Iterative Bias Estimation for an Ultra-Wideband Localization System

Bas van der Heijden [*] **Anton Ledergerber** [**] **Rajan Gill** [**]
Raffaello D'Andrea [**]

[*] *Delft Center for Systems and Control, TU Delft, 2628 CN Delft, The Netherlands (e-mail: b.heijden@hotmail.com).*
[**] *Institute for Dynamic Systems and Control, ETH Zurich, 8092 Zurich, Switzerland (email: {antonl, rgill, rdandrea}@ethz.ch)*

Abstract: An iterative bias estimation framework is presented that mitigates position-dependent ranging errors often present in ultra-wideband localization systems. State estimation and control are integrated, such that the positioning accuracy improves over iterations. The framework is experimentally evaluated on a quadcopter platform, resulting in improvements in the tracking performance with respect to ground truth, and also smoothing the overall flight by significantly reducing unwanted oscillations; see `https://youtu.be/J-htfbzf40U` for a video.

Keywords: Ultra-wideband technology, adaptive observer design, iterative and repetitive control, Bayesian methods, sensor fusion, recursive least squares, classification.

1. INTRODUCTION

Ultra-wideband (UWB) localization systems are one of the enabling technologies for indoor robotics (Alarifi et al. (2016)). Often, the time-of-flight of transmitted UWB radio signals is measured to acquire range measurements for positioning. Under non line-of-sight (NLOS) conditions, these time-of-flight measurements are usually biased. NLOS conditions frequently occur in indoor robotics and the resulting systematic errors in the range measurements limit the positioning accuracy of UWB localization systems, as described in Denis et al. (2003).

Previous works addressed NLOS conditions by building channel classifiers that include additional information, such as floor plans of the environment in Meissner et al. (2010), channel impulse response data of the received UWB radio signal in Schroeder et al. (2007) and references therein, or models trained with labeled data in Marano et al. (2010). Other works mitigated range errors directly, by using deep learning on channel impulse response data in Tiemann et al. (2017), or using tracking algorithms, while assuming a temporal evolution model for the range error in Denis et al. (2005) and Jourdan et al. (2005).

The scenario considered in this paper is that of an autonomous robotic agent that is tasked to traverse a given reference path repeatedly, where every repetition is called an iteration denoted by index j. The reference path is a parametrized curve, defined as

$$\boldsymbol{\sigma}(\cdot) : [0, \lambda_{\max}] \to \mathbb{R}^3 \text{ with } \lambda_{\max} \in \mathbb{R}^+. \tag{1}$$

The agent moves autonomously, therefore requiring an estimate of its state $\boldsymbol{x}$ to determine the motion control input $\boldsymbol{u}$. The scenario is situated in a densely cluttered indoor environment, causing GPS localization and conventional UWB localization to be inaccurate. In addition to a UWB transceiver, the agent carries an inertial measurement unit (IMU) and barometer, that provide angular rates, accelerations, and altitude measurements. Though the agent is able to roughly traverse the path, the agent is unable to track the reference path accurately due to the aforementioned systematic measurement errors.

Unlike previous works, this paper exploits the repetitive nature of indoor robotic applications by improving the positioning accuracy over iterations. An iterative framework is proposed that combines estimation with control, and can deal with non-static environments because the systematic range error is estimated adaptively. Apart from an IMU and barometer, the proposed framework does not require any additional information such as labeled data (e.g. from a motion-capture system) or floor plans.

The paper is organized as follows. In Section 2, we present experimentally obtained range error data from which a biased range measurement model is derived. In Section 3, the iterative bias estimator framework is presented. Subsequently, the framework is experimentally evaluated on a quadcopter platform in Section 4. We conclude the paper with an outlook in Section 5.

2. RANGE MEASUREMENT ANALYSIS

The UWB localization system considered consists of multiple UWB transceivers placed at known locations $\boldsymbol{p}^a$, which are hereafter called anchors. Each anchor a provides the agent with the time-of-flight of transmitted radio signals, such that the distance towards the anchor can be inferred. The set of range measurements is denoted by $\mathcal{Z}^{\text{uwb}}$. The standard measurement model for such a range measurement $k \in \mathcal{Z}^{\text{uwb}}$ is

$$z_k = \parallel \boldsymbol{p}^{a_k} - \boldsymbol{p}(t_k) \parallel + \nu_k^{\text{uwb}}, \tag{2}$$

where z_k, t_k, and a_k denote the measured range, the timestamp and the anchor of the range measurement k, respectively. Furthermore, $\boldsymbol{p}(t_k)$ denotes the agent's position in the global frame at the time the range measurement k

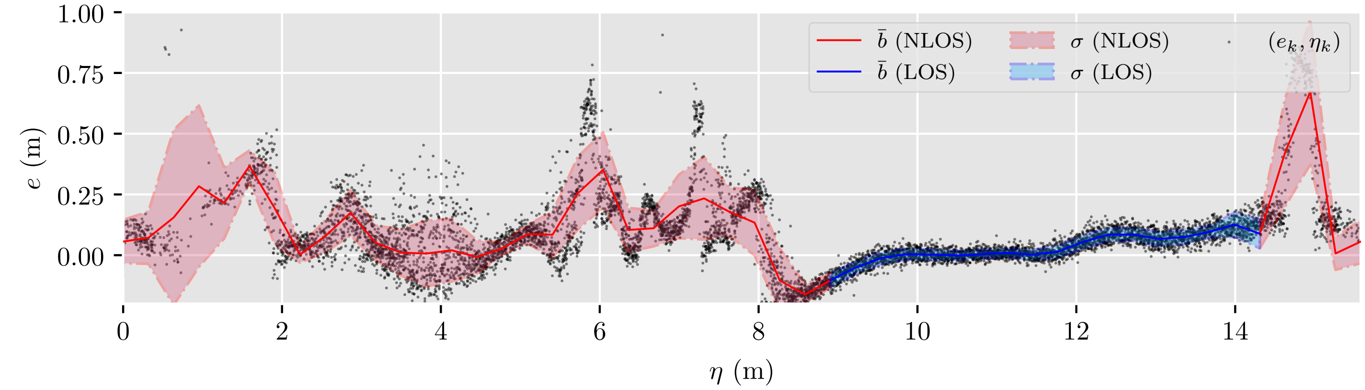

Fig. 1. The range error set of the dashed anchor in Fig. 2, with respect to the path variable η. Two piece-wise linear parametrizations (see (23), (24)) are fitted on the range error set (see (6)), using a least-squares approach with $N^b = 50$. The anchor has both LOS (blue) and NLOS (red) with parts of the path.

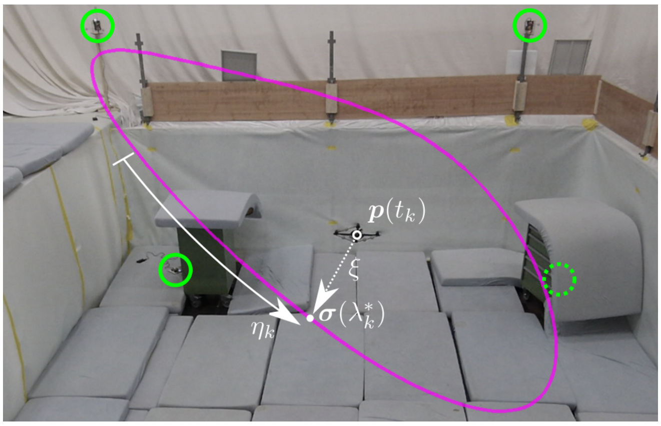

Fig. 2. The state transformation is illustrated for an experimental set-up with a closed reference path $\boldsymbol{\sigma}$ (magenta). The locations of three visible anchors are marked with solid circles. One anchor is behind a metal trolley, and marked with a dashed circle. There are three other anchors not visible in the photo.

was taken. Often, the measurement noise ν_k^{uwb} is assumed to be zero mean, normally distributed white noise.

2.1 Position Dependent Bias

The model in (2) describes the ideal situation. However, in practice range measurements are often inaccurate. Main causes for range errors are propagation delays due to small manufacturing differences in the UWB modules, and NLOS conditions. The former results in a mostly constant error. In contrast, the latter is expected to vary with respect to the geometry of the environment, the anchor placement, and the agent's position. The error in the range measurement $k \in \mathcal{Z}^{\text{uwb}}$ is defined as

$$e_k = z_k - \| \boldsymbol{p}^{a_k} - \boldsymbol{p}(t_k) \|, \tag{3}$$

and is visualized for an example path and anchor in Fig 1. Consider the set-up in Fig. 2. A quadcopter is commanded to traverse the path accurately, for 21 times, using motion-capture data and using a control approach as described in Kumar and Gill (2017). The quadcopter gathers a set of range measurements from the dashed anchor, which has has both line-of-sight (LOS) and NLOS with parts of the path. The motion-capture system provides the true position $\boldsymbol{p}(t_k)$ for all range measurements $k \in \mathcal{Z}^{\text{uwb}}$. The true position is used to calculate the range error e_k in (3), and a path variable η_k. The path variable η_k denotes the arc-length along the path, as visualized in Fig. 2. The path variable is calculated with a state transformation, explained in more detail in Kumar and Gill (2017). First, the closest point to the reference path $\boldsymbol{\sigma}(\lambda_k^*)$ must be calculated with

$$\lambda_k^* = \underset{\lambda \in [0, \lambda_{\max}]}{\arg\min} \ \| \boldsymbol{p}(t_k) - \boldsymbol{\sigma}(\lambda) \|, \tag{4}$$

such that the path variable can be calculated with

$$\eta_k = \int_0^{\lambda_k^*} \left\| \frac{d\boldsymbol{\sigma}(r)}{dr} \right\| dr. \tag{5}$$

Hence, a set of range error e_k and path variable η_k pairs,

$$\mathcal{B} = \{(e_k, \eta_k) \text{ s.t. } k \in \mathcal{Z}^{\text{uwb}}\}, \tag{6}$$

is collected. Fig. 1 shows the subset of $\mathcal{B}$ collected with the dashed anchor in Fig. 2, and illustrates the spatial range error evolution along the path. It is evident that the range error is systematic, caused by NLOS conditions, and dependent on the position along the path. When comparing the variance of the range error for positions along the path, the figure further reveals that there is a position-dependent noise component, in addition to the white noise ν_k^{uwb}.

2.2 Biased Range Measurement Model

The model in (2) is extended with a path-dependent bias $b^{a_k}(\eta_k)$ to capture the position dependency of the range error,

$$z_k = \| \boldsymbol{p}^{a_k} - \boldsymbol{p}(t_k) \| + b^{a_k}(\eta_k) + \nu_k^{\text{uwb}}. \tag{7}$$

As the bias evolution is assumed to be a correlated process with respect to the position along the path, the mean and variance of $b^{a_k}(\eta_k)$ are parametrized as functions of the path variable η_k and a set of bias parameters $\boldsymbol{\theta}^{a,\bar{b}}, \boldsymbol{\theta}^{a,\sigma^2} \in \mathbb{R}^{N^b}$. For ease of notation, the anchor superscript a is dropped in $\boldsymbol{\theta}^{a,\bar{b}}, \boldsymbol{\theta}^{a,\sigma^2}$ in the following, but do note that each anchor has its own bias profile. We denote the piece-wise linear parametrizations of its mean and variance by

$$\mathrm{E}[b(\eta_k)] = \bar{b}(\eta_k, \boldsymbol{\theta}^{\bar{b}}) \qquad \mathrm{Var}[b(\eta_k)] = \sigma^2(\eta_k, \boldsymbol{\theta}^{\sigma^2}), \tag{8}$$

and further specify this parametrization in Section 3.3. The number of parameters N^b per anchor is chosen such that a satisfactory fit on the underlying bias profile is obtained. See Fig. 1 for an example of a fit on experimental data for $N^b = 50$.

3. ITERATIVE BIAS ESTIMATOR

The iterative bias estimator estimates the bias parameters $\boldsymbol{\theta}^{\bar{b}}, \boldsymbol{\theta}^{\sigma^2}$. An attempt was made to assume constant variance, augment the agent's state $\boldsymbol{x}$ with the mean bias parameters $\boldsymbol{\theta}^{\bar{b}}$, and estimate the augmented state in a Kalman filter framework. Though stable configurations were found in simulation, the estimator was prone to diverge in practice. The divergent behavior was attributed to the augmented state being weakly observable, as observability is generally a local property for non-linear systems (Isidori (1995)), and to the model mismatch between the bias parametrization and the actual bias profile.

Instead, the control and estimation architecture in Fig. 3 is proposed. Every timestep, the motion controller receives an estimate of the state $\boldsymbol{x}$ from the online state estimator, which it uses to determine a control input $\boldsymbol{u}$ that forces the agent's position on the path. The agent's sensor measurements are fused in the online state estimator and logged over one iteration j, after which they are sent to the iterative bias estimator, with its components encircled by the dashed line in Fig. 3. The iterative bias estimator relies on a classifier to distinguish reliable (i.e. unbiased) from unreliable range measurements. The iterative framework enables the reliable measurements to be fused in a non-causal state estimator whose state trajectory estimate is used to infer the range error in both the reliable, and unreliable range measurements. The range errors are combined over iterations in a recursive least squares (RLS) filter, allowing the estimation of the bias' variance, in addition to its mean. A forgetting factor in the RLS filter allows the framework to deal with non-static environments. The updated bias parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}, \boldsymbol{\theta}^{\sigma^2}_{(j)}$ are subsequently used by the online state estimator to evaluate the bias of an anchor for a given position $\boldsymbol{p}$. The set of logged measurements in iteration j is defined as the union

$$\mathcal{Z}_{(j)} = \mathcal{Z}^{\text{uwb}}_{(j)} \cup \mathcal{Z}^{\text{acc}}_{(j)} \cup \mathcal{Z}^{\text{gyr}}_{(j)} \cup \mathcal{Z}^{\text{bar}}_{(j)}, \tag{9}$$

of the range, acceleration, angular rate, and altitude measurement sets, respectively. The components of the iterative bias estimator are specified in the following subsections.

3.1 *Classifier*

The logged measurements are sent to a classifier. The classifier must partition range measurements $\mathcal{Z}^{\text{uwb}}_{(j)}$ into a reliable partition $\mathcal{Z}^{\text{uwb,rel}}_{(j)}$ and an unreliable partition $\mathcal{Z}^{\text{uwb,unrel}}_{(j)}$. The classification algorithm is an adapted version of the Residual Weighting algorithm presented in Chen (1999), which is based on a least squares (LS) algorithm. The algorithm herein differs in that its task is to classify reliable measurements, instead of estimating the position. Furthermore, the LS algorithm uses the altitude measurements in addition to the range measurements.

When the measurements $\mathcal{Z}_{(j)}$ are received, the range measurements are first partitioned into N^{sub} subsequences, i.e.

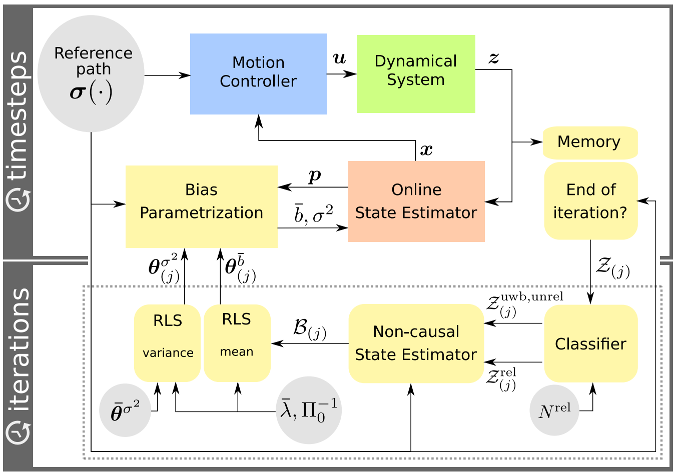

Fig. 3. The iterative bias estimation framework, integrated into a common architecture for controlling the motion of a dynamical system.

$$\mathcal{Z}^{\text{uwb}}_{(j)} = \{\mathcal{Z}^{\text{uwb}}_{(j),1}, \mathcal{Z}^{\text{uwb}}_{(j),2}, \ldots, \mathcal{Z}^{\text{uwb}}_{(j),N^{\text{sub}}}\}. \tag{10}$$

Each subsequence $\mathcal{Z}^{\text{uwb}}_{(j),n} \in \mathcal{Z}^{\text{uwb}}_{(j)}$ contains consecutive range measurements obtained with different anchors. Hence, if N^a is the number of anchors from which sequential range measurements are obtained, such a subsequence is of size $|\mathcal{Z}^{\text{uwb}}_{(j),n}| = N^a$. The classifier labels N^{rel} range measurements per subsequence $\mathcal{Z}^{\text{uwb}}_{(j),n}$ as reliable, where N^{rel} is set to the expected number of anchors with LOS at any given time. Reliable range measurements are consistent with each other, which is reflected in a low residual of a LS position estimate given reliable range measurements. Therefore, a LS position estimate is calculated for different range measurement combinations $\mathcal{C}^i_{(j),n} \subset \mathcal{Z}^{\text{uwb}}_{(j),n}$, $|\mathcal{C}^i_{(j),n}| = N^{\text{rel}}$. These combination are obtained by taking all $|\mathcal{Z}^{\text{uwb}}_{(j),n}|$ choose N^{rel} different combinations, resulting in N^{c} combinations per subsequence. The combination $\mathcal{C}^i_{(j),n}$ that results in the lowest LS position estimate residual $r^i_{(j),n}$ is classified as reliable, i.e.

$$\mathcal{Z}^{\text{uwb,rel}}_{(j),n} = \mathcal{C}^i_{(j),n} \text{ s.t. } i = \underset{i \in 1,2,\ldots,N^c}{\arg\min}\; r^i_{(j),n}. \tag{11}$$

The altitude measurements $l \in \mathcal{Z}^{\text{bar}}_{(j),n}$, measured within the time spanned by the range measurement timestamps of $\mathcal{Z}^{\text{uwb}}_{(j),n}$, are additionally fused into the LS estimate. The residual of such a LS estimate is given by

$$r^i_{(j),n} = \min_{\boldsymbol{p} \in \mathbb{R}^3} \left(r^{i,\text{uwb}}_{(j),n}(\boldsymbol{p}) + r^{\text{bar}}_{(j),n}(p_z) \right), \tag{12}$$

where p_z is the z-coordinate of the position $\boldsymbol{p}$, and

$$r^{i,\text{uwb}}_{(j),n}(\boldsymbol{p}) = \frac{1}{\Sigma^{\text{uwb}}} \sum_{k \in \mathcal{C}^i_{(j),n}} (z_k - \| \boldsymbol{p}^{a_k} - \boldsymbol{p} \|)^2, \tag{13}$$

$$r^{\text{bar}}_{(j),n}(p_z) = \frac{1}{\Sigma^{\text{bar}}} \sum_{l \in \mathcal{Z}^{\text{bar}}_{(j),n}} (z_l - p_z)^2. \tag{14}$$

where Σ^{uwb} and Σ^{bar} are the corresponding measurement noise variances. Then, the set of reliable range measurements over a complete iteration is given as the union of the combinations with the lowest residuals, i.e.

$$\mathcal{Z}^{\text{uwb,rel}}_{(j)} = \{\mathcal{Z}^{\text{uwb,rel}}_{(j),1}, \mathcal{Z}^{\text{uwb,rel}}_{(j),2}, \ldots, \mathcal{Z}^{\text{uwb,rel}}_{(j),N^{\text{sub}}}\}, \tag{15}$$

and the unreliable set as the complement to $\mathcal{Z}^{\text{uwb,rel}}_{(j)}$, i.e.

$$\mathcal{Z}^{\text{uwb,unrel}}_{(j)} = \mathcal{Z}^{\text{uwb}}_{(j)} \setminus \mathcal{Z}^{\text{uwb,rel}}_{(j)}. \tag{16}$$

3.2 Non-causal Estimator

For each iteration j, the non-causal estimator estimates the range error set $\mathcal{B}_{(j)}$ defined in (6). For this, it requires estimates of the position at every time $t_k, k \in \mathcal{Z}^{\text{uwb}}_{(j)}$. An adapted version of the continuous-time batch optimization framework presented in Furgale et al. (2015) is used, as follows.

The measurement set $\mathcal{Z}^{\text{rel}}_{(j)} = \{\mathcal{Z}^{\text{uwb,rel}}_{(j)}, \mathcal{Z}^{\text{acc}}_{(j)}, \mathcal{Z}^{\text{gyr}}_{(j)}, \mathcal{Z}^{\text{bar}}_{(j)}\}$ is used to find the maximum a-posteriori estimate of the state trajectory $\boldsymbol{x}(t, \boldsymbol{c})$, expressed as the weighted sum of a finite set of temporal basis functions with parameters $\boldsymbol{c}$. It is assumed that the position trajectory $\boldsymbol{p}(t)$ can be inferred from the state trajectory $\boldsymbol{x}(t, \boldsymbol{c})$. Using Bayes' law, the posterior estimate can be written as

$$p(\boldsymbol{x}(t,\boldsymbol{c})|\mathcal{Z}^{\text{rel}}_{(j)}) = \frac{p(\boldsymbol{x}(t,\boldsymbol{c}))}{p(\mathcal{Z}^{\text{rel}}_{(j)})} p(\mathcal{Z}^{\text{rel}}_{(j)}|\boldsymbol{x}(t,\boldsymbol{c})). \tag{17}$$

Assuming that the sensor measurements are conditionally independent given $\boldsymbol{x}(t, \boldsymbol{c})$, the posterior is rewritten as

$$p(\boldsymbol{x}(t,\boldsymbol{c})|\mathcal{Z}^{\text{rel}}_{(j)}) = \frac{p(\boldsymbol{x}(t,\boldsymbol{c}))}{p(\mathcal{Z}^{\text{rel}}_{(j)})} \prod_{m \in \mathcal{Z}^{\text{rel}}_{(j)}} p(z_m|\boldsymbol{x}(t_m,\boldsymbol{c})). \tag{18}$$

The measurement likelihood for an arbitrary sensor measurement m with a Gaussian distribution is given by

$$p(z_m|\boldsymbol{x}(t_m,\boldsymbol{c})) = \mathcal{N}(h_m(\boldsymbol{x}(t_m,\boldsymbol{c})), \Sigma_m), \tag{19}$$

where the measurement model $h_m(\boldsymbol{x}(t_m, \boldsymbol{c}))$ denotes the mean of z_m, and Σ_m denotes the noise covariance matrix. Only reliable range measurements are used in the non-causal estimator. Nevertheless, small differences in the antennas cause a constant offset o^{uwb}_a, slightly different for each anchor. To compensate for o^{uwb}_a, the model

$$h_k(\boldsymbol{x}(t_k,\boldsymbol{c}), o^{\text{uwb}}_{a_k}) = \parallel \boldsymbol{p}^{a_k} - \boldsymbol{p}(\boldsymbol{x}(t_k,\boldsymbol{c})) \parallel + o^{\text{uwb}}_{a_k} \tag{20}$$

is used for the reliable range measurements $k \in \mathcal{Z}^{\text{uwb,rel}}_{(j)}$. Similarly, slow drifting behavior in the barometer can occur, and causes an offset o^{bar} in the altitude measurements that is approximately constant over one iteration. Therefore, a constant offset o^{bar} is estimated each iteration by assuming the barometer measurement model

$$h_l(\boldsymbol{x}(t_l,\boldsymbol{c}), o^{\text{bar}}) = p_z(\boldsymbol{x}(t_l,\boldsymbol{c})) + o^{\text{bar}} \tag{21}$$

for the altitude measurements $l \in \mathcal{Z}^{\text{bar}}_{(j)}$. See Ledergerber and D'Andrea (2018) for the measurement models of a biased IMU, with the state modeled as a rigid body. In case the IMU is unbiased, the bias terms in the models can be omitted. Then, the maximum a-posteriori estimates $({}^*\boldsymbol{c}, {}^*o^{\text{uwb}}_a, {}^*o^{\text{bar}})$ are found by minimizing the negative logarithm of the posterior (18),

$$\min_{\boldsymbol{c}, o^{\text{uwb}}_a, o^{\text{bar}}} \sum_{m \in \mathcal{Z}^{\text{rel}}_{(j)}} -\log(p(z_m|\boldsymbol{x}(t_m,\boldsymbol{c}), o^{\text{uwb}}_a, o^{\text{bar}})), \tag{22}$$

where the term $p(\mathcal{Z}^{\text{rel}}_{(j)})$ is omitted, as it does not influence the optimized solution, and the prior $p(\boldsymbol{x}(t, \boldsymbol{c}))$ is omitted because the ratio of measurements to optimization variables is large. This reduces (22) to a maximum-likelihood problem (Furgale et al. (2015)). A local optimization is performed, where the estimates of the online state estimator serve as the initial guess for the optimization. With the resulting position trajectory $\boldsymbol{p}(t, {}^*\boldsymbol{c})$, an estimate of the range error set $\mathcal{B}_{(j)}$ is calculated with (4), (5), and (6). It is important to note here, that the range error is calculated for all range measurements, i.e. for measurements classified as reliable and for measurements classified as unreliable. The barometer offset ${}^*o^{\text{bar}}$ is provided to the classifier and online state estimator, such that the drift is compensated for in the altitude measurements $z_l, l \in \mathcal{Z}^{\text{bar}}$ before being incorporated in (14) and before being fused in the online state estimator.

3.3 Recursive Least Squares Filter

The bias parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}, \boldsymbol{\theta}^{\sigma^2}_{(j)}$ are estimated recursively by combining the range error estimates $\mathcal{B}_{(j)}$ with the sets of range error estimates from all previous iterations, i.e. $\mathcal{B}_{(1..j)} = \cup_{i=1}^{j} \mathcal{B}_{(i)}$, yielding the two piece-wise linear bias parametrizations introduced in (8) and defined as

$$\bar{b}(\eta_k, \boldsymbol{\theta}^{\bar{b}}_{(j)}) = \boldsymbol{u}^\top(\eta_k)\boldsymbol{\theta}^{\bar{b}}_{(j)} \tag{23}$$

$$\sigma^2(\eta_k, \boldsymbol{\theta}^{\sigma^2}_{(j)}) = \boldsymbol{u}^\top(\eta_k)\boldsymbol{\theta}^{\sigma^2}_{(j)}. \tag{24}$$

The elements of vector

$$\boldsymbol{u}^\top(\eta_k) = [u_1(\eta_k), u_2(\eta_k), \ldots, u_{N^b}(\eta_k)] \tag{25}$$

are zero, except for

$$u_i(\eta_k) = 1 - \frac{\eta_k - \tau_i}{\tau_{i+1} - \tau_i} \qquad u_{i+1}(\eta_k) = \frac{\eta_k - \tau_i}{\tau_{i+1} - \tau_i}, \tag{26}$$

where i is s.t. $\tau_i \leq \eta_k < \tau_{i+1}$, and where the knots τ_i of the piece-wise linear parameterization are given as

$$\tau_i = \frac{\eta_{\max}}{N^b - 1}(i-1) \text{ for } i = \{1, \ldots, N^b\}, \tag{27}$$

where $\eta_{\max}$ is the arc-length of the reference path. To solve for the mean bias parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}$, a regressor matrix $U \in \mathbb{R}^{N^b \times |\mathcal{B}_{(1..j)}|}$ and range error vector $\boldsymbol{e} \in \mathbb{R}^{|\mathcal{B}_{(1..j)}|}$ are constructed with the set $\mathcal{B}_{(1..j)}$ of all range error, path variable pairs up to iteration j, i.e.,

$$U(\mathcal{B}_{(1..j)}) = [\boldsymbol{u}(\eta_1), \boldsymbol{u}(\eta_2), ..., \boldsymbol{u}(\eta_{|\mathcal{B}_{(1..j)}|})] \tag{28}$$

$$\boldsymbol{e}(\mathcal{B}_{(1..j)}) = [e_1, e_2, ..., e_{|\mathcal{B}_{(1..j)}|}]^\top. \tag{29}$$

Subsequently, the problem of finding the mean bias parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}$ can be formulated as a least squares problem,

$$\min_{\boldsymbol{\theta}^{\bar{b}}_{(j)}} \left[(\boldsymbol{\theta}^{\bar{b}}_{(j)} - \bar{\boldsymbol{\theta}}^{\bar{b}})^\top \Pi_0^{-1} (\boldsymbol{\theta}^{\bar{b}}_{(j)} - \bar{\boldsymbol{\theta}}^{\bar{b}}) + \parallel \boldsymbol{e} - U^\top \boldsymbol{\theta}^{\bar{b}}_{(j)} \parallel^2 \right], \tag{30}$$

where the initial guess $\bar{\boldsymbol{\theta}}^{\bar{b}}$ and the weighting matrix Π_0^{-1} represent the prior knowledge. Each iteration, $\boldsymbol{\theta}^{\bar{b}}_{(j)}$ is estimated by solving (30) recursively with the regularized RLS algorithm described in Section 21.4 of Sayed and Kailath (2000). A forgetting factor λ is included in the algorithm, so that the framework is adaptive to non-static environments. Similarly, the variance bias parameters $\boldsymbol{\theta}^{\sigma^2}_{(j)}$ are estimated, by recursively solving

$$\min_{\boldsymbol{\theta}^{\sigma^2}_{(j)}} \left[(\boldsymbol{\theta}^{\sigma^2}_{(j)} - \bar{\boldsymbol{\theta}}^{\sigma^2})^\top \Pi_0^{-1} (\boldsymbol{\theta}^{\sigma^2}_{(j)} - \bar{\boldsymbol{\theta}}^{\sigma^2}) + \parallel \boldsymbol{r} - U^\top \boldsymbol{\theta}^{\sigma^2}_{(j)} \parallel^2 \right], \tag{31}$$

where $\bar{\boldsymbol{\theta}}^{\sigma^2}$ is the initial guess for the variance. In practice, $\bar{\boldsymbol{\theta}}^{\sigma^2}$ is set to a large value, while $\bar{\boldsymbol{\theta}}^{\bar{b}}$ is initialized to zero, as the shape of the bias profile is unknown. Vector $\boldsymbol{r}$ denotes the squared residual between the range error data and mean bias fit. The squared residual is a common metric for estimating variances, as described in Buckley et al. (1988), and is defined as

$$\boldsymbol{r} = (\boldsymbol{e} - U^{\mathsf{T}}\boldsymbol{\theta}^{\bar{b}}_{(j)}) \odot (\boldsymbol{e} - U^{\mathsf{T}}\boldsymbol{\theta}^{\bar{b}}_{(j)}), \tag{32}$$

where $\odot$ is the element-wise multiplication operator. The estimated parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}, \boldsymbol{\theta}^{\sigma^2}_{(j)}$ are provided to the bias parametrization block in Fig. 3, such that the online state estimator can evaluate the mean and variance bias parametrizations in (23), and (24) for a given position $\boldsymbol{p}$, using (4), and (5).

4. EXPERIMENTAL EVALUATION ON A QUADCOPTER PLATFORM

The proposed iterative bias estimation framework was experimentally evaluated on a quadcopter platform. The quadcopter is the autonomous robotic agent, tasked to fly a closed reference path repeatedly. First, the implementation details are provided, showing how the framework was able to run in real-time. Subsequently, the experimental results are presented.

4.1 Implementation

The motion controller and online state estimator are run on a Snapdragon flight board, located onboard a quadcopter. The iterative bias estimator is run in a separate process, offboard on a laptop with an i7-Intel processor. The onboard and offboard processes communicate with a TCP client/server set-up. Both the anchors and quadcopter are equipped with DWM 1000 modules and communicate using the two-way ranging algorithm with repeated reply described in Mueller et al. (2015). The quadcopter communicates with every anchor in sequential order at a frequency of 200 Hz. The MPU9250 IMU and BMP280 barometer modules are integrated on the Snapdragon flight board, and provide measurements with a frequency of 1 kHz and 50 Hz, respectively.

The online state estimator estimates the position, velocity, orientation representation, and angular velocity of the quadcopter in a Kalman filter framework, specified in Ledergerber and D'Andrea (2018). The range measurements are fused using model (7) via an unscented Kalman filter. For every position sigma point $\boldsymbol{p}^{(i)}$, the resulting $(\bar{b}, \sigma^2)^{(i)}$ can be incorporated into the range measurement update.

A path following controller is implemented as the motion controller of the quadcopter. The controller minimizes the agent's distance towards the path and makes the agent traverse along the path with a constant reference speed of 1.5 m/s. See Kumar and Gill (2017) for more details.

It is assumed that the quadcopter has LOS with 5 out of 7 anchors at any given time. Therefore, N^{rel} is set to 5, and $N^c = 21$ for $|\mathcal{Z}^{\mathrm{uwb}}_{(j),n}| = 7$. For an iteration j with a duration of 10 seconds, roughly $|\mathcal{Z}^{\mathrm{uwb}}_{(j)}| \approx 2000$ range measurements are logged, resulting in $N^{\mathrm{sub}} \approx |\mathcal{Z}^{\mathrm{uwb}}_{(j)}|/N^a \approx 286$ partitions, as defined in (10). For every partition, (12) must be evaluated for all $N^c = 21$ combinations, resulting in a total of $286 * 21 = 6006$ computations of (12). All computations are independent from one another, so that the computations can be parallelized.

The non-causal state estimator is implemented similarly to Ledergerber and D'Andrea (2018). The state trajectory $\boldsymbol{x}(t)$ is assumed to consist of a position trajectory $\boldsymbol{p}(t)$ and quaternion trajectory $\boldsymbol{q}(t)$. Both trajectories are parametrized by a set of uniform cubic B-splines. A small uniform knot spacing of the splines increases the approximation power of the parametrization. However, we find that a larger spacing of 500 ms performs better, because it has a regularizing effect on the optimization in (22). The IMU is assumed to be unbiased, because it showed no drifting behavior after calibrating prior to flying.

The RLS filter has a forgetting factor $\bar{\lambda} = 0.9$, which means that the estimated parameters $\boldsymbol{\theta}^{\bar{b}}_{(j)}, \boldsymbol{\theta}^{\sigma^2}_{(j)}$ depend on roughly the last 10 iterations. A large uncertainty in the bias is encoded by setting each element in the initial variance $\bar{\boldsymbol{\theta}}^{\sigma^2}$ to a large value of 0.15^2. The weight matrix is set to be $\Pi_0^{-1} = UU^{\mathsf{T}}$, where U is constructed with (28) using N^{sub} uniformly sampled path variables η along the path. The reference path is a closed path, so the parametrization in (25), (26), and (27) is adjusted accordingly.

4.2 Experimental Results

The experiments were performed in the Flying Machine Arena of ETH Zurich (Lupashin et al. (2014)). See Fig. 2 for the anchor placement and reference path the quadcopter is tasked to fly. Objects (e.g. metal trolleys) placed in the room cause some anchors to have NLOS with parts of the path, inducing a position-dependent bias in the range measurements. See Fig. 1 for an example.

Minimizing the distance towards the path is considered to be the most important control objective, because it ensures a safe flight. This distance is referred to as the tracking error, and is defined as

$$\xi = \parallel \boldsymbol{p} - \boldsymbol{\sigma}(\lambda^*) \parallel . \tag{33}$$

The first time-derivative of the tracking error is denoted by $\dot{\xi}$. A large $\dot{\xi}$ means the distance error varies rapidly, which expresses itself in an oscillatory flight around the reference path. The mean absolute error (MAE) in ξ and $\dot{\xi}$ is shown in Fig. 4 for four runs. The performance of the proposed framework is compared with that of two different experiments. In a base-case (base) experiment, no bias estimation is performed. In a motion-capture (mc) experiment, the true range error set $\mathcal{B}$ is obtained with a motion-capture system, and subsequently fed to the RLS filter such that the best-possible bias fit is obtained. The performance of the motion-capture experiment is interpreted as the best case. Each experiment is executed multiple times. Only the mean performance μ with one standard deviation σ is plotted for the base-case and motion-capture experiments.

Fig. 4 shows that the proposed framework consistently improves performance over iterations in both ξ, and $\dot{\xi}$, with respect to the base-case experiments. Concerning $\dot{\xi}$, a similar performance level with respect to the motion-capture experiment is obtained. The improvements in $\dot{\xi}$ are ascribed to the inclusion of a position-dependent noise component in addition to the mean bias, and results in

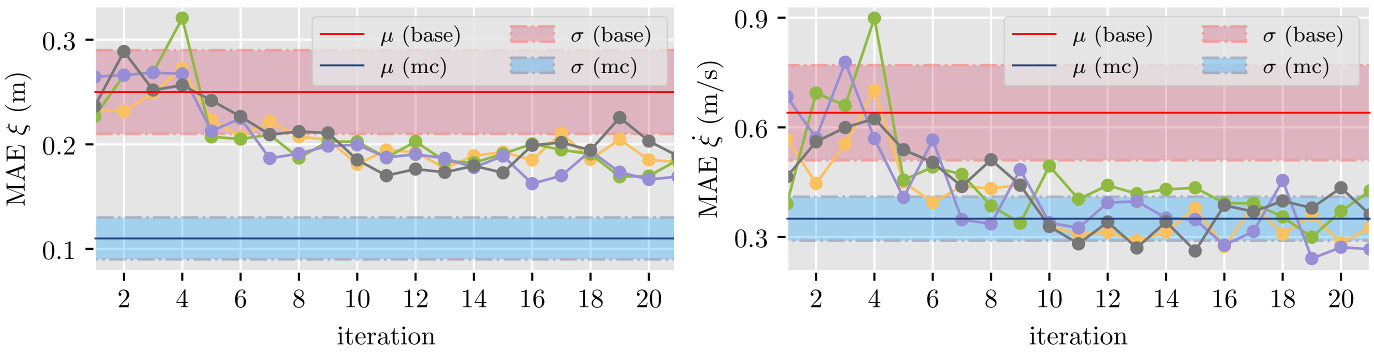

Fig. 4. The path tracking performance of the proposed framework is plotted for four runs (solid-dotted), alongside the mean performance μ with one standard deviation σ of the base-case (red) and motion-capture (blue) experiments.

reduced oscillations during flight. See `https://youtu.be/J-htfbzf40U` for a video, showcasing the improved flight performance. The performance of the iterative bias estimation framework largely depends on the ability of the classifier to correctly distinguish reliable from unreliable measurements. Therefore, the difference in path tracking performance improvement between the framework (26%), and best case (56%), is primarily attributed to the sub-optimal performance of the classifier.

5. OUTLOOK

Improvements in the classifier are expected to be most effective in improving the overall performance of the framework. For example, the additional inclusion of IMU measurements into the presented classification scheme is worth investigating. Also, completely different classification schemes based on channel impulse response data should be tested and could lead to a reduction of the classifier's computational complexity. Finally, it would be interesting to apply iterative learning control on top of the presented framework for high-performance maneuvering under UWB.

ACKNOWLEDGEMENTS

The authors would like to thank Marc-Andrè Corzillius for the design of the electrical hardware and Michael Egli for construction of the platform. Many people have contributed to the Flying Machine Arena in which this research was conducted. A list of these people can be found at: `http://flyingmachinearena.org/people`.

REFERENCES

Alarifi, A., Al-Salman, A., Alsaleh, M., Alnafessah, A., Al-Hadhrami, S., Al-Ammar, M., and Al-Khalifa, H. (2016). Ultra wideband indoor positioning technologies: Analysis and recent advances. *Sensors*, 16(707).

Buckley, M., Eagleson, G., and Silverman, B. (1988). The estimation of residual variance in nonparametric regression. *Biometrika*, 75, 189–199.

Chen, P. (1999). A non-line-of-sight error mitigation algorithm in location estimation. *IEEE Wireless Communications and Networking Conference*, 1, 316–320.

Denis, B., Keignart, J., and Daniele, N. (2003). Impact of nlos propagation upon ranging precision in uwb systems. In *IEEE Conference on Ultra Wideband Systems and Technologies, 2003*, 379–383.

Denis, B., Ouvry, L., Uguen, B., and Tchoffo-Talom, F. (2005). Advanced bayesian filtering techniques for uwb tracking systems in indoor environments. In *2005 IEEE International Conference on Ultra-Wideband*, 638–643.

Furgale, P., Tong, C., Barfoot, T., and Shin, S. (2015). Continuous-time batch trajectory estimation using temporal basis functions. *International Journal of Robotics Research*, 34(14), 1688–17101.

Isidori, A. (1995). *Nonlinear control systems*. Springer.

Jourdan, D.B., Deyst, J.J., Win, M.Z., and Roy, N. (2005). Monte carlo localization in dense multipath environments using uwb ranging. In *2005 IEEE International Conference on Ultra-Wideband*, 314–319.

Kumar, S. and Gill, R. (2017). Path following for quadrotors. *IEEE Conference on Control Technology and Applications*, 2075–2081.

Ledergerber, A. and D'Andrea, R. (2018). Calibrating away inaccuracies in uwb range measurements: A maximum likelihood approach. *IEEE Access*, 6, 78719–78730.

Lupashin, S., Hehn, M., Mueller, M., Schoellig, A., Sherback, M., and D'Andrea, R. (2014). A platform for aerial robotics research and demonstration: The flying machine arena. *Mechatronics*, 24(1), 41–54.

Maranó, S., Gifford, W.M., Wymeersch, H., and Win, M.Z. (2010). Nlos identification and mitigation for localization based on uwb experimental data. *IEEE Journal on Selected Areas in Communications*, 28(7), 1026–1035.

Meissner, P., Steiner, C., and Witrisal, K. (2010). Uwb positioning with virtual anchors and floor plan information. In *2010 7th Workshop on Positioning, Navigation and Communication*, 150–156.

Mueller, M., Hamer, M., and D'Andrea, R. (2015). Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation. *IEEE ICRA*, 1730–1736.

Sayed, A. and Kailath, T. (2000). Recursive least-squares adaptive filters. In *Digital Signal Processing Handbook*, chapter 21. CRC press LLC.

Schroeder, J., Galler, S., Kyamakya, K., and Jobmann, K. (2007). Nlos detection algorithms for ultra-wideband localization. In *2007 4th Workshop on Positioning, Navigation and Communication*, 159–166.

Tiemann, J., Pillmann, J., and Wietfeld, C. (2017). Ultra-wideband antenna-induced error prediction using deep learning on channel response data. In *2017 IEEE 85th Vehicular Technology Conference*, 1–5.