth) quadrotor has hi-precision GNSS and its trajectory (blue) is fixed.



(b) The magnitude of the weighted average of position estimation covariance of every quadrotor (solid curves) along the reference (blue) and the optimized (orange) trajectories. The mean of this value over time is the quality-of-localization score (dashed lines).



(c) Standard deviation of position estimation on quadrotors with basic GNSS along the reference (blue) and optimized trajectories (orange).



(d) Percentage improvement in quality-of-localization as confidence in GNSS data on quadrotors with basic GNSS worsens relative to their counterpart enjoying high-precision GNSS-aiding

Fig. 3. Quadrotor trajectories and position estimation performance before and after trajectory optimization

Test Case 1: Applies the trajectory optimization algorithm to a reference trajectory, then compares the development of position estimation uncertainty and the quality-of-localization metric J_{QoLo} along each trajectory.

Here, the quadrotors attempt to reach a destination 40 meters away down the x-axis. The reference trajectories are *uniformly straight* and used as the initial guess for the solver.

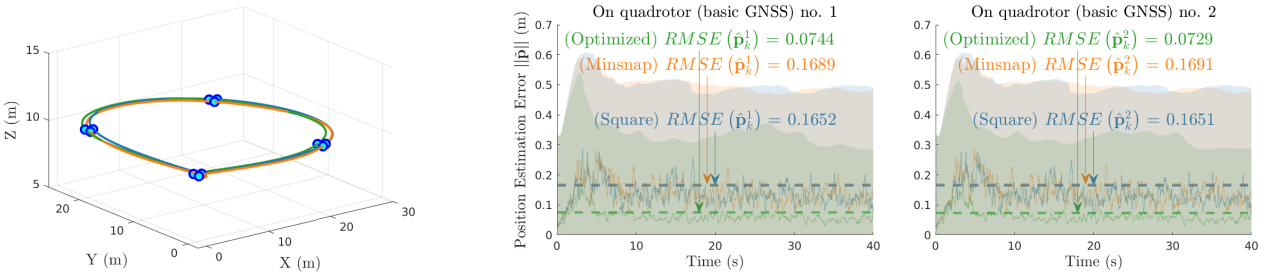
Figures 3a shows the reference and optimized trajectories of the quadrotor team, while Figures 3b and 3c demonstrates the improvement in position estimation accuracy brought by the optimized trajectories. Both the standard deviation of position estimation and J_{QoLo} show that when the optimized trajectories are taken, uncertainty in position estimation is maintained at a consistent low for the greater part of a mission of 40 seconds. It only rebounded near the end when the vehicles returned to a fixed formation at the endpoint.

The optimized trajectories let the quadrotors break off from a rigid formation since the straight reference trajectory caused the position estimation covariance to rise sharply as shown in Figure 3b, This is due to the nature of 2-degree-of-freedom sensors such as bearing sensors or monocular cameras: the distance to a target (*c.f. depth*) is only observable when there is apparent relative motion between them.

Test case 1 is also repeated while a relative GNSS capability factor $\sigma^2(\mathbf{p}_k^i)/\sigma^2\left(\mathbf{p}_k^{\text{HiPrecGNSS}}\right)$ is varied from 0.1 to 1000. Figure 3d shows that once confidence in GNSS on some quadrotors becomes more than 10 times worse than that on their counterpart enjoying high precision GNSS, trajectory optimization bolsters localization quality by up to 80%.

Test Case 2: Applies trajectory optimization to a reference trajectory, then simulate a team of quadrotors flying along those trajectories as configured in Figure 2, where the CL system is placed in the loop along with a quadrotor simulator, tracking controller, and simulated noise and disturbances. Up to 20 simulation trials are conducted for each trajectory.

Here, the quadrotor team navigates to four waypoints arranged in a square with 10m sides. Two sets of trajectories were used as references: 1) A square, which reflects trajectories that open-source autopilots will take, but is intuitively detrimental to interrobot tracking due to its uniformity. 2) A trajectory generated by a minimum-snap trajectory generator, which is also used as the initial guess for the solver in the



(a) Square (dotted), minimum-snap (dashed), and optimized (solid) trajectories of the quadrotor team; Differences appear small due to scale; See performance comparison on the right

(b) Magnitude of estimation error (solid lines), RMS error (dashed lines) and confidence (shaded areas), obtained along the square (blue), minimum-snap (orange line and grayish-orange area), and optimized (blue) trajectories. Results are averaged over 20 trials

Fig. 4. Trajectories and position estimation performance of the Cooperative Localization System in Test Case 2

hopes it will be better than the square. These trajectories and the optimization solution are shown in Figure 4a.

TABLE II
PERFORMANCE OF ACTIVE CL IN TEST CASE 2

Value	Trajectory	min	max	mean
RMSE $\left(\hat{\mathbf{p}}_k^1\right)$	Square	0.1245	0.1960	0.1652
	Minsnap	0.1408	0.1949	0.1689
	Optimized	**0.0684**	**0.0828**	**0.0744**
RMSE $\left(\hat{\mathbf{p}}_k^2\right)$	Square	0.1231	0.1988	0.1651
	Minsnap	0.1431	0.1934	0.1691
	Optimized	**0.0664**	**0.0824**	**0.0729**
% diff. in $\int \left\|\sigma\left(\hat{\mathbf{p}}_k^1\right)^2\right\| dt$	Optimized	2.3140	5.4401	3.7614
% diff. in $\int \left\|\sigma\left(\hat{\mathbf{p}}_k^2\right)^2\right\| dt$	Optimized	1.9620	5.0220	3.2578

Figure 4b displays the improvement in position estimation precision and accuracy along the optimized trajectory. The EKF that underlies the CL system is convergent in all cases, barring initial spikes in estimation errors lasting until 10s.

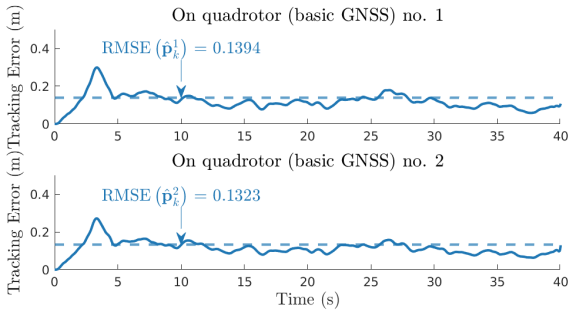


Fig. 5. Magnitude of position tracking error by NMPC

Quantitatively, the first section in Table II shows that the optimized trajectories always achieve the lowest position estimation error and consistently improve RMS error of position estimation by $\approx 50\%$. Surprisingly, position estimation performance along the minimum-snap trajectory is as bad as that along the square trajectory. This shows that the lack of relative motion between vehicles, instead of the 'straightness' of trajectories, contributes to worse CL performance.

Furthermore, the estimation uncertainty along the optimized trajectory closely matches its noiseless counterpart computed during the evaluation of the objective function J_{QoLo}. The second section in Table II shows that the cumulative magnitude of estimation variance with/without noise differs by a mere percent. The NMPC tracking controller lets the quadrotors faithfully follow the solutions of trajectory optimization, achieving errors on the order of a tenth of a meter as shown in Figure 5. This answers the question that in Active CL, the objective function in trajectory optimization *can* be reasonably computed along noiseless trajectories.

Throughout the simulations, evaluating J_{QoLo} consumes $0.0757 \pm 0.0081 s$[1], making for a $0.0561 \pm 0.012 s / \dim \mathcal{P}$ cost for each `fmincon` iteration. The time taken to solve (25) is on the order of minutes, and this cost scales linearly with $\dim \mathcal{P}$ and superlinearly with N_{uav} as the latter drives up the size of all matrices in the EKF. This precludes real-time replanning or solving (25) in a moving-horizon manner and is noted as a factor that limits the current method's potential.

VI. CONCLUSION

In this paper, an Active Cooperative Localization system tailored to quadrotor UAVs that builds upon previous covariance-minimizing trajectory optimizers is developed.

Extensive simulations demonstrated the effectiveness of this system and answered an open question on the validity of optimizing quality-of-localization metrics computed noiselessly. They pave the way for further simulations with concrete sensor models and eventually real-world experiments, which the authors are planning to do in future work.

Finally, the covariance minimization approach is somewhat limited by the the computational cost of the underlying EKF. A more recent approach, *observability*-based quality of localization, potentially sidesteps this issue. Investigating it presents a great opportunity for future theoretical studies.

ACKNOWLEDGMENT

The authors would like to acknowledge the sponsorship by the Natural Sciences and Engineering Research Council of Canada (NSERC) under grant RGPIN-2023-05148.

[1]Computational times are reported as mean ± std. dev.

REFERENCES

[1] Ryo Kurazume, Shigemi Nagata, and Shigeo Hirose. "Cooperative positioning with multiple robots". In: *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*. IEEE. 1994, pp. 1250–1257.

[2] Ioannis Rekleitis, Gregory Dudek, and Evangelos Milios. "Multi-robot collaboration for robust exploration". In: *Annals of Mathematics and Artificial Intelligence* 31.1 (2001), pp. 7–40.

[3] Ioannis M Rekleitis, Gregory Dudek, and Evangelos E Milios. "Multi-robot cooperative localization: a study of trade-offs between efficiency and accuracy". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Vol. 3. IEEE. 2002, pp. 2690–2695.

[4] Ke Zhou and Stergios I Roumeliotis. "Multirobot active target tracking with combinations of relative observations". In: *IEEE Transactions on Robotics* 27.4 (2011), pp. 678–695.

[5] Fabio Morbidi and Gian Luca Mariottini. "Active target tracking and cooperative localization for teams of aerial vehicles". In: *IEEE transactions on control systems technology* 21.5 (2012), pp. 1694–1707.

[6] Robert Grabowski, Luis E Navarro-Serment, Christiaan JJ Paredis, et al. "Heterogeneous teams of modular robots for mapping and exploration". In: *Autonomous Robots* 8.3 (2000), pp. 293–308.

[7] Yaohong Qu and Youmin Zhang. "Cooperative localization against GPS signal loss in multiple UAVs flight". In: *Journal of Systems Engineering and Electronics* 22.1 (2011), pp. 103–112.

[8] Amedeo Rodi Vetrella, Roberto Opromolla, Giancarmine Fasano, et al. "Autonomous flight in GPS-challenging environments exploiting multi-UAV cooperation and vision-aided navigation". In: *AIAA Information Systems-AIAA Infotech Aerospace*. 2017, p. 0879.

[9] Flavia Causa, Amedeo Rodi Vetrella, Giancarmine Fasano, et al. "Multi-UAV formation geometries for cooperative navigation in GNSS-challenging environments". In: *2018 IEEE/ION position, location and navigation symposium (PLANS)*. 2018, pp. 775–785.

[10] Flavia Causa, Marija Popovi, Giancarmine Fasano, et al. "Navigation aware planning for tandem UAV missions in GNSS challenging Environments". In: *AIAA Scitech 2019 Forum*. 2019, p. 0916.

[11] Flavia Causa and Giancarmine Fasano. "Improving navigation in GNSS-challenging environments: Multi-UAS cooperation and generalized dilution of precision". In: *IEEE Transactions on Aerospace and Electronic Systems* 57.3 (2020), pp. 1462–1479.

[12] Nikolas Trawny and Timothy Barfoot. "Optimized motion strategies for cooperative localization of mobile robots". In: *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*. Vol. 1. IEEE. 2004, pp. 1027–1032.

[13] Yukikazu S Hidaka, Anastasios I Mourikis, and Stergios I Roumeliotis. "Optimal formations for cooperative localization of mobile robots". In: *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE. 2005, pp. 4126–4131.

[14] Xun S. Zhou, Ke X. Zhou, and Stergios I. Roumeliotis. "Optimized motion strategies for localization in leader-follower formations". In: Institute of Electrical and Electronics Engineers (IEEE), Dec. 2011, pp. 98–105. DOI: 10.1109/iros.2011.6095169.

[15] Liang Zhang, Zexu Zhang, Roland Siegwart, et al. "A connectivity-prediction algorithm and its application in active cooperative localization for multi-robot systems". In: *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2020, pp. 9824–9830.

[16] Rohith Boyinine, Rajnikant Sharma, and Kevin Brink. "Observability based path planning for multi-agent systems to aid relative pose estimation". In: *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE. 2022, pp. 912–921.

[17] Guoqiang Mao, Sam Drake, and Brian DO Anderson. "Design of an extended kalman filter for uav localization". In: *2007 Information, Decision and Control*. IEEE. 2007, pp. 224–229.

[18] Rajnikant Sharma and Clark Taylor. "Cooperative navigation of MAVs in GPS denied areas". In: *2008 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*. IEEE. 2008, pp. 481–486.

[19] Anusna Chakraborty, Clark N Taylor, Rajnikant Sharma, et al. "Cooperative localization for fixed wing unmanned aerial vehicles". In: *2016 IEEE/ION Position, Location and Navigation Symposium (PLANS)*. IEEE. 2016, pp. 106–117.

[20] Anusna Chakraborty, Rajnikant Sharma, and Kevin Brink. "Cooperative localization for multi-rotor UAVs". In: *AIAA Scitech 2019 Forum*. 2019, p. 0684.

[21] Daniel Mellinger and Vijay Kumar. "Minimum snap trajectory generation and control for quadrotors". In: *2011 IEEE international conference on robotics and automation*. IEEE. 2011, pp. 2520–2525.

[22] Joan Sola. "Quaternion kinematics for the error-state Kalman filter". In: *arXiv preprint arXiv:1711.02508* (2017).

[23] Zhepei Wang, Xin Zhou, Chao Xu, et al. "Geometrically constrained trajectory optimization for multicopters". In: *IEEE Transactions on Robotics* 38.5 (2022), pp. 3259–3278.

[24] Adam Bry, Charles Richter, Abraham Bachrach, et al. "Aggressive flight of fixed-wing and quadrotor aircraft in dense indoor environments". In: *The International Journal of Robotics Research* 34.7 (2015), pp. 969–1002.

[25] Robin Verschueren, Gianluca Frison, Dimitris Kouzoupis, et al. “acados – a modular open-source framework for fast embedded optimal control”. In: *Mathematical Programming Computation* (Oct. 2021). ISSN: 1867-2957. DOI: 10.1007/s12532-021-00208-8. URL: https://doi.org/10.1007/s12532-021-00208-8.

[26] Qifan Yang, Xuerong Yang, Yajun Yang, et al. “Distributed cooperative localization based on bearing-only sensors”. In: *IEEE Sensors Journal* 21.20 (2021), pp. 23645–23657.