

# Range-Visual-Inertial Sensor Fusion for Micro Aerial Vehicle Localization and Navigation

Abhishek Goudar, Wenda Zhao, and Angela P. Schoellig

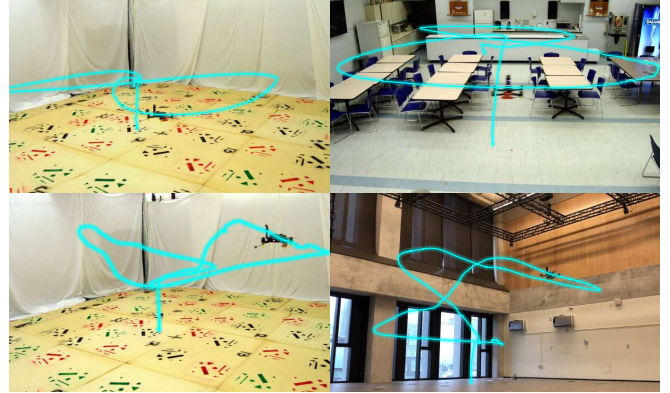
**Abstract**—We propose a fixed-lag smoother-based sensor fusion architecture to leverage the complementary benefits of range-based sensors and visual-inertial odometry (VIO) for localization. We use two fixed-lag smoothers (FLS) to decouple accurate state estimation and high-rate pose generation for closed-loop control. The first FLS combines ultrawideband (UWB)-based range measurements and VIO to estimate the robot trajectory and any systematic biases that affect the range measurements in cluttered environments. The second FLS estimates smooth corrections to VIO to generate pose estimates at a high rate for online control. The proposed method is lightweight and can run on a computationally constrained micro-aerial vehicle (MAV). We validate our approach through closed-loop flight tests involving dynamic trajectories in multiple real-world cluttered indoor environments. Our method achieves decimeter-to-sub-decimeter-level positioning accuracy using off-the-shelf sensors and decimeter-level tracking accuracy with minimally-tuned open-source controllers.

## I. INTRODUCTION

Range-based positioning involves determining the position of a robot by measuring the distance to specific landmarks known as anchors. Range-based positioning is appealing as it is lightweight, drift-free, and computationally inexpensive. Additionally, unlike camera or lidar-based positioning, it does not require a detailed map of the environment or persistent features. Common approaches to range-based positioning include radio-frequency (RF)-based positioning technologies such as the Global Positioning System (GPS) for outdoor environments [1]

However, urban canyons and most indoor environments are not amenable to GPS-based positioning. Ultrawideband (UWB) [2] is an alternative RF-based technology suitable for such environments. A challenge of RF technologies such as UWB is that the measurements are affected by objects in the environment that obstruct the line of sight between the anchor and the robot, resulting in biased *non-line of sight* (NLOS) measurements [3]. The effect of NLOS measurements is more pronounced in cluttered environments. The combination of biases and the low-dimensional nature of range measurements can result in non-smooth state estimates that is undesirable for closed-loop control [4].

Since range measurements are low dimensional, a single range sensor cannot determine the full pose of a robot. As such, range sensors are typically combined with other sensing modalities such as wheel odometry, inertial odometry,



**Fig. 1:** Overlay of trajectories from closed-loop flight experiments using our proposed method for localization. The proposed method uses a dual-rate fixed-lag smoother architecture to combine range measurements from ultrawideband radios and visual inertial odometry for localization. The UTIAS testbed (left top and left bottom) and the Myhal testbed (right bottom) present challenging scenarios in terms of poor geometry of anchors for range-based positioning, and sparse features for visual inertial odometry, respectively. The UTIAS cafeteria (right top) is challenging as ultrawideband signals are affected by reflections from the obstacles. A video of our experiments is available at: [http://tiny.cc/uwb\\_vio](http://tiny.cc/uwb_vio).

and visual odometry [5]–[7]. Decreasing cost of high-quality cameras and inertial measurement units (IMUs) has led to the development of mature visual-inertial odometry (VIO) [8] algorithms. VIO provides smooth relative 3D pose measurements at a high rate. We refer to output obtained from a VIO algorithm as simply odometry. The smooth nature of VIO makes it suitable for closed-loop control. However, VIO is susceptible to drift, especially in dynamic environments with changing scenery.

The above discussion highlights the complementary nature of range-based positioning and odometry-based positioning. We refer to the combination of range and odometry-based localization as *range-aided* localization. To leverage their complementary benefits for accurate state estimation and closed-loop control, any sensor fusion scheme must (i) be tractable for online estimation on computationally constrained platforms, (ii) estimate any biases associated with the range measurements, and (iii) provide high-rate, drift-free, smooth state estimates for reliable trajectory tracking.

A common approach to state estimation is to use a *batch* trajectory estimator [9, Chapter 4]. While such an approach can provide smooth and accurate state estimates, it is not well-suited for online control. Filtering-based methods [5], [10], [11], while computationally lightweight, can result in poorer accuracy and non-smooth estimates as they discard older measurements. These non-smooth estimates can result in oscillatory behavior when used for online control [12]. In this paper, we propose a sensor fusion architecture that meets

The authors are with the Learning Systems and Robotics Lab ([www.learnsyslab.org](http://www.learnsyslab.org)) at the Technical University of Munich, Germany, and the University of Toronto Institute for Aerospace Studies, Canada. They are also affiliated with the University of Toronto Robotics Institute, the Munich Institute of Robotics and Machine Intelligence (MIRMI), and the Vector Institute for Artificial Intelligence. E-mail: {abhishek.goudar, wenda.zhao}@robotics.utias.utoronto.ca, angela.schoellig@tum.de

all of the above requirements. The following are the main contributions of our work.

- We propose a dual fixed-lag smoother (FLS) architecture to fuse range measurements from UWB radios and VIO for localization and navigation. The decoupled nature of our architecture enables high-accuracy trajectory estimation and high-rate smooth pose generation for online control simultaneously.
- We present a computationally lightweight method to estimate the systematic biases in range measurements. Unlike previous works, our approach does not require ground truth pose information, additional sensors, or training data. The proposed bias estimation method reduces the trajectory estimation error by 50% compared to not estimating the biases.
- We demonstrate the effectiveness of our approach in closed-loop flights where the proposed method achieves decimeter-to-sub-decimeter-level localization accuracy and decimeter-level tracking accuracy in real-world cluttered indoor environments.
- We release our source code<sup>1</sup> and the dataset<sup>2</sup> collected during the experiments.

## II. RELATED WORK

In this section, we review previous works that combine range sensors with other sensors to achieve accurate localization. We also review works that estimate biases in UWB measurements. Finally, we review works that focus on the fusion of range and odometry measurements for navigation.

**UWB-aided localization and navigation:** As alluded to earlier, a single UWB radio cannot estimate the full 3D pose, and as such it is used with other sensors such as an IMU or camera. The fusion of UWB and IMU measurements can be done either by using range measurements, referred to as *range-based approach*, or by first estimating position from range measurements and then using the position as input, referred to as *position-based approach*. Systems adopting range-based approaches [5], [13], position-based approaches [14], and a combination thereof [15] have been shown previously. The common sensor fusion strategies include parametric filtering-based methods such as the extended Kalman filter (EKF) [5], [10], [11], [13], [16] and non-parametric methods such as particle filters [3]. More recently, optimization-based approaches have gained traction [7], [17]. Computer vision algorithms have also been used in conjunction with UWB sensors to achieve accurate localization using filtering-based [4], [11], [18] and optimization-based approaches [7]. More recently, tightly-coupled approaches that combine range measurements with data from cameras, IMUs, and lidars [19], [20] for multi-modal simultaneous localization and mapping (SLAM) have been demonstrated. In addition to localization, our proposed system is also designed for reliable closed-loop trajectory tracking on computationally constrained platforms.

**Bias estimation in UWB-based localization:** As mentioned previously, in cluttered environments, UWB measurements are

susceptible to noise and bias. The identification of biases in UWB measurements requires additional information from other sources. Measurements from a motion capture system are used to estimate biases in [14], [21], [22]. Fiducial markers-based pose estimation is used to facilitate bias estimation in [4]. In [19] additional sensors such as lidars are used. The methods proposed in [12], [23] use training data collected from multiple closed-loop flights to learn the bias models. A variety of models including parametric models [3], [12], [14] and non-parametric models such as Gaussian processes [21], neural networks [22], and kernel density-based models [24] have been used to estimate the biases in range measurements. Some of the limitations of previous works are that they require accurate ground truth information or training data and can be computationally intractable for online estimation. In contrast, our method does not require ground truth or training data and is tractable to run online on computationally constrained platforms.

**Multisensor fusion:** We review works that focus on the fusion range measurements obtained from UWB or GPS-based systems and odometry obtained from VIO or IMU for closed-loop control. A common approach is to combine absolute and relative measurements *directly* using a filtering-based method [11], [25] or an optimization-based method [7]. An alternative approach is to model the sensor fusion problem as a frame alignment problem [26]. The benefit of such an approach is that the core estimation pipeline runs at a lower rate to correct the drift in the high-rate relative measurements, which is then used to perform control. We extend the approach of [26] to use range measurements (instead of position measurements) with online bias estimation. Additionally, we augment the frame alignment method of [26] to generate *smooth* frame alignment.

To the best of the authors' knowledge, sensor fusion of UWB-based range measurements and odometry from VIO for (i) estimation of the robot state, (ii) generation of smooth state estimates at high-rate for closed-loop control, and (iii) online estimation of systematic biases in range measurements without additional sensors on computationally constrained platforms has not been demonstrated previously.

## III. PROBLEM STATEMENT

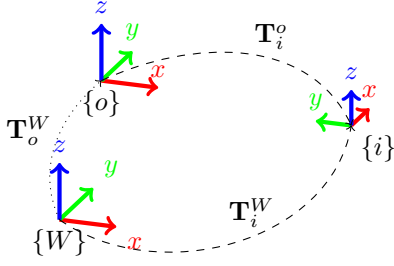
The objective of our work is to not only estimate the robot trajectory and the biases associated with range measurements, but also to generate smooth state estimates at high-rate for closed-loop control. We assume that (i) the position of the anchors is known, and (ii) the robot is equipped with a UWB radio and a high-rate VIO sensor.

We introduce the notation that will be used throughout the paper. Elements of the special Euclidean *Lie* group  $\mathbf{T} \in SE(3)$  are used to represent 3D poses. A generic pose  $\mathbf{T}$  is parameterized as  $\mathbf{T} = \{\mathbf{p}, \mathbf{R}\}$ , where  $\mathbf{p} \in \mathbb{R}^3$  represents the 3D position and  $\mathbf{R} \in SO(3)$ , a member of the special orthogonal group, represents the orientation. The pose of a frame  $A$  in frame  $B$  is represented by  $\mathbf{T}_B^A = \{\mathbf{p}_B^A, \mathbf{R}_B^A\}$ .

We denote the world frame by  $\{W\}$ , the robot frame by  $\{i\}$ , and the local frame associated with VIO by  $\{o\}$ . Unlike the world frame, the local frame is not fixed and is set to

<sup>1</sup>[https://github.com/utiasDSL/ra\\_lan.git](https://github.com/utiasDSL/ra_lan.git)

<sup>2</sup>[https://utiasdsl.github.io/utias\\_ra\\_loc](https://utiasdsl.github.io/utias_ra_loc)



**Fig. 2:** The problem of sensor fusion of UWB range measurements and VIO odometry is modeled as a frame alignment problem between the world frame  $\{W\}$  and the odometry frame  $\{o\}$ . The *absolute* but non-smooth estimate  $\mathbf{T}_i^W$  obtained from fusing UWB and VIO measurements is combined with the *relative* but smooth estimate  $\mathbf{T}_i^o$  from VIO to estimate the offset  $\mathbf{T}_o^W$  between the two frames.

the point in space where the VIO algorithm is initialized. The UWB range measurements are reported in the world frame whereas odometry from VIO is in the local frame. The bias associated with the measurements from the  $l^{\text{th}}$  anchor,  $a_l$ , is denoted by  $b_{a_l} \in \mathbb{R}$ .

#### IV. MODELLING

In the next subsection, we provided an overview of our system architecture. A detailed description of the individual components is provided in the subsequent subsection.

##### A. Motivation and System description

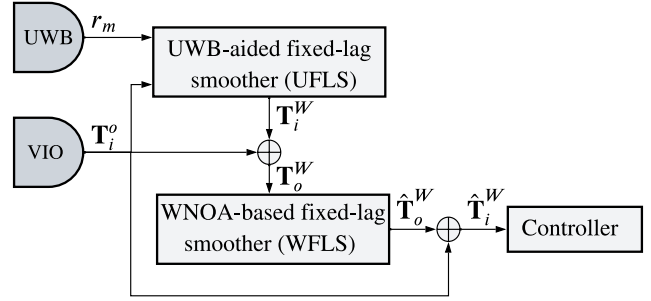
Similar to [26], we formulate the problem of sensor fusion as a frame alignment problem. Unlike [26], which considers a position-based approach, we use a range-based approach. A limitation of the position-based approach is that the effects of the biases in range measurements are not discernible from the position measurements. In a range-based approach, biases result in longer range measurements, and hence, the biases can be estimated.

The block diagram of our system architecture is shown in Figure 3. First, we combine range measurements and odometry to estimate the robot trajectory and any systematic biases in range measurements using a fixed-lag smoother (UFLS). The estimated robot pose,  $\mathbf{T}_i^W$ , is non-smooth due to the sparse and non-smooth nature of range measurements. To keep the computational cost low, UFLS is run at a lower rate compared to individual sensor update rates. However, the low-rate and non-smooth nature of the output of UFLS makes it unsuitable for online control.

The output of VIO,  $\mathbf{T}_i^o$ , is typically high-rate and smooth but drifts over time. An approach to get high-rate drift-free estimates is to combine the output of UFLS,  $\mathbf{T}_i^W$ , and the output of VIO,  $\mathbf{T}_i^o$ , by performing frame alignment. Specifically, the output of UFLS and VIO can be used to obtain the frame offset between the world frame and the local frame

$$\mathbf{T}_o^W = \mathbf{T}_i^W (\mathbf{T}_i^o)^{-1}. \quad (1)$$

The frame diagram for such a setup is shown in Figure 2. We have omitted the dependency on time for notational clarity. This frame offset can be combined with the latest odometry-based pose estimate [26] to generate high-rate and drift-free



**Fig. 3:** Overview of the proposed architecture. Range measurements from UWB radii and odometry from VIO are combined in the first fixed-lag smoother (UFLS) to estimate the robot pose  $\mathbf{T}_i^W$  and the systematic biases in range measurements. The estimated robot pose is then combined with the latest estimate from VIO  $\mathbf{T}_i^o$  to estimate the frame offset  $\mathbf{T}_o^W$  between the world frame and the local frame. A white-noise-on-acceleration (WNOA) motion model is used to generate a smooth frame offset,  $\hat{\mathbf{T}}_o^W$ , using the second fixed-lag smoother (WFLS). The smoothed frame offset is then combined with VIO-based odometry to obtain smooth, high-rate, and drift-free robot pose  $\hat{\mathbf{T}}_i^W$  which is sent to the controller.

pose estimates,  $\hat{\mathbf{T}}_i^W = \hat{\mathbf{T}}_o^W \mathbf{T}_i^o$ . While this addresses the issue of generating drift-free high-rate pose estimates, in a range-based setup, the frame offset  $\mathbf{T}_o^W$  (and hence  $\hat{\mathbf{T}}_i^W$ ) is non-smooth on account of  $\mathbf{T}_i^W$  being non-smooth.

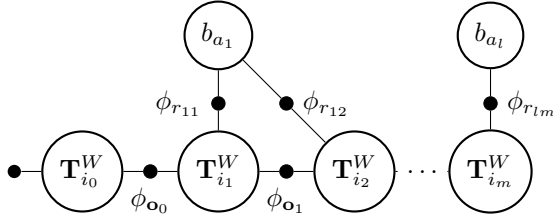
To generate smooth frame offset estimates, we impose a white-noise-on-acceleration (WNOA) [27], [28] motion model on the frame offset. The motivation for choosing such a model is that (i) the WNOA model provides a *smooth prior* which acts as a regularization term to filter any non-smooth component, and (ii) the WNOA model leads to motion priors with sparse system matrices which be solved efficiently. The motion prior and the frame offset estimates obtained from (1) are combined in a WNOA-based fixed-lag smoother (WFLS in Figure 3) to generate smooth frame offset  $\hat{\mathbf{T}}_o^W$ . This smoothed frame offset is then combined with the latest odometry-derived pose estimate to generate drift-free, high-rate, and smooth robot pose estimate,  $\hat{\mathbf{T}}_i^W = \hat{\mathbf{T}}_o^W \mathbf{T}_i^o$ .

The advantage of such an architecture is that the estimation of the biases and the robot trajectory can happen at a lower rate while still being able to generate smooth state estimates using drift-corrected odometry for online control. Note that the odometry from VIO is used twice in entire pipeline. However, the odometry estimate used for calculating the frame offset is different (older) compared to the odometry estimate combined with the smoothed frame offset for control, which precludes double counting of odometry.

##### B. UWB-aided fixed-lag smoother

In UFLS, a fixed window of range and odometry measurements are combined to estimate the robot trajectory and the biases in range measurements. The size of the window is parameterized by time duration  $\delta t_{\text{ufls}}$ . Measurements older than  $\delta t_{\text{ufls}}$  are discarded and states older than  $\delta t_{\text{ufls}}$  are marginalized. The factor graph corresponding to such a setup is shown in Figure 4. The UWB range measurement at any time  $t$  between the robot and an anchor  $a_l$  is given by

$$r_{lt} = \|\mathbf{p}_{a_l}^W - \mathbf{R}_{it}^W \mathbf{p}_u^i - \mathbf{p}_{it}^W\|_2 + b_{a_l} + \eta_{rt}, \quad (2)$$



**Fig. 4:** Factor graph for the UWB-aided fixed-lag smoother (UFLS). Preintegrated odometry is added as a binary factor  $\phi_{o_t}$  to constrain consecutive robot poses. Each range measurement adds a binary factor  $\phi_{r_{it}}$  between the robot pose  $\mathbf{T}_{i_t}^W$  and the node  $b_{a_l}$  representing the bias associated with anchor  $a_l$ .

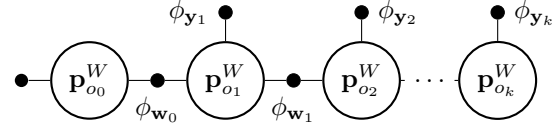
where  $\|\cdot\|_2$  is the  $\ell^2$  norm,  $\mathbf{T}_{it}^W = \{\mathbf{p}_{it}^W, \mathbf{R}_{it}^W\}$  is the pose of the robot at time  $t$ ,  $\mathbf{p}_{i_u}^u$  is the position of the UWB radio w.r.t the body frame a.k.a *lever arm*,  $\eta_{rt} \sim \mathcal{N}(0, \sigma_r^2)$  is the additive white Gaussian noise,  $\mathbf{p}_{a_l}^W$  is the anchor position, and  $b_{a_l}$  is the bias associated with measurements from anchor  $a_l$ , respectively. In contrast to previous works, we model the biases as part of the same factor graph as the robot trajectory. The motivation for such a model is to capture the correlations between the robot trajectory and the measurement biases. Although biases are assumed to be constant in a particular window of the smoother, they can change between subsequent windows to accommodate spatially varying biases. A binary factor  $\phi_{r_{it}}$  between the latent robot state and the latent anchor-specific bias is added for every range measurement. To filter outliers, we employ a simple Euclidean distance check where range measurements that exceed the predicted range by  $\delta r_{thr}$  are discarded.

As mentioned previously, the update rate of VIO is generally higher than that of UWB. Consequently, multiple odometry estimates are received between two subsequent range measurements. The conventional approach of adding a node in the factor graph for every odometry estimate can result in a large graph size and increase the computational cost quickly. Alternatively, only those odometry estimates that are *time-aligned* with the range measurements can be included in the graph [26], however, in doing so, useful odometry information is discarded. In order to include all of the odometry information while keeping the computational cost low, we take inspiration from IMU preintegration [29] and propose a similar method to integrate multiple odometry estimates into a single preintegrated measurement.

*Odometry preintegration:* We represent uncertainty in 3D poses using the formulation of [30] but with the *right perturbation* convention. Specifically, a generic pose is decomposed into a nominal pose  $\bar{\mathbf{T}}_{i_t} \in SE(3)$  and a perturbation  $\xi_t \in \mathbb{R}^{6 \times 1}$ :

$$\mathbf{T}_{i_t} = \bar{\mathbf{T}}_{i_t} \exp(\xi_t^\wedge), \quad (3)$$

where  $\exp$  is a *retraction* operation for  $SE(3)$ ,  $\wedge$  maps an element of  $\mathbb{R}^6$  to an element of the associated *Lie algebra*  $se(3)$ , and  $\xi_t \sim \mathcal{N}(\mathbf{0}, \Sigma_{i_t})$  is a Gaussian random variable. Consider a sequence of monotonic time steps  $\{t_0, t_1, \dots, t_k\}$ , and the corresponding odometry estimates obtained at those time steps:  $\{\{\bar{\mathbf{T}}_{i_1}^o, \Sigma_{i_1}^o\}, \{\bar{\mathbf{T}}_{i_2}^o, \Sigma_{i_2}^o\}, \dots, \{\bar{\mathbf{T}}_{i_k}^o, \Sigma_{i_k}^o\}\}$ . Here,  $\bar{\mathbf{T}}_{i_t}^o \in SE(3)$  represents the pose of the robot frame  $\{i\}$  in the odometry frame  $\{o\}$  at time  $t$  and  $\Sigma_{i_t}$



**Fig. 5:** Factor graph for the white-noise-on-acceleration (WNOA) fixed-lag smoother (WFLS). Motion priors obtained using the WNOA motion model are added as binary factors  $\phi_{w_t}$  to constrain the position component of the frame offset,  $\mathbf{p}_{o_t}^W$ . Frame offsets computed by UFLS are added as unary measurement factors  $\phi_{y_t}$ .

is a positive definite matrix representing the corresponding covariance. First, we convert the odometry estimates into *incremental* odometry between consecutive time steps:  $\{\{\Delta \bar{\mathbf{T}}_{i_1}^o, \Sigma_{i_1}^o\}, \{\Delta \bar{\mathbf{T}}_{i_2}^o, \Sigma_{i_2}^o\}, \dots, \{\Delta \bar{\mathbf{T}}_{i_k}^o, \Sigma_{i_k}^o\}\}$ , where

$$\Delta \bar{\mathbf{T}}_{i_{t+1}}^o = (\bar{\mathbf{T}}_{i_t}^o)^{-1} \bar{\mathbf{T}}_{i_{t+1}}^o, \quad (4)$$

$$\Sigma_{i_{t+1}}^o = \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_{t+1}}^o)^{-1}} \Sigma_{i_t}^o \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_{t+1}}^o)^{-1}}^T + \Sigma_{i_{t+1}}^o, \quad (5)$$

and,  $\mathbf{Ad}_{\mathbf{T}}$  is the *Adjoint* operator for  $SE(3)$ . The incremental odometry estimates are then composed to give a single preintegrated measurement:

$$\Delta \bar{\mathbf{T}}_{i_{t+2}}^o = \Delta \bar{\mathbf{T}}_{i_{t+1}}^o \Delta \bar{\mathbf{T}}_{i_{t+2}}^o, \quad (6)$$

$$\Sigma_{i_{t+2}}^o = \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_{t+2}}^o)^{-1}} \Sigma_{i_{t+1}}^o \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_{t+2}}^o)^{-1}}^T + \Sigma_{i_{t+2}}^o. \quad (7)$$

The derivation of the above equations can be found in Appendix A. Equations (14) and (15) provide the mean and the covariance of the preintegrated odometry measurement which captures the total *relative* displacement and orientation change as given the individual odometry estimates. Each preintegrated odometry measurement  $\phi_{o_t}$  is added as a binary factor between consecutive nodes in the factor graph as shown in Figure 4. We estimate the latent states using *maximum a posteriori* (MAP) inference.

### C. WNOA-based fixed-lag smoother

The output of UFLS is combined with a time-aligned odometry estimate to estimate the frame offset using (1). The WNOA-based fixed-lag smoother (WFLS) uses a WNOA motion model and noisy estimates of the frame offset,  $\mathbf{T}_{o_t}^W$ , to generate smooth estimates of the frame offset,  $\hat{\mathbf{T}}_{o_t}^W$ . The factor graph for this setup is shown in Figure 3. Generally, range measurements have a more direct impact on the position component  $\mathbf{p}_{o_t}^W$  of the frame offset compared to the rotation component  $\mathbf{R}_{o_t}^W$ . This is because a range measurement influences the rotation component only through the lever arm,  $\mathbf{p}_{i_u}^u$ , which is generally small compared to the magnitude of range measurement. Hence, we impart a motion model only on the position component.

*Motion model:* We adopt the Gaussian process (GP)-based continuous-time formulation proposed in [27] and impart the following linear time-invariant motion model on the position component  $\mathbf{p}_{o_t}^W$ :

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{L}\eta_w(t),$$

with

$$\mathbf{x}(t) = \begin{bmatrix} \mathbf{p}_{o_t}^W \\ \dot{\mathbf{p}}_{o_t}^W \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix}, \mathbf{L} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix},$$



where  $\mathbf{I}$  and  $\mathbf{0}$  are the identity and zero matrices of appropriate dimensions,  $\dot{\mathbf{p}}_{o_t}^W = d\mathbf{p}_{o_t}^W/dt$ , and  $\boldsymbol{\eta}_w(t) \sim \mathcal{GP}(\mathbf{0}, \mathbf{Q}_w\delta(t-t'))$  is white noise drawn from a zero-mean GP with power spectral density matrix  $\mathbf{Q}_w$ .

Consider a sequence of monotonic times  $(t_0, t_1, \dots, t_k)$ . The mean state at time  $t_j$  is

$$\mathbf{x}_{t_j} = \Phi(t_j, t_{j-1})\mathbf{x}_{j-1}, \quad (8)$$

with the state transition matrix

$$\Phi(t_j, t_{j-1}) = \begin{bmatrix} \mathbf{I} & \Delta t_{j:j-1}\mathbf{I} \\ \mathbf{0} & \mathbf{I} \end{bmatrix},$$

where  $\Delta t_{j:j-1} = t_j - t_{j-1}$ . The corresponding covariance matrix is

$$\mathbf{Q}_{j:j-1} = \begin{bmatrix} \frac{1}{3}\Delta t_{j:j-1}^3 \mathbf{Q}_w & \frac{1}{2}\Delta t_{j:j-1}^2 \mathbf{Q}_w \\ \frac{1}{2}\Delta t_{j:j-1}^2 \mathbf{Q}_w & \Delta t_{j:j-1} \mathbf{Q}_w \end{bmatrix}. \quad (9)$$

The motion prior given by (8) and (15) is added as a binary factor  $\phi_{w_{j-1}}$  in the factor graph (see Figure 5). The measurement in this case is the position component of the frame offset at time  $t_j$  computed using (1)

$$\mathbf{y}_j = \mathbf{p}_{o_j}^W + \boldsymbol{\eta}_{oj}, \quad (10)$$

where  $\boldsymbol{\eta}_{oj} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\mathbf{p}_{o_j}^W})$  is the additive white Gaussian noise. The covariance  $\Sigma_{\mathbf{p}_{o_j}^W}$  is obtained from UFLS as the marginal covariance. We add a unary factor  $\phi_{\mathbf{y}_j}$  to the factor graph upon the receipt of a new frame offset. Similar to UFLS, latent states in WFLS are inferred using MAP inference. As before, the window size is parameterized by a time duration  $\delta t_{\text{wfls}}$ . The window size and the update rate of WFLS are independent of UFLS update rate and can be run at a lower or higher rate depending on the application. The output of WFLS is a smooth estimate of the position component of the frame offset  $\hat{\mathbf{p}}_{o_j}^W$ . The rotation component of the frame offset  $\hat{\mathbf{R}}_{o_j}^W$  is obtained by plugging in the most recent estimate from UFLS and a time-aligned VIO measurement in (1). Although we have imposed a motion model only on the position component, the extension to include the rotation component is straightforward.

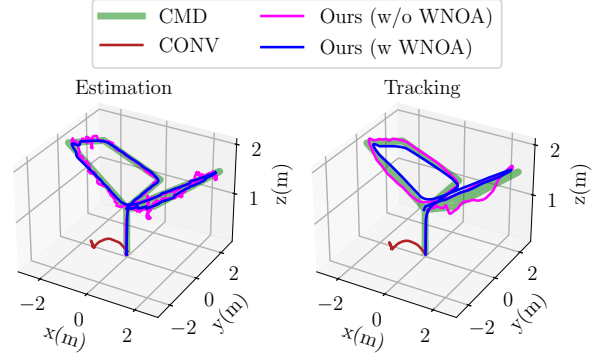
## V. EXPERIMENTAL RESULTS

In this section, we demonstrate the effectiveness of our method through simulation and real experiments. In simulation, we present qualitative results which highlight the importance of the WNOA motion model on the frame offset. We defer results quantifying the efficacy and accuracy of our method to real world experiments.

### A. Simulation

We use the Gazebo simulator and a quadrotor from the Rotors simulator package as our test platform. The quadrotor is equipped with a generic range sensor and an odometry sensor. The noise parameters for the simulated sensors are chosen to reflect real sensor behavior.

We perform a qualitative comparison of the closed-loop performance of our method against two different baselines in simulation as it provides a safe environment to test different methodologies. The first baseline is the conventional (CONV) method where the output of the fixed-lag smoother UFLS is



**Fig. 6:** Trajectory estimation (left) and tracking (right) performance of different approaches. The inclusion of a white-noise-on-acceleration (WNOA) motion model (Ours w WNOA) provides smooth corrections to VIO which results in reliable tracking of the commanded (CMD) trajectory compared to not including a WNOA motion model (Ours w/o WNOA) or the conventional approach (CONV) where output of UFLS is used for control.

used for closed-loop control. The second baseline uses the output of WFLS but without any smoothing, dubbed Ours (w/o WNOA). This is similar to the method of [26]. We refer to our proposed method as Ours (w WNOA). The same minimum-jerk trajectory is commanded in all experiments. The stock proportional-integral-derivative (PID) controllers from the Rotors simulator package are used for both position and attitude control.

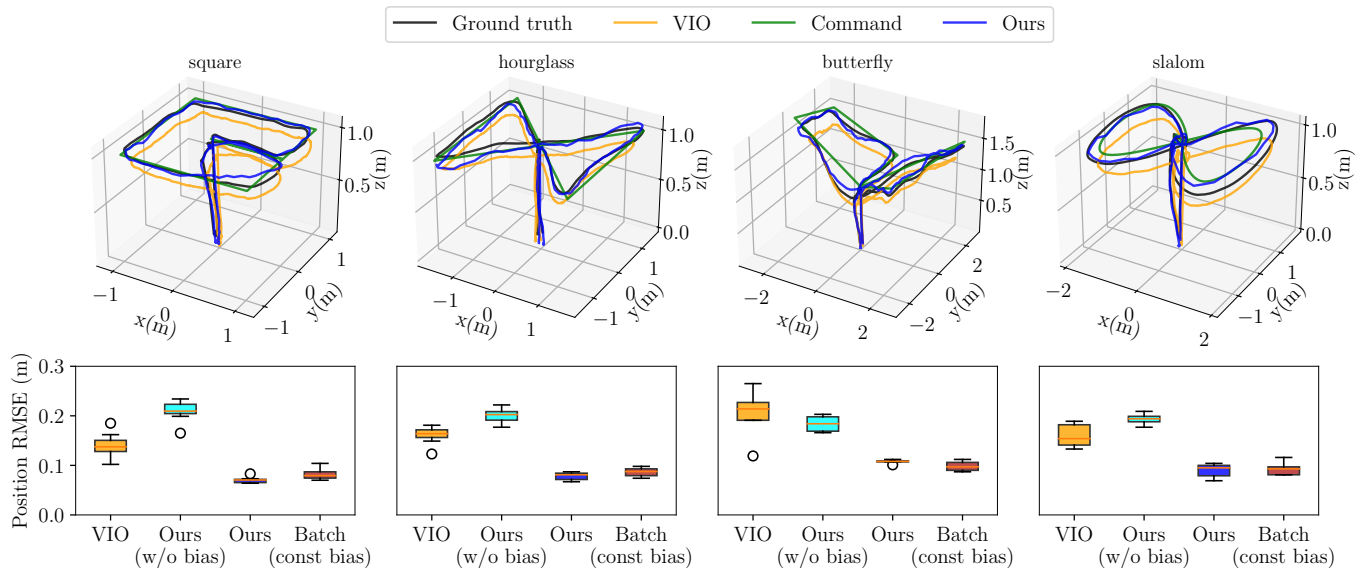
The results are shown in Figure 6. The CONV baseline results in unstable tracking due to the low rate and non-smooth output of UFLS. The second baseline tracks the commanded trajectory, however, the non-smooth nature of the estimated frame offset results in poor tracking performance. In contrast, our proposed method tracks the commanded trajectory reliably. We provide additional details on the benefit of the additional smoothing in Appendix B.

### B. Real experiments

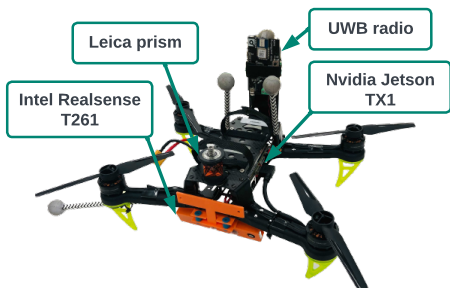
To quantify the performance of the proposed method, we performed closed-loop flight experiments in three real indoor environments: (i) University of Toronto, Institute for Aerospace Studies (UTIAS) testbed, (ii) UTIAS cafeteria, and (iii) Myhal testbed. Due to space constraints, we provide quantitative results from the UTIAS testbed and the UTIAS cafeteria. Footage of the flight experiments from all three environments can be found in the accompanying video<sup>1</sup>.

*Baselines:* We focus on trajectory estimation accuracy, which is the core contribution of our work. We compare our method against three different baselines. The first baseline consists of VIO-only estimation, which we refer to as *VIO*. Since VIO provides relative pose estimates, we initialize the VIO algorithm close to the ground truth only for comparison. The second baseline is similar to the method proposed in [26] where no bias estimation is done, which we refer to as *Ours (w/o bias)*. Note that we had to modify the method of [26] to incorporate range measurements as the UWB system used in our experiments does not provide position measurements. The third baseline is batch trajectory estimation [9, Chapter 4].

<sup>1</sup>[http://tiny.cc/uwb\\_vio](http://tiny.cc/uwb_vio)



**Fig. 7:** Results from closed-loop flight experiments at the UTIAS testbed. The top row shows the qualitative trajectory estimation performance. The ground truth trajectory also shows the closed-loop tracking performance compared to the commanded trajectory (Command). The bottom row shows the position root-mean-square error (RMSE) box plots for (i) visual-inertial odometry (VIO), (ii) our method without anchor bias compensation (Ours w/o bias), (iii) batch estimation with constant anchor bias compensation (Batch const bias), and (iv) our method including anchor bias compensation (Ours). The maximum speed for the slalom path is 2 m/s and for the others is 1 m/s.



**Fig. 8:** Our test platform used in experiments is a quadrotor equipped with a DW1000-based UWB radio and an Intel Realsense T261. All computation is performed onboard on a Nvidia Jetson TX1 computer. In areas where a motion capture system is not available, a Leica total station is used to get ground truth position by tracking the prism on the quadrotor.

This represents the ideal scenario where all of the sensor data is available for estimation. For batch estimation, we consider a single bias latent variable for each anchor across the entire trajectory, which we refer to as *Batch (const bias)*. We refer to our method as *Ours*.

*Setup:* We use DW1000-based UWB radios from Bitcraze. The position of the UWB anchors is measured using a Leica total station. Our test platform is a custom-built quadrotor (see Figure 8) equipped with a UWB radio which provides range measurements, and an Intel Realsense T261 that provides VIO. We disable the the loop closure feature on the T261 to run it in VIO-only mode. All computation is performed onboard on a Jetson TX1 single-board computer. The lower-level flight controller is run on a PX4-based autopilot with stock firmware. The quadrotor is also equipped with a Leica prism to obtain ground truth position in environments where a motion capture system is not available. The UWB radios are operated in two-way range (TWR) mode. The range measurement outlier rejection threshold is chosen as  $\delta r_{\text{thr}} = 0.5$  m, which is larger



**Fig. 9:** A panoramic view of the UTIAS cafeteria with the positions of the UWB anchors highlighted in red circles. The dimensions of the cafeteria are 10 m  $\times$  10 m  $\times$  5 m.

than  $3\sigma_r$  with  $\sigma_r = 0.1$  m. The two fixed-lag smoothers, UFLS and WFLS, are implemented using the GTSAM library [31]. The frequency of range measurements and odometry from VIO are 17 Hz and 200 Hz, respectively. The update rate of UFLS is 5 Hz and that of WFLS is 10 Hz with  $\delta t_{\text{ufls}} = 1$  s and  $\delta t_{\text{wfls}} = 1$  s.

*Experimental procedure:* The proposed estimation and control pipelines run online on the onboard computer of the quadrotor. For navigation, we generate minimum-jerk trajectories using coarse user-defined waypoints. A trajectory sampler uses the estimated state to generate pose and velocity setpoints to track the reference minimum-jerk trajectory. In a new environment, we perform a single autonomous calibration flight to obtain coarse values for anchor-related biases, which are then used as priors for subsequent flights. In each experiment, estimation results for the proposed method along with the sensor data are recorded onboard. The recorded sensor data is used to evaluate the different baselines *offline*.

*UTIAS testbed:* Our first test space is an indoor testbed with 8 UWB anchors at the corners of a flight arena of dimensions 7 m  $\times$  8 m  $\times$  3.5 m. The arena is equipped with a Vicon motion capture system for ground truth. We performed

multiple closed-loop flight experiments with 4 different paths shown in Figure 7. The commanded speed for the first 3 paths is 1 m/s, whereas the commanded speed for the slalom path is 2 m/s. Box plots showing the absolute position root-mean-square error (RMSE) are provided in the bottom row of Figure 7. The top row shows qualitative estimation and tracking results from one such experiment for the different paths. Note that since we are doing closed-loop flights, the ground truth (GT) trajectory shows the tracking performance.

The box plots highlight the importance of anchor bias estimation as it doubles the estimation accuracy in many cases. Since batch estimation considers the full measurement history, it is expected to be better than the proposed method. However, in some cases, the proposed method is more accurate than batch estimation, which can be attributed to *spatially* varying biases. Specifically, batch estimation considers constant anchor biases for the entire trajectory whereas the proposed method considers the biases to be constant only within the current estimation window. This observation is consistent with the results of [3]. In most cases we achieve a position RMSE below 10 cm. The performance on the butterfly trajectory is poorer as the MAV travels to the extremities of the test arena where the estimation is affected by the dilution of precision [1, Section 7.3]. The position RMSE values observed in the square path are similar to the values observed in [17]. The results are promising considering that the DW1000-based UWB radios used in our experiments have a precision of  $\pm 10$  cm compared to the P440 UWB radios used in [17] which have a precision of  $\pm 1$  cm. The results show that the assumption of constant bias within the current estimation window is sufficient to achieve (sub)decimeter-level accuracy in many cases.

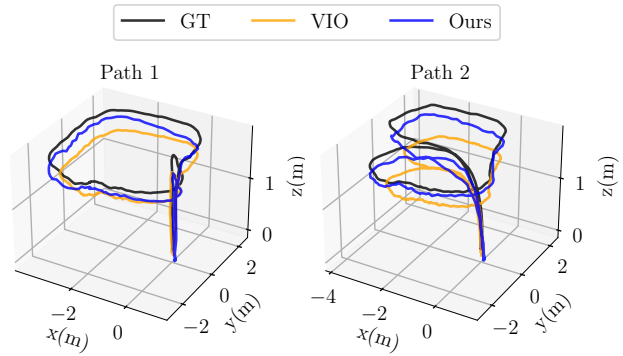
Although the core contribution of our work is on localization, we provide the best-case trajectory tracking performance for the four test trajectories in Table I. The results show that our method achieves smooth and reliable tracking at moderate speeds with minimally tuned off-the-shelf open-source controllers. The higher tracking error in the slalom path can be attributed to the minimally tuned controller as the MAV overshoots the commanded trajectory at higher speeds. Note however that the trajectory estimation accuracy is still high in this case (see boxplot in Figure 7).

*UTIAS cafeteria:* Our second test space is the UTIAS cafeteria. A panoramic view of the cafeteria is shown in Figure 9. The dimensions of the cafeteria are 10 m  $\times$  10 m  $\times$  5 m with 8 UWB anchors installed on the ceiling and on the floor. This is a very challenging setup for UWB-based localization as all of the anchors are occluded by metallic and non-metallic objects that induce serve NLOS scenarios.

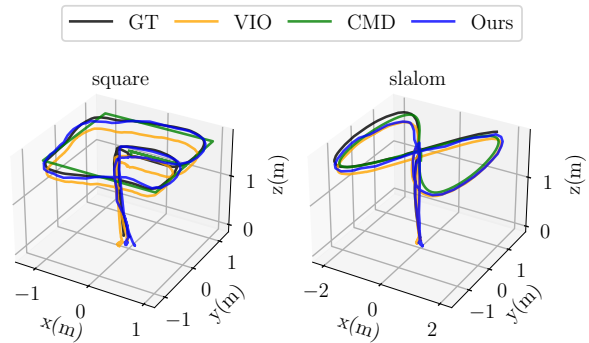
For ground truth data, we use a Leica total station in tracking mode. In this mode, the total station tracks a prism that is mounted on the quadrotor (see Figure 8) and outputs position of the prism at 5 Hz. We observed that the total station would

**TABLE I:** Best-case trajectory tracking performance using the proposed method from flight tests in the UTIAS testbed.

Path	square	hourglass	butterfly	slalom
RMS Tracking Error (m)	0.106	0.118	0.135	0.232



**Fig. 10:** Trajectory estimation results from two experiments in the UTIAS cafeteria. The quadrotor is manually moved along different paths in the experiments. The proposed estimation pipeline is run online onboard the quadrotor and results from our method (Ours), VIO-only estimation (VIO), and ground truth (GT) are recorded. The corresponding position root-mean-square error (RMSE) values are provided in Table II.



**Fig. 11:** Results from closed-loop flight experiments in the UTIAS cafeteria. The estimated trajectories from VIO-only estimation (VIO) and our proposed method (Ours) are shown. The maximum speeds for the square path and the slalom path are 1 m/s and 2 m/s, respectively.

lose track of the quadrotor during closed-loop flights. Hence, to quantify the estimation accuracy we simulated closed-loop flights by moving the quadrotor manually at lower speeds. In each of the experiments, the estimation pipeline runs online onboard the quadrotor. Results from two such experiments are shown in Figure 10. The corresponding position RMSE values from four experiments are provided in Table II. The accuracy of the proposed method is lower compared to the previous case but similar to that observed with the butterfly path in the UTIAS testbed. We see that in some cases the proposed method performs better than the batch approach, which can be attributed to spatially varying biases.

We performed flight experiments with the same four paths

**TABLE II:** Position root-mean-square error (RMSE) for VIO-only estimation, batch trajectory estimation with constant bias compensation (Batch const bias) and our method (Ours) from experiments in UTIAS cafeteria.

Algorithm	Position RMSE (m)			
	Path 1	Path 2	Path 3	Path 4
VIO	0.272	0.434	0.208	0.227
Batch (const bias)	<b>0.125</b>	0.217	<b>0.150</b>	0.137
Ours	0.130	<b>0.154</b>	0.152	<b>0.121</b>

from the previous setup. Results from two such experiments are shown in Figure 11. During the experiments, we observed that most of the anchors were occluded and less than half were in line of sight at any given time. The proposed method is still able to accurately track the desired trajectories at moderate speeds in highly-cluttered environments.

## VI. CONCLUSION AND FUTURE WORK

In this paper, we presented a dual fixed-lag smoother approach to fusion of range measurements and VIO for localization and navigation of MAVs. We showed that the proposed method achieves accurate localization and also generates smooth pose estimates for control. The proposed approach is able to estimate any systematic biases in range measurements resulting in a two-fold increase in localization accuracy. Our method is lightweight and can run on a computationally constrained MAV. Through multiple real world experiments, we showed that the proposed approach achieves decimeter-to-sub-decimeter-level localization accuracy with off-the-shelf sensors and open-source controllers in cluttered indoor environments.

## REFERENCES

- [1] Elliott D. Kaplan. *Understanding GPS : Principles and Applications*. 1996.
- [2] Mohammadreza Yavari and Bradford G Nickerson. Ultra wideband wireless positioning systems. *Technical Report TR14-230*, 506, 2014.
- [3] Amanda Prorok, Lukas Gonon, and Alcherio Martinoli. Online model estimation of ultra-wideband TDOA measurements for mobile robot localization. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 807–814, 2012.
- [4] David Hoeller, Anton Ledergerber, Michael Hamer, and Raffaello D’Andrea. Augmenting Ultra-Wideband Localization with Computer Vision for Accurate Flight. *IFAC-PapersOnLine*, 50(1):12734–12740, 2017.
- [5] Jeroen D. Hol, Fred Dijkstra, Henk Luinge, and Thomas B. Schon. Tightly coupled UWB/IMU pose estimation. In *International Conference on Ultra-Wideband*, pages 688–692. IEEE, 2009.
- [6] Jose-Luis Blanco, Javier Gonzalez, and Juan-Antonio Fernandez-Madrugal. A pure probabilistic approach to range-only SLAM. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1436–1441, 2008.
- [7] Thien Hoang Nguyen, Thien-Minh Nguyen, and Lihua Xie. Range-focused Fusion of Camera-IMU-UWB for Accurate and Drift-reduced Localization. *IEEE Robotics and Automation Letters*, 6(2):1–1, 2021.
- [8] Davide Scaramuzza and Zichao Zhang. *Aerial Robots, Visual-Inertial Odometry of*, pages 1–9. Springer Berlin Heidelberg, 2020.
- [9] Timothy D. Barfoot. *State estimation for robotics*. Cambridge University Press, Second edition, 2023.
- [10] Mark W. Mueller, Michael Hamer, and Raffaello D’Andrea. Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadcopter state estimation. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 2015-June, pages 1730–1736, 2015.
- [11] Karol Hausman, Stephan Weiss, Roland Brockers, Larry Matthies, and Gaurav S. Sukhatme. Self-calibrating multi-sensor fusion with probabilistic measurement validation for seamless sensor switching on a UAV. In *Proc. of the IEEE International Conference on Robotics and Automation*, pages 4289–4296, 2016.
- [12] Bas van der Heijden, Anton Ledergerber, Rajan Gill, and Raffaello D’Andrea. Iterative Bias Estimation for an Ultra-Wideband Localization System. *IFAC-PapersOnLine*, 53(2):1391–1396, 2020. 21st IFAC World Congress.
- [13] Abhishek Goudar and Angela P Schoellig. Online Spatio-temporal Calibration of Tightly-coupled Ultrawideband-aided Inertial Localization. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 1161–1168. IEEE, 2021.
- [14] Kexin Guo, Zhirong Qiu, Cunxiao Miao, Abdul Zaini, Chun-Lin Chen, Wei Meng, and Lihua Xie. Ultra-Wideband-Based Localization for Quadcopter Navigation. *Unmanned Systems*, 04:1–12, 02 2016.
- [15] Jungwon Kang, Kunwoo Park, Zahra Arjmandi, Gunho Sohn, Mozhdeh Shahbazi, and Patrick Ménard. Ultra-Wideband Aided UAV Positioning Using Incremental Smoothing with Ranges and Multilateration. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 4529–4536, 2020.
- [16] Jiaxin Li, Yingcai Bi, Kun Li, Kangli Wang, Feng Lin, and Ben M. Chen. Accurate 3D Localization for MAV Swarms by UWB and IMU Fusion. In *Proc. of the IEEE International Conference on Control and Automation, ICCA*, volume 2018-June, pages 100–105. IEEE, 2018.
- [17] Xu Fang, Chen Wang, Thien-Minh Nguyen, and Lihua Xie. Graph Optimization Approach to Range-Based Localization. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 51(11):6830–6841, 2021.
- [18] Hanna E. Nyqvist, Martin A. Skoglund, Gustaf Hendeby, and Fredrik Gustafsson. Pose estimation using monocular vision and inertial sensors aided with ultra wide band. *International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, (October):1–10, 2015.
- [19] Thien-Minh Nguyen, Shenghai Yuan, Muqing Cao, Thien Hoang Nguyen, and Lihua Xie. VIRAL SLAM: Tightly Coupled Camera-IMU-UWB-Lidar SLAM. *arXiv:2105.03296*, 2021.
- [20] Yanjun Cao and G. Beltrame. VIR-SLAM: visual, inertial, and ranging SLAM for single and multi-robot systems. *Autonomous Robots*, 45:905 – 917, 2020.
- [21] Anton Ledergerber and Raffaello D’Andrea. Ultra-wideband range measurement model with Gaussian processes. *1st Annual IEEE Conference on Control Technology and Applications, CCTA 2017*, 2017-Janua:1929–1934, 2017.
- [22] Wenda Zhao, Jacopo Panerati, and Angela P. Schoellig. Learning-Based Bias Correction for Time Difference of Arrival Ultra-Wideband Localization of Resource-Constrained Mobile Robots. *IEEE Robotics and Automation Letters*, 6(2):3639–3646, 2021.
- [23] Anton Ledergerber and Raffaello D’Andrea. Calibrating Away Inaccuracies in Ultra Wideband Range Measurements: A Maximum Likelihood Approach. *IEEE Access*, 6:78719–78730, 2018.
- [24] Acshi Haggemiller, Maximilian Krogus, and Edwin Olson. Non-parametric error modeling for ultra-wideband localization networks. In *Proc. of the IEEE International Conference on Robotics and Automation*, volume 2019-May, pages 2568–2574, 2019.
- [25] Simon Lynen, Markus W. Achtelik, Stephan Weiss, Margarita Chli, and Roland Siegwart. A robust and modular multi-sensor fusion approach applied to MAV navigation. In *Proc. of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, pages 3923–3929, 2013.
- [26] Ruben Mascaro, Lucas Teixeira, Timo Hinzmann, Roland Siegwart, and Margarita Chli. GOMSF: Graph-Optimization Based Multi-Sensor Fusion for robust UAV Pose estimation. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1421–1428, 2018.
- [27] Timothy D. Barfoot, Chi Hay Tong, and Simo Särkkä. Batch Continuous-Time Trajectory Estimation as Exactly Sparse Gaussian Process Regression. *Robotics: Science and Systems*, 2014.
- [28] Sean Anderson and Timothy D. Barfoot. Full STEAM ahead: Exactly sparse Gaussian process regression for batch continuous-time trajectory estimation on SE(3). In *Proc. of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, volume 2015-December, pages 157–164, 2015.
- [29] Christian Forster, Luca Carlone, Frank Dellaert, and Davide Scaramuzza. On-Manifold Preintegration for Real-Time Visual-Inertial Odometry. *IEEE Transactions on Robotics*, 33(1):1–21, 2017.
- [30] Timothy D. Barfoot and Paul T. Furgale. Associating uncertainty with three-dimensional poses for use in estimation problems. *IEEE Transactions on Robotics*, 30(3):679–693, 2014.
- [31] Frank Dellaert et al. borglab/gtsam. <https://doi.org/10.5281/zenodo.5794541>, May 2022.



## APPENDIX

This appendix accompanies the arXiv version of this paper.

### A. Odometry preintegration

We represent the uncertainty associated with 3D poses on the corresponding Lie algebra [9]. Specifically, a pose  $\mathbf{T}_{i_t} \in SE(3)$  is decomposed into a nominal pose  $\bar{\mathbf{T}}_{i_t} \in SE(3)$  and a small perturbation  $\boldsymbol{\xi}_t \in \mathbb{R}^{6 \times 1}$ :

$$\mathbf{T}_{i_t} = \bar{\mathbf{T}}_{i_t} \exp(\boldsymbol{\xi}_t^\wedge), \quad (11)$$

where  $\exp$  is the retraction operation for  $SE(3)$ , and  $\wedge$  is the *hat* operator that maps an element of  $\mathbb{R}^6$  to an element of the Lie algebra  $\mathfrak{se}(3)$ , and  $\boldsymbol{\xi}_t \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{i_t})$  is a Gaussian random variable. Consider a sequence of monotonic time steps  $(t_0, t_1, \dots, t_k)$  and the corresponding odometry estimates  $\{\{\bar{\mathbf{T}}_{i_1}^o, \boldsymbol{\Sigma}_{i_1}\}, \{\bar{\mathbf{T}}_{i_2}^o, \boldsymbol{\Sigma}_{i_2}\}, \dots, \{\bar{\mathbf{T}}_{i_k}^o, \boldsymbol{\Sigma}_{i_k}\}\}$  as shown in Figure 12. Here  $\bar{\mathbf{T}}_{i_t}^o \in SE(3)$  represents the pose of the robot frame  $\{i\}$  with respect to the odometry frame  $\{o\}$  at time  $t$  and  $\boldsymbol{\Sigma}_{i_t}$  is a positive definite matrix representing the corresponding covariance. First, we convert the absolute odometry measurements into incremental odometry between consecutive time steps:

$$\begin{aligned} \Delta \mathbf{T}_{i_t}^{i_{t+1}} &= (\mathbf{T}_{i_t}^o)^{-1} \mathbf{T}_{i_{t+1}}^o, \\ &= (\bar{\mathbf{T}}_{i_t}^o \exp(\boldsymbol{\xi}_t^\wedge))^{-1} \bar{\mathbf{T}}_{i_{t+1}}^o \exp(\boldsymbol{\xi}_{t+1}^\wedge), \\ &= \exp(-\boldsymbol{\xi}_t^\wedge) \underbrace{(\bar{\mathbf{T}}_{i_t}^o)^{-1} \bar{\mathbf{T}}_{i_{t+1}}^o}_{\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}}} \exp(\boldsymbol{\xi}_{t+1}^\wedge), \\ &= \Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}} \exp(-(\mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}})^{-1}} \boldsymbol{\xi}_t)^\wedge) \exp(\boldsymbol{\xi}_{t+1}^\wedge), \end{aligned}$$

where  $\mathbf{Ad}_()$  is the *Adjoint* operator for  $SE(3)$ . Thus, the incremental odometry can be represented as combination of a nominal pose:

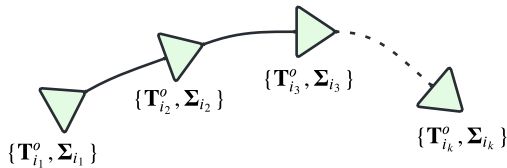
$$\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}} = (\bar{\mathbf{T}}_{i_t}^o)^{-1} \bar{\mathbf{T}}_{i_{t+1}}^o \quad (12)$$

and a perturbation:

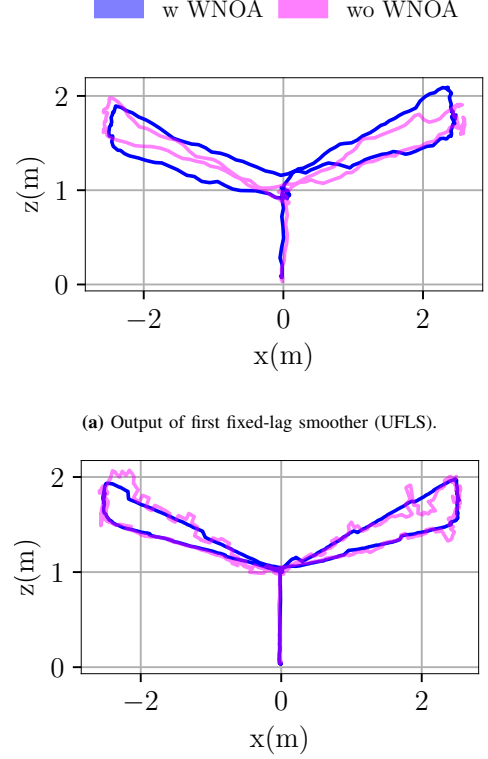
$$\exp(\boldsymbol{\xi}_{t+1}^\wedge) = \exp(-(\mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}})^{-1}} \boldsymbol{\xi}_t)^\wedge) \exp(\boldsymbol{\xi}_{t+1}^\wedge).$$

To calculate the covariance associated with incremental odometry, we apply the *Baker–Campbell–Hausdorff* (BCH) formula to the above equation and use a second-order approximation [9]:

$$\boldsymbol{\Sigma}'_{i_{t+1}} = \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}})^{-1}} \boldsymbol{\Sigma}_{i_t} \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}})^{-1}}^T + \boldsymbol{\Sigma}_{i_{t+1}} \quad (13)$$



**Fig. 12:** Visual-inertial odometry (VIO) estimates the robot pose  $\mathbf{T}_{i_t}^o$  at different times steps along with the corresponding uncertainty  $\boldsymbol{\Sigma}_{i_t}$ . In odometry preintegration multiple odometry measurements are summarized with a single preintegrated odometry measurement.



(b) Output of second fixed-lag smoother (WFLS) used for closed-loop control.

**Fig. 13:** Trajectory tracking performance with (w WNOA) and without (wo WNOA) additional smoothing. (a) The trajectory estimated by the first fixed-lag smoother (UFLS). (b) Output of the second fixed-lag smoother used for close-loop control.

Using (12) and (13), we can convert the odometry measurements into a sequence of incremental odometry measurements  $\{\{\Delta \bar{\mathbf{T}}_{i_1}^{i_2}, \boldsymbol{\Sigma}'_{i_2}\}, \{\Delta \bar{\mathbf{T}}_{i_2}^{i_3}, \boldsymbol{\Sigma}'_{i_3}\}, \dots, \{\Delta \bar{\mathbf{T}}_{i_{k-1}}^{i_k}, \boldsymbol{\Sigma}'_{i_k}\}\}$ .

The incremental odometry measurements can be composed recursively as follows.

$$\begin{aligned} \Delta \mathbf{T}_{i_t}^{i_{t+2}} &= \Delta \mathbf{T}_{i_t}^{i_{t+1}} \Delta \mathbf{T}_{i_{t+1}}^{i_{t+2}}, \\ &= \Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}} \exp(\boldsymbol{\xi}_{t+1}^\wedge) \Delta \bar{\mathbf{T}}_{i_{t+1}}^{i_{t+2}} \exp(\boldsymbol{\xi}_{t+2}^\wedge), \\ &= \Delta \bar{\mathbf{T}}_{i_t}^{i_{t+2}} \Delta \bar{\mathbf{T}}_{i_{t+1}}^{i_{t+2}} \exp(\boldsymbol{\xi}_{t+1}^\wedge) \exp(\boldsymbol{\xi}_{t+2}^\wedge), \end{aligned}$$

where  $\boldsymbol{\xi}_{t+1}'' = \mathbf{Ad}_{(\bar{\mathbf{T}}_{i_t}^{i_{t+2}})^{-1}} \boldsymbol{\xi}_{t+1}$ . The mean and covariance of the composition can be derived similar to (12) and (13):

$$\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+2}} = \Delta \bar{\mathbf{T}}_{i_t}^{i_{t+1}} \Delta \bar{\mathbf{T}}_{i_{t+1}}^{i_{t+2}}, \quad (14)$$

$$\boldsymbol{\Sigma}'_{i_{t+2}} = \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+2}})^{-1}} \boldsymbol{\Sigma}'_{i_{t+1}} \mathbf{Ad}_{(\Delta \bar{\mathbf{T}}_{i_t}^{i_{t+2}})^{-1}}^T + \boldsymbol{\Sigma}'_{i_{t+2}}. \quad (15)$$

Equations (14) and (15) provide the mean and the covariance associated with the preintegrated odometry measurement.

### B. Effect of WNOA smoothing

In this section, we provide additional details to highlight the benefit of additional smoothing for online control.

As mentioned previously, we decouple the core state estimation routine and the localization routine that is used for control. Specifically, the core state estimation is carried out using a

fixed-lag smoother (UFLS) that combines range measurement and odometry and does not impose any smoothness requirements. The output of UFLS will be non-smooth on account of corrections from range measurements and hence is not used directly for close-loop control (see Figure 13a).

The output of UFLS is then fed to a second smoother (WFLS) which imposes smoothness requirements using a white-noise-on-acceleration (WNOA) motion model. Note that the output of this particular smoother is used purely for control and does not influence the core state estimation routine or the localization RMSE. This additional smoothing could result in lower trajectory tracking accuracy but we observed empirically that without additional smoothing the trajectory tracking accuracy is lower and results in unstable flights as shown in Figure 13b.