Quadrotor State Estimation with IMU and Delayed Real-time Kinematic GPS

Kanishke Gamagedara, Taeyoung Lee, and Murray Snyder

***Abstract*—This paper presents an estimation scheme for the six degree-of-freedom position and attitude of an aerial vehicle by integrating an inertial measurement unit and a low-cost real-time kinematic GPS unit, which delivers a precise relative position measurement corrupted by a time delay. Assuming that the time delay is known, the extended Kalman filter is generalized to fuse the time-lagged position measurement from GPS with the synchronous attitude and angular velocity measurements from IMU. More specifically, it is formulated as an optimal prediction where the past state is corrected by the position measurement before being propagated up to the current time with the history of the IMU measurements. This provides a compact formulation to merge multiple sensors with varying time-delays in an optimal fashion. The efficacy of the proposed approach is illustrated by a numerical example, experimental data collected over a Navy research vessel, and an outdoor autonomous flight of a quadrotor unmanned aerial vehicle.**

I. Introduction

The capability to estimate the position and the attitude of any unmanned aerial vehicle (UAV) is critical to complete its mission safely and successfully. For indoor flights, the position is often measured by motion capture systems with higher bandwidth and accuracy, or using vision based techniques. Whereas in outdoor flights, the most common approach for position measurement is using navigation satellite systems such as GPS. However, the commonly used single point position GPS sensors have a considerable error, typically with a standard deviation of $1\,\mathrm{m}$. The disturbances to signals broadcasted by satellites, such as multi-path error or atmospheric delays, contribute to this error [1].

The real-time kinematic (RTK) GPS systems minimize these errors by integrating two GPS receivers, namely a base GPS receiver and a rover GPS receiver. The base GPS receiver sends its observations to the rover GPS receiver through a telemetry unit. If both GPS units are in close proximity to each other, it can be assumed that the sources of external measurement errors are similar on both of them. The rover GPS receiver compares the base measurements with the GPS signals it received, and corrects for the external disturbances, thereby providing a centimeter level accuracy [2]. This correction takes a non-trivial amount of time for calculation, especially for low-cost RTK GPS units built in a small form factor. When the telemetry communication delay is added on top of the computational delay, the position measurement from RTK GPS is not suitable for autonomous flight of UAV.

Another challenge is that the attitude should be estimated concurrently with the position. It is commonly measured with an inertial measurement unit (IMU) that measures the acceleration, angular velocity, and magnetic field resolved in the body-fixed frame. A typical IMU provides those measurements in real-time with a higher frequency. As such, it is required to integrate the delayed measurements of GPS with the synchronous IMU measurements for the estimation of the six-dimensional position and attitude of UAV.

Inertial measurement sensors provide attitude and acceleration measurements at a high frequency. While the position can be estimated by integrating accelerometer measurement, its accuracy degrades quickly over time. In contrast, GPS sensors provide an absolute positioning with a lower frequency. This complementary nature of the IMU and GPS sensors yields a natural way of providing improved and reliable navigation solutions [3]. For this reason, integrating IMU and GPS measurements for navigation has been well studied over the last few decades [4], [5], [6].

In general, GPS and IMU integration can be primarily categorized into two types: tightly coupled and loosely coupled systems [7]. Tightly coupled systems process raw IMU and raw GPS observations, and they can provide a solution even when less than four satellites are available [8]. The loosely coupled systems use measurements already processed by separate IMU and GPS sensors to provide an integrated solution. This method is cheaper as the algorithms can be applied to off-the-shelf sensors, but may not perform well if the number of satellites visible to the GPS sensor is low. Several variations of both of these methods have been tested on UAVs by various researchers [9], [10], [11].

In this work, we focus on loosely coupled integration of IMU measurements and RTK GPS with a non-trivial delay. Incorporating the delayed measurements in a Kalman filter has been discussed in [12], where a minimum mean squared error estimator is presented for a linear Gaussian system. Alternatively, delayed measurements are considered by re-calculating the states through the delay period [13]. To reduce the computational cost associated with the re-calculation, it is proposed to update the covariance matrix as if the measurement was arrived without a delay, given that the measurement transition matrix and the measurement noise covariance matrix are known. Then, when the delayed measurement actually arrives, the state correction is executed with an additional correction term which has a cheaper computational cost. This makes the estimates suboptimal during the delay period, but

Kanishke Gamagedara, Taeyoung Lee, and Murray Snyder, Department of Mechanical and Aerospace Engineering, The George Washington University, Washington DC, USA. {kanishkegb,tylee,snydermr@gwu.edu}

This research has been supported in parts by the grant USNA N0018920P0364.

becomes optimal when the delayed measurement arrives.

The above approaches assume that the filter states are not corrected by any other measurements during the delay period, i.e., all of the sensors are subject to the same time delay. To overcome this problem, alternative approaches have been suggested. To ensure the optimality of the estimates through the delay period, in [14], it is proposed to run a secondary estimator in a parallel fashion through the delay period, and to fuse it to the main filter once the delayed measurement arrives, thereby improving [13]. Alternatively, it is also proposed to extrapolate the delayed measurement to the current time, and then to add a correction to the estimated state after incorporating the measurement. This approach is applicable to the cases where the measurement transition matrix and the measurement noise covariance matrix are unknown through the delay period.

A similar approach has been used in [15] to integrate delayed GPS measurement with non-delayed IMU measurements to autonomously navigate a UAV, but using versions of sigma-point Kalman filters. Given the delay is known, this method augments the state vector and the covariance matrix with the ones when the measurement is supposed to arrive. Then, the augmented state and covariance matrix are propagated until the delayed measurement actually arrives, which are used to calculate the cross covariance between the measurement and the delayed state.

The above mentioned techniques assume that the delay is known. Authors in [16] have presented a linear Kalman filter that incorporates delays using covariance union technique, which is applicable when the delay is unknown.

In this paper, we consider a pose estimation with RTK GPS and IMU, where only the position and velocity measurements from RTK GPS are delayed by a given fixed time, and all the other measurements from IMU are acquired frequently without any latency. The proposed approach has an intuitive interpretation as an optimal prediction, where the current measurement is used to predict prospective states. When the delayed position measurement becomes available, the prior state corresponding to the measurement instance is corrected, and then it is propagated over the delayed period up to the current time while being adjusted by the saved IMU measurements. As such, it is a generalization of [13] to the sensor fusion between multiple sensors where only a part of measurements is delayed. Instead of making an approximation to the covariance matrix, the IMU measurements are sub-sampled while the corrected prior state is propagated to the current time. As such, this approach incorporates all of the information available to estimate the current state without sacrificing the optimality while utilizing the improved computing power and the higher memory capacity of modern computing modules. There is no additional inherent approximation to depreciate optimality, except sub-sampling IMU measurements. The desirable properties of the proposed approach are illustrated by a numerical example with a detailed benchmark study against other techniques. They are further demonstrated by experimental results where the proposed estimator is applied to autonomous flight of a quadrotor UAV connected to a low cost RTK GPS unit. We further present the detailed software structures to implement the proposed estimation scheme.

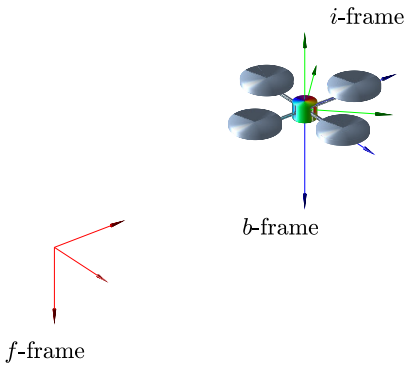


Fig. 1. Frame definitions: red - fixed north-east-down frame, blue: body frame fixed on the body of the UAV, green: frame fixed on the IMU frame used for its measurements

II. Problem Formulation

In this section, we present the mathematical model of the six degree of freedom kinematics considered in this paper, and the measurements from IMU and GPS.

A. Kinematics

Let the b-frame be the body-fixed frame of the quadrotor UAV with the origin located at its center of gravity. The local north-east-down (NED) frame is defined as the f-frame, with the origin at the base GPS sensor. We distinguish the frame fixed to the IMU, in which the IMU measurement readings are resolved, from the b-frame, and it is denoted by the i-frame. These frames are illustrated in Figure 1. The orientation of the IMU with respect to the body-fixed frame is described by a fixed rotation matrix $R_{bi} \in \mathsf{SO(3)}$, where the three-dimensional special orthogonal group is given by

$$\mathsf{SO(3)} = \{R \in \mathbb{R}^{3\times 3} \,|\, R^T R = I_{3\times 3},\ \det[R] = +1\}.$$

Let $x \in \mathbb{R}^3$ be the position of the origin of the b-frame, or the center of gravity of the UAV. The corresponding velocity and acceleration are denoted by $v \in \mathbb{R}^3$ and $a \in \mathbb{R}^3$, respectively. All of x, v, and a are resolved in the f-frame. The rotation matrix from the b-frame to the f-frame is denoted by $R \triangleq R_{fb} \in \mathsf{SO(3)}$. Further, $\Omega \in \mathbb{R}^3$ is the angular velocity of the quadrotor resolved in the b-frame. The equations of motion for (x, v, R) are given by

$$\dot{x} = v, \tag{1}$$

$$\dot{v} = a, \tag{2}$$

$$\dot{R} = R\hat{\Omega}. \tag{3}$$

Here, the hat map $\wedge : \mathbb{R}^3 \to \mathfrak{so}(3)$ is defined such that $\hat{x}y = x \times y$ and $\hat{x}^T = -\hat{x}$ for any $x, y \in \mathbb{R}^3$. The inverse of hat map is denoted by the vee map $\vee : \mathfrak{so}(3) \to \mathbb{R}^3$. The set of 3 by 3 skew-symmetric matrices are denoted by $\mathfrak{so}(3) = \{S \in \mathbb{R}^{3\times 3} \,|\, S = -S^T\}$.

Let the acceleration measurement and the angular velocity measurements from the IMU be a_{IMU}, and $\Omega_{\mathrm{IMU}} \in \mathbb{R}^3$, which are treated as an exogenous time-varying signal resolved in

the b-frame. The acceleration measurement along the gravity is corrupted by an additive noise $w_a \in \mathbb{R}$ and a bias $b_a \in \mathbb{R}$, where the noise is Gaussian with $w_a \sim \mathcal{N}(0, Q_a)$ for a noise covariance $Q_a \in \mathbb{R}$, and the bias is driven by a Wiener process $W_{b_a} \in \mathbb{R}$ as

$$db_a = B_a dW_{b_a}, \tag{4}$$

for a scaling factor $B_a \in \mathbb{R}$. This implies that $b_a(\tau) - b_a(t)$ follows the Gaussian distribution of $\mathcal{N}(0, |\tau - t| B_a B_a^T)$ for any $t, \tau > 0$. We consider the accelerometer bias along the third component only, as it turned out that the biases along the other components are negligible in the experiments presented in Section V. However, it is readily generalized to include all of biases. Also, the angular velocity measurement is affected by an additive noise $w_\Omega \in \mathbb{R}^3$ following a Gaussian distribution $w_\Omega \sim \mathcal{N}(0, Q_\Omega)$ for $Q_\Omega \in \mathbb{R}^{3\times 3}$.

More specifically, the acceleration and the angular velocity in (2)-(3) are written in terms of the IMU measurements as

$$a = R R_{bi}(a_{\mathrm{IMU}} + w_a) + e_3 b_a + g e_3, \tag{5}$$
$$\Omega = R_{bi}(\Omega_{\mathrm{IMU}} + w_\Omega), \tag{6}$$

where $g \in \mathbb{R}$ is the gravitational acceleration and $e_3 = [0, 0, 1] \in \mathbb{R}^3$.

B. IMU Measurements

As discussed above, the measurements of the acceleration and the angular velocity from IMU are considered as a time-varying signal driving (1)–(3). In addition, IMU measures the orientation of the i-frame with respect to the f-frame, namely $R_{fi} = R R_{bi} \in \mathsf{SO(3)}$. The actual output of the IMU, namely $R_{\mathrm{IMU}} \in \mathsf{SO(3)}$

$$R_{\mathrm{IMU}} = R R_{bi} \exp \hat{\zeta}_R. \tag{7}$$

where $\exp : \mathfrak{so}(3) \to \mathsf{SO(3)}$ is the matrix exponential and $\zeta_R \in \mathbb{R}^3$ corresponds to the measurement noise for R_{IMU}, and it is assumed that $\zeta_R \sim \mathcal{N}(0, V_R)$ for a covariance matrix $V_R \in \mathbb{R}^{3\times 3}$.

C. RTK GPS Measurements

RTK GPS provides the measurements for the position x and the velocity v in the f-frame, corrupted by additive noise. Furthermore, the measurement is delayed by a fixed, known duration $t_D > 0$. More explicitly, the GPS measurements that become available at t are given by

$$x_{\mathrm{GPS}}(t) = x(t - t_D) + \zeta_x, \tag{8}$$
$$v_{\mathrm{GPS}}(t) = v(t - t_D) + \zeta_v, \tag{9}$$

where the additive measurement noise ζ_x and $\zeta_v \in \mathbb{R}^3$ are Gaussian with $\zeta_x \sim \mathcal{N}(0, V_x)$ and $\zeta_v \sim \mathcal{N}(0, V_v)$ for covariance matrices $V_x, V_v \in \mathbb{R}^{3\times 3}$.

D. Estimation Problem

Suppose that the initial position and the velocity satisfy

$$x(0) \sim \mathcal{N}(\bar{x}_0, P_{x_0}), \quad v(0) \sim \mathcal{N}(\bar{v}_0, P_{v_0}),$$

for the prescribed mean values and the covariances. Further, the initial attitude is a random matrix given by

$$R(0) = \bar{R}_0 \exp(\hat{\eta}_0),$$

where $\bar{R}_0 \in \mathsf{SO(3)}$ is the initial mean attitude, and $\eta_0 \in \mathcal{N}(0, P_{R_0})$ for $P_{R_0} \in \mathbb{R}^{3\times 3}$ represents the distribution of the initial attitude about its mean attitude. The initial bias is also distributed as $b_a(0) \sim \mathcal{N}(0, P_{b_0})$.

The objective is to construct the distribution of $(x(t), v(t), R(t), b_a(t))$ conditioned by the IMU attitude measurements $R_{\mathrm{IMU}}(\tau)$ and the GPS measurements $(x_{\mathrm{GPS}}(\tau), v_{\mathrm{GPS}}(\tau))$ available up to $\tau \le t$, for a given delay t_D and the history of other IMU measurements $(\Omega_{\mathrm{IMU}}(\tau), a_{\mathrm{IMU}}(\tau))$.

III. Extended Kalman Filter with Time-Delay

The proposed approach is based on the extended Kalman filter with Gaussian distributions, where the mean is propagated though the kinematics equation in the absence of the process noise and the covariance is propagated by its linearized form. The propagated distribution is corrected by the linearized measurement equation. As such, it is composed of the propagation step and the correction step. Two main challenges are integrating the delayed GPS measurements with the synchronous IMU measurements through the corrections steps, and dealing with the fact that the rotation matrix representing the attitude evolves on the compact Lie group. The former is tackled by formulating it as an optimal prediction, and the latter is addressed by representing the distribution in the tangent space of the special orthogonal group.

A. Uncertainty Distribution

Throughout this paper, the uncertainties in (x, v, R, b_a) are represented as follows. The mean values are denoted by $(\bar{x}, \bar{v}, \bar{R}, \bar{b}_a)$. Next, for the distribution about the mean, let

$$x = \bar{x} + \delta x,$$
$$v = \bar{v} + \delta v,$$
$$b_a = \bar{b}_a + \delta b_a,$$

for perturbations $(\bar{x}, \bar{v}, \bar{b}_a) \in \mathbb{R}^{3+3+1}$. For the attitude,

$$R = \bar{R} \exp(\hat{\eta}),$$

for $\eta \in \mathbb{R}^3$, which is uniquely determined when the angle between R and $\bar{R}$ is less than $180°$. We consider cases when the distribution of R is sufficiently concentrated that the probability that $\|\eta\| > \pi$ is not significant throughout this paper, which is common in attitude estimation. In particular, for highly concentrated cases, or $\|\eta\| \ll 1$, the above is approximated by

$$R = \bar{R}(I_{3\times 3} + \hat{\eta}) + \mathcal{O}(\|\eta\|^2),$$

which yields the following perturbation of the attitude

$$\delta R = R\hat{\eta}. \tag{10}$$

Define $\mathbf{x} \in \mathbb{R}^{10}$ be the concatenation of the perturbations:

$$\mathbf{x} = (\delta x, \delta v, \eta, \delta b_a).$$

And let $P \in \mathbb{R}^{10\times 10}$ be the covariance of $\mathbf{x}$ given by

$$P = \mathrm{E}[\mathbf{x}\mathbf{x}^T].$$

In short, the uncertainty distribution of the state and the bias is defined by the mean $(\bar{x}, \bar{v}, \bar{R}, \bar{b}_a)$, and the covariance P.

B. Prediction

Suppose that we have the distribution at $t = t_k$, given by $(\bar{x}_k, \bar{v}_k, \bar{R}_k, \bar{b}_{a_k})$ and P_k. The objective of prediction is to propagate those to the next time step t_{k+1} over (1)–(4).

The mean values $(\bar{x}, \bar{v}, \bar{R})$ are easily propagated by discretizing (1)–(3) in the absence of the process noise via the following second order explicit method:

$$\bar{x}_{k+1} = \bar{x}_k + h_k\bar{v}_k + \frac{h_k^2}{2}\bar{a}_k, \tag{11}$$

$$\bar{v}_{k+1} = \bar{v}_k + \frac{h_k}{2}(\bar{a}_k + \bar{a}_{k+1}), \tag{12}$$

$$\bar{R}_{k+1} = \bar{R}_k \exp\left\{\frac{h_k}{2}(\bar{\Omega}_k + \bar{\Omega}_{k+1})^\wedge\right\}, \tag{13}$$

$$\bar{b}_{a_{k+1}} = \bar{b}_{a_k}, \tag{14}$$

where $h_k = t_{k+1} - t_k$ is the discrete time step, and

$$\bar{\Omega}_k = R_{bi}\Omega_{\mathrm{IMU}_k}, \tag{15}$$

$$\bar{a}_k = \bar{R}_k R_{bi} a_{\mathrm{IMU}_k} + e_3\bar{b}_{a_k} + g e_3. \tag{16}$$

Next, to propagate the covariance P_k, we first study how the perturbation vector $\mathbf{x}$ evolves. This is achieved according to the extended Kalman filter, i.e., we linearize the equations of motion. Equation (1) and (4) are already linear. Substituting the perturbations into (2) and (5),

$$\begin{aligned}\delta\dot{v} &= R\hat{\eta}R_{bi}a_{\mathrm{IMU}} + RR_{bi}w_a \\ &= -R(R_{bi}a_{\mathrm{IMU}})^\wedge\eta + e_3\delta b_a + RR_{bi}w_a,\end{aligned} \tag{17}$$

as $\hat{x}y = -\hat{y}x$ for any $x, y \in \mathbb{R}^3$. Next, we find two equivalent expressions for $\delta\dot{R}$, by taking the time-derivative of (10) and by substituting (10) into (3) as

$$R\hat{\dot{\eta}} + R\hat{\Omega}\hat{\eta} = R\hat{\eta}\hat{\Omega}.$$

Left-multiplying R^T, and using the identity $\hat{x}\hat{y} - \hat{y}\hat{x} = (\hat{x}y)^\wedge$ for any $x, y \in \mathbb{R}^3$, we obtain $\dot{\eta} = -\hat{\Omega}\eta$. Substituting (6),

$$\dot{\eta} = -(R_{bi}\Omega_{\mathrm{IMU}})^\wedge\eta + R_{bi}w_\Omega, . \tag{18}$$

Together with (1) and (4), these can be rearranged into a matrix form as

$$d\mathbf{x} = (A(t)\mathbf{x} + F_1(t)\mathbf{w}_1)dt + F_2 dW_{b_a}, \tag{19}$$

where $\mathbf{w}_1 = [w_a; w_\Omega] \in \mathbb{R}^6$, and the matrices $A(t) \in \mathbb{R}^{10\times 10}$, $F_1(t) \in \mathbb{R}^{10\times 6}$, $F_2 \in \mathbb{R}^{10\times 1}$ are given by

$$A(t) = \begin{bmatrix} 0_{3\times3} & I_{3\times3} & 0_{3\times3} & 0_{3\times1} \\ 0_{3\times3} & 0_{3\times3} & -R(R_{bi}a_{\mathrm{IMU}})^\wedge & e_3 \\ 0_{3\times3} & 0_{3\times3} & -(R_{bi}\Omega_{\mathrm{IMU}})^\wedge & 0_{3\times1} \\ 0_{1\times3} & 0_{1\times3} & 0_{1\times3} & 0 \end{bmatrix},$$

$$F_1(t) = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ RR_{bi} & 0_{3\times3} \\ 0_{3\times3} & R_{bi} \\ 0_{1\times3} & 0_{1\times3} \end{bmatrix},$$

$$F_2 = \begin{bmatrix} 0_{9\times1} \\ B_a \end{bmatrix}.$$

The last term of (19) implies that an additive noise of $F_2 w_{b_k}$ is added over the period $[t_k, t_{k+1}]$, where $w_{b_k} \in \mathbb{R}$ follows $w_{b_k} \sim \mathcal{N}(0, h_k)$. Define $\mathbf{w}_k = [w_{a_k}; w_{\Omega_k}; w_{b_k}] \in \mathbb{R}^7$. It is straightforward to show $\mathbf{w}_k \sim \mathcal{N}(0, Q_k)$, where its covariance $Q_k \in \mathbb{R}^{7\times7}$ is

$$Q_k = \mathrm{diag}[Q_a, Q_\Omega, h_k]. \tag{20}$$

According to [17, p 330], the discrete-time linearized equation is given by

$$\mathbf{x}_{k+1} = A_k\mathbf{x}_k + F_k\mathbf{w}_k, \tag{21}$$

where $A_k \in \mathbb{R}^{10\times10}$ and $F_k \in \mathbb{R}^{10\times7}$ are

$$A_k = I_{10\times10} + h_k A(t_k)\Psi, \tag{22}$$

$$F_k = h_k\Psi[F_1(t_k), F_2], \tag{23}$$

with a matrix $\Psi \in \mathbb{R}^{10\times10}$

$$\begin{aligned}\Psi = I_{10\times10} + \frac{h_k}{2}A(t_k)\times \\ \left(I + \frac{h_k}{3}A(t_k)\left(I + \cdots\left(I + \frac{h_k}{N}A(t_k)\right)\right)\right),\end{aligned}$$

for an integer $N \geq 1$. Finally, from (21), the covariance is propagated as

$$P_{k+1} = A_k P_k A_k^T + F_k Q_k F_k^T. \tag{24}$$

In summary, the prediction step is completed by propagating the means with (11)–(14), and by propagating the covariance with (24). Note that the prediction step requires the IMU measurements $(\Omega_{\mathrm{IMU}}, a_{\mathrm{IMU}})$ at (15) and (16).

C. Correction with IMU Attitude Measurements

Next, we develop a correction step for the IMU attitude measurements presented in Section II-B. As the correction step is assumed to be executed instantaneously and there is no delay in the attitude measurement, we drop the subscript k indicating the time step throughout this subsection.

From (7), the expected attitude measurement is given by

$$\bar{R}_{\mathrm{IMU}} = \bar{R}R_{bi} \tag{25}$$

where $R_{bi} \in \mathsf{SO(3)}$ was a fixed rotation matrix describing how IMU is attached to the body-fixed frame. The discrepancy between above and the actual measurement (7) is referred to as a residual error, which is contributed by two errors: the attitude

estimation error represented as $R = \bar{R}\exp(\hat{\eta})$ for $\eta \in \mathbb{R}^3$, and the sensor measurement error given by (7) for $\zeta_R \in \mathbb{R}^3$. More explicitly,

$$R_{\mathrm{IMU}} = \bar{R}\exp\hat{\eta}R_{bi}\exp\hat{\zeta}_R. \tag{26}$$

As the attitude evolves on $\mathsf{SO(3)}$, the residual error does not appear in an additive form. Instead, we wish to represent it at the tangent space of $\mathsf{SO(3)}$ at $\bar{R}$ as

$$R_{\mathrm{IMU}} = \bar{R}\exp(\delta\hat{z})R_{bi}, \tag{27}$$

for a vector $\delta z \in \mathbb{R}^3$ considered as the residual error for the attitude measurement. As $\exp(\delta\hat{z}) = I_{3\times3} + \delta\hat{z} + \mathcal{O}(\|\delta z\|^2)$, we have

$$\delta z = (\bar{R}^T R_{\mathrm{IMU}} R_{bi}^T - I_{3\times3})^\vee,$$

up to the first order. In practice, the expression in the above parentheses may not be skew-symmetric. To avoid such cases, we project it to the space of skew-symmetric matrices as

$$\delta z = \frac{1}{2}(\bar{R}^T R_{\mathrm{IMU}} R_{bi}^T - R_{bi}R_{\mathrm{IMU}}^T\bar{R})^\vee. \tag{28}$$

Next, we find the relation between δz and η, ζ_R as follows. Using $\exp(\widehat{R\zeta}) = R\exp\hat{\zeta}R^T$ for any $R \in \mathsf{SO(3)}$ and $\zeta \in \mathbb{R}^3$, (26) is rewritten as

$$R_{\mathrm{IMU}} = \bar{R}\exp\hat{\eta}\exp(\widehat{R_{bi}\zeta_R})R_{bi},$$

which is compared with (27) to obtain

$$\exp(\delta\hat{z}) = \exp\hat{\eta}\exp(\widehat{R_{bi}\zeta_R}).$$

According to the BCH formula [18], after ignoring higher-order terms,

$$\delta z = \eta + R_{bi}\zeta_R,$$

which is rewritten as

$$\delta z = H\mathbf{x} + G\mathbf{v}, \tag{29}$$

where $\mathbf{v} = \zeta_R \in \mathbb{R}^{3\times3}$, and $H \in \mathbb{R}^{3\times10}$ and $G \in \mathbb{R}^{3\times3}$ are defined as

$$H = [0_{3\times3}, 0_{3\times3}, I_{3\times3}, 0_{3\times1}], \quad G = R_{bi}. \tag{30}$$

Now, the attitude measurement equation (29) is written in the usual form of the extended Kalman filter in $\mathbb{R}^3$. Therefore, the posterior mean of the perturbation and the posterior covariance are given by

$$\bar{\mathbf{x}}^+ = K\delta z, \tag{31}$$

$$P^+ = (I_{10\times10} - KH)P. \tag{32}$$

where $K \in \mathbb{R}^{10\times3}$ and $S \in \mathbb{R}^{3\times3}$ are

$$K = PH^TS^{-1}, \quad S = HPH^T + GV_RG^T.$$

Finally, we have to update the mean of the state, as the above correction is completed for the perturbation. The posterior mean of each state is

$$\bar{x}^+ = \bar{x} + \delta\bar{x}^+, \quad \bar{v}^+ = \bar{v} + \delta\bar{v}^+, \tag{33}$$

$$\bar{R}^+ = \bar{R}\exp(\hat{\bar{\eta}}^+), \quad \bar{b}_a^+ = \bar{b}_a + \delta\bar{b}_a^+. \tag{34}$$

In summary, once R_{IMU} becomes available, the residual δz is computed by (28), and the mean and the covariance are updated as (31)–(34).

D. Correction with Delayed GPS Measurements

As presented in Section II-C, the RTK GPS sensor considered in this paper provides measurements for the position and velocity with a known time delay t_D. Let the positive integer D be chosen such that $t_{k-D} = t_k - t_D$. The measurement equations (8)–(9) are rewritten as

$$z_{k-D} = H\mathbf{x}_{k-D} + \zeta, \tag{35}$$

where $z = [x; v] \in \mathbb{R}^6$, and $\xi = [\zeta_x; \zeta_v] \in \mathbb{R}^6$ with $\xi \sim \mathcal{N}(0_{6\times1}, V)$ for $V = \mathrm{diag}(V_x, V_v) \in \mathbb{R}^{6\times6}$. The matrix $H \in \mathbb{R}^{6\times10}$ is

$$H = \begin{bmatrix} I_{3\times3} & 0_{3\times3} & 0_{3\times4} \\ 0_{3\times3} & I_{3\times3} & 0_{3\times4} \end{bmatrix}, \tag{36}$$

which should be distinguished from (30). Due to the time delay, z_{k-D} becomes available at t_k.

For instance, the particular RTK GPS unit considered in Section V provides the measurement at 5 Hz delayed by $t_D = 0.4\,\mathrm{s}$, corresponding to two measurement cycles. When the time is discretized by the IMU frequency of 200 Hz, we have $D = 80$.

The objective of the GPS correction step is to construct $(x_k, v_k, R_k, b_{a_k})|z_{k-D}$, i.e., the current state conditioned by the position and velocity measurements in the past. More precisely, let $\mathcal{Z}_{k-D,k}$ be the set of measurements available until t_k measured for the states up to t_{k-D}, excluding the GPS measurement z_{k-D}. It includes the IMU measurements up to t_{k-D}, and all of the GPS measurements except z_{k-D}. Then, $(x_{k-D}, v_{k-D}, R_{k-D}, b_{a_{k-D}})|\mathcal{Z}_{k-D,k}$ represents the given prior knowledge of the state at t_{k-D}. The additional measurements became available between t_{k-D} and t_k is denoted by $\mathcal{W}_{k-D,k}$, which includes the IMU measurements between t_{k-D} and t_k, and the new GPS measurement z_{k-D}. The correction step is to construct $(x_k, v_k, R_k, b_{a_k})|\mathcal{Z}_{k-D,k}, \mathcal{W}_{k-D,k}$, i.e., the current state conditioned by the additional measurement $\mathcal{W}_{k-D,k}$. It is summarized into

$$\begin{gathered}\text{Given } (x_{k-D}, v_{k-D}, R_{k-D}, b_{a_{k-D}})|\mathcal{Z}_{k-D,k}, \text{and } \mathcal{W}_{k-D,k}, \\ \text{Find } (x_k, v_k, R_k, b_{a_k})|\mathcal{Z}_{k-D,k}, \mathcal{W}_{k-D,k}.\end{gathered}$$

This is essentially estimating the current state utilizing a past prior state estimate and a past measurement. This is referred to as optimal prediction [19], which is addressed by a sequence of correction at t_{k-D} and prediction to t_k. More explicitly, it can be solved by (i) correcting the prior state at t_{k-D} using the new coincidental GPS measurement z_{k-D}, and (ii) propagating it to t_k by repeating the IMU attitude correction step and the prediction step presented in the last two subsections.

As the procedure for the second step (ii) of the IMU correction and the prediction is identical to the preceding developments, here we present the coincidental GPS corrections for the step (i). Let the prior distribution $(x_{k-D}, v_{k-D}, R_{k-D}, b_{a_{k-D}})|\mathcal{Z}_{k-D,k}$ be defined by the mean $(\bar{x}_{k-D}, \bar{v}_{k-D}, \bar{R}_{k-D}, \bar{b}_{a_{k-D}})$ and the covariance P_{k-D} of the perturbation $\mathbf{x}_{k-D}$. Let the posterior distribution conditioned by z_{k-D}, namely

$(x_{k-D}, v_{k-D}, R_{k-D}, b_{a_{k-D}})|\mathcal{Z}_{k-D,k}, z_{k-D}$, be denoted by the superscript $+$.

From the measurement equation (35),

$$\bar{\mathbf{x}}^+_{k-D} = K(z_{k-D} - [\bar{x}_{k-D}; \bar{v}_{k-D}]), \tag{37}$$
$$P^+_{k-D} = (I_{10\times 10} - KH)P_{k-D}. \tag{38}$$

where $K \in \mathbb{R}^{10\times 6}$ is given by

$$K = P_{k-D}H^T S^{-1}, \quad S = HP_{k-D}H^T + V.$$

Given the corrected perturbed state $\bar{\mathbf{x}}^+$, the posterior mean of each state is updated by (33)–(34). After completing this, the posterior state at t_{k-D} is propagated to t_k using the saved IMU measurements.

In summary, the correction with the delayed GPS measurement is completed by correcting the state at t_{k-D} with (37)–(38), and propagating the posterior state up to t_k by repeating the prediction and the IMU correction. The proposed approach does not sacrifice the optimality in estimation. However, there are the memory requirements to save the IMU measurements over $[t_{k-D}, t_k]$ and the computation required for state propagation over the same period, which are intensified as the delay amount is increased relative to the IMU measurement frequency. This can be easily mitigated by sub-sampling IMU measurements over the prediction. The effects of the sub-sampling on the trade-off between the computational load and the accuracy will be discussed later. It turns out that a compact, mobile computing module available in these days can implement the proposed approach in real-time without affecting the accuracy noticeably.

IV. Numerical Examples

A. Simulation of the Proposed Method

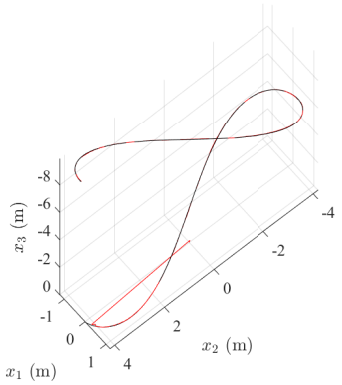
The proposed extended Kalman filter with time delayed position measurements has been implemented and verified through numerical simulations. For the particular RTK GPS unit used in the experimental results presented in the section, the relative position and the velocity are provided at 5 Hz with a delay of 0.4 s. Further, based on the actual sensor specifications, the IMU measurement frequency was chosen to be 200 Hz. These values are adopted for numerical simulation. For the delayed GPS corrections, there are $0.4 \times 200 = 80$ time steps for IMU correction and propagation, which are sub-sampled into 10 steps.

The measurement noise and the process noise are sampled according to

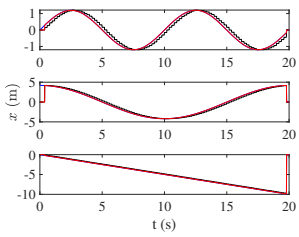
$$w_a \sim \mathcal{N}(0, 0.02^2 I_{3\times 3}), \quad w_\Omega \sim \mathcal{N}(0, 0.05^2 I_{3\times 3}),$$
$$\zeta_R \sim \mathcal{N}(0, 0.01^2 I_{3\times 3}), \quad \zeta_x, \zeta_v \sim \mathcal{N}(0, 0.01^2 I_{3\times 3}).$$

The simulated trajectory is a Lissajous curve with a continuously increasing altitude as shown in Figure 2-(a). Further, the body continuously rotates around its all three body-fixed axes with a time-varying angular velocity. More specifically, the simulated true position, velocity and body acceleration are given by

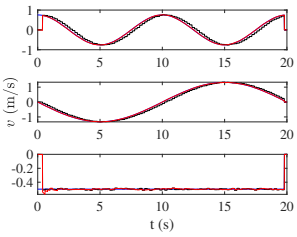
$$x(t) = [1.2\sin(0.2\pi t),\ 4.2\cos(0.1\pi t),\ -0.5t]^T,$$
$$v(t) = [1.2(0.2\pi)\cos(0.2\pi t),\ -4.2(0.2\pi)\sin(0.1\pi t),\ -0.5]^T,$$
$$a(t) = [-1.2(0.2\pi)^2\sin(0.2\pi t),\ -4.2(0.1\pi)^2\cos(0.1\pi t),\ 0]^T,$$



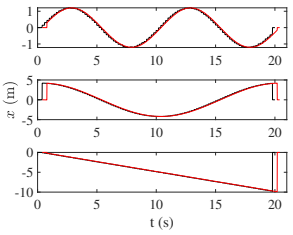
(a) Trajectory



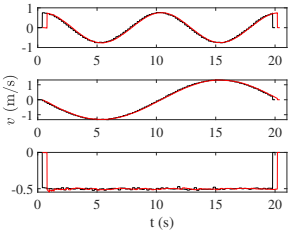
(b) Position, x



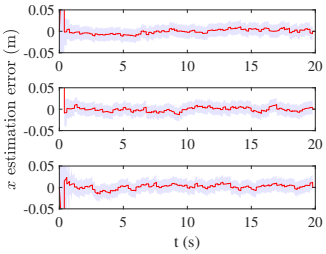
(c) Velocity, v



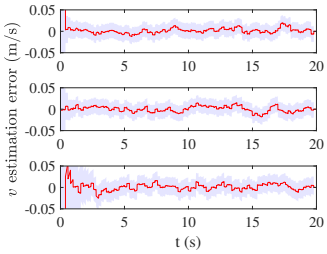
(d) Position with delayed corrected measurement



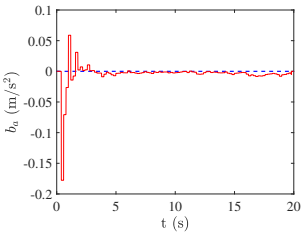
(e) Velocity with delayed corrected measurement



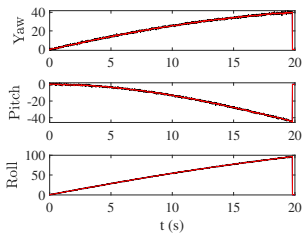
(f) Position estimation error



(g) Velocity estimation error



(h) Accelerometer bias, b_a



(i) Attitude, in degrees

Fig. 2. Simulation results for the estimator: blue dashed lines show the simulated true data, the red solid lines show the estimation from the proposed method, and solid black lines show the measurements. On plots (f) and (g), the error between the actual values and the estimates for the position and the velocity are shown. The light blue areas represent the estimated covariance each state.

and the simulated true attitude and the angular velocity are given by

$$R(t) = \begin{bmatrix} \cos t & -\cos t \sin t & \sin t^2 \\ \cos t \sin t & \cos t^3 - \sin t^2 & -\cos t \sin t - \cos t^2 \sin t \\ \sin t^2 & \cos t \sin t + \cos t^2 \sin t & \cos t^2 - \cos t \sin t^2 \end{bmatrix},$$

$$\Omega(t) = \begin{bmatrix} \cos t + 1 \\ \sin t - \sin(2t)/2 \\ \cos t - \cos t^2 + 1 \end{bmatrix}.$$

Further, a bias of $1.5\,\mathrm{m\,s^{-2}}$ was added to the acceleration measurement along the direction of gravity.

Figure 2 shows the results of the proposed estimator. For the numerical simulation, it is considered that the quadrotor follows the above trajectories exactly. Based on the true data, noisy measurements were generated and the proposed estimator was executed with those measurements. The blue, dashed lines represent the simulated true data, while black solid lines show sensor measurements where available. The red, solid lines show estimated states by the proposed filter with the forward propagation step.

As RTK GPS measurements are received at $5\,\mathrm{Hz}$, the black lines appear to be piece-wise constant, and they are shifted toward right compared to the actual states (see Figure 2-(b)-(c)) due to this delay. When new RTK GPS data are available, the delayed GPS correction and the forward propagation are executed as described in Section III-D. Further, for comparison, the estimated position and velocity are plotted with the GPS measurements corrected for the delay in Figure 2-(d),(e). These illustrate that the state estimated by the proposed method converge to actual state at the presence of non-zero time delay in the measurements.

Next, the error between the actual state and the estimated state for the position and the velocity are presented in Figure 2-(f),(g). The light blue areas represent the 3-sigma line of each state, extracted from the covariance matrix, P. It can be observed that the errors in the estimated position and the velocity stay lower than $0.03\,\mathrm{m}$ and $0.05\,\mathrm{m\,s^{-1}}$, respectively. Further, as shown in Figure 2-(h), the accelerometer bias estimated by the delayed position correction converges to the actual value after the initial delay period.

In addition to the position estimation, the attitude correction (31)–(34) was performed when new IMU data are available. This estimated attitude is shown in Figure 2-(i). It should be noted that the estimator was executed with the attitude defined in the configuration space $\mathsf{SO}(3)$, as in (3), and the attitude matrix was converted into yaw-pitch-roll angles when generating the figure. It can be observed that the estimated attitude converges to the actual attitude within the first few milliseconds.

B. Effects of Sub-Sampling

Next, we study the effects of sub-sampling in the delayed GPS correction. Two additional cases are considered: the ideal case when there is no delay in the GPS measurements, and another implementation of the proposed method for the delayed GPS measurements where IMU is not sub-sampled. These are implemented using Matlab R2019b on a computer with an AMD Ryzen 5 1600 $3.2\,\mathrm{GHz}$ CPU. Each method is executed 1000 times to compute the average values of the error and the computation time.

Table I presents a comparison of error and computational time of the proposed method against these non-delayed case and the delayed case without sub-sampling. In the table, the column of average IMU loop time includes the computational time for the non-delayed measurements related calculations, namely the prediction and the IMU correction. The computational time for the delayed GPS correction and propagation (if any) is included in the average GPS loop time column. The total loop time column includes the total computation time of all the loops, divided by the number of loops. All the percentage values in this table are relative to the delayed case without sub-sampling.

The process of sub-sampling simply reduces the number of sequential computations required when the delayed measurement arrives, thereby reducing the computational time for GPS correction. Comparing the proposed method with sub-sampling and one without sub-sampling, it is shown that sub-sampling decreased the GPS loop time into 32%, while increasing the position error by only 1%, thereby justifying sub-sampling. Compared with the ideal case without delay, the proposed method with sub-sampling increases the error by 3% and the computation time by 5%, and as such, the adverse effects of time-delay are well mitigated.

C. Benchmark Study

In addition, the performances of the proposed method are compared against three existing methods ([13], [14], [15]) that deal with delayed measurements.

First, [14] and [15] are applied to the above case, and the corresponding results are presented at the last row of Table I. It is shown that the computation time of [14] is comparable to the proposed method with sub-sampling, but the position estimation error is greater than the proposed method by about 35%. The results of the method proposed in [15] has a 5% lower error compared to the results with sub-sampling. However, this improvement comes with the substantial computational load increased by 423%. This is mainly caused by the time required to compute the square-root of the covariance matrix for calculating sigma points. Computing the matrix square roots alone takes a time comparable to the total loop time of any the other technique compared in this benchmark study. [1]

Next, as the technique presented in [13] does not deal with the case where only parts of measurements are delayed, it cannot be directly applied to the presented case to integrate IMU with delayed GPS measurements. For a fair comparison with [13], we consider an additional case where the position dynamics are decoupled from the attitude dynamics. More explicitly, for the position estimation, the state is reduced to $\mathbf{x}^p = [x_k; v_k; b_{a_k}] \in \mathbb{R}^7$ with the corresponding covariance matrix $P^p \in \mathbb{R}^{7\times 7}$, for which [13], [14], and the proposed method are applied. The corresponding results are summarized

[1]The codes used for these comparisons can be found at https://github.com/fdcl-gwu/dkf-comparison.git

TABLE I
COMPARISON OF ESTIMATION ERROR AND COMPUTATIONAL TIME FOR THE PROPOSED METHOD

Method	Average position error		Average loop time					
			Prediction + IMU Correction		GPS Correction		Total	
	m	%	ms	%	ms	%	ms	%
Non-delayed	0.0875	98	0.0403	102	0.0875	11	0.1044	76
Proposed without sub-sampling	0.0892	100	0.0395	100	0.7699	100	0.1366	100
Proposed with sub-sampling	0.0897	101	0.0398	101	0.2453	32	0.1112	81
Larsen's method [14]	0.1214	136	0.0402	102	0.1327	17	0.1108	81
Sigma-Point KF [15]	0.0844	95	0.5622	1424	0.2398	31	0.5771	423

TABLE II
BENCHMARK STUDY WITH OTHER TECHNIQUES

Method	Average position error		Average loop time					
			Prediction + IMU Correction		GPS Correction		Total	
	m	%	ms	%	ms	%	ms	%
Non-delayed	0.0959	100	0.0619	100	0.0461	9	0.0644	75
Proposed without sub-sampling	0.0961	100	0.0616	100	0.4899	100	0.0860	100
Proposed with sub-sampling	0.0975	101	0.0617	100	0.1463	30	0.0691	80
Alexander's method [13]	0.1441	150	0.0622	101	0.0783	16	0.0663	77
Larsen's method [14]	0.1442	150	0.0621	101	0.0512	10	0.0648	75

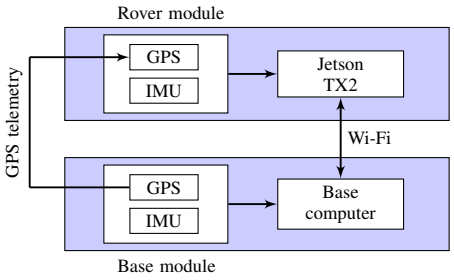


Fig. 3. Schematic of the hardware setup: GPS telemetry sends corrections to the rover RTK GPS module, sending commands and receiving data to/from the the computing module onboard the rover module performed through a Wi-Fi connection

in Table II. All the percentage values presented are compared with the proposed method without sub-sampling.

It can be observed that the position error resulting from [13] or [14] is about 50% greater than either of the proposed approaches. The proposed approach without sub-sampling requires the greatest computation time, but the proposed approach with sub-sampling has a similar level of computation time with [13] and [14]. In short, the proposed approach with sub-sampling exhibits the best trade-off between the accuracy of estimation and the computational load.

V. FLIGHT EXPERIMENTS

The proposed estimator has been validated through several flight experiments. This section summarizes the hardware configuration and the software implementation of a quadrotor aerial vehicle and present the corresponding results.

A. Hardware Configuration

The experimental setup consists of two modules, namely, a base module and a rover module as shown in Figure 3. The rover module consists of a 9-axis VectorNav VN100 IMU sensor, a SwiftNav Piksi Multi RTK GPS sensor, and a FreeWave telemetry unit. An NVIDIA Jetson TX2 is chosen as the onboard computing module and it is attached on a Connect Tech Orbitty Carrier board. All the sensors are mounted on a custom designed printed circuit board which performs voltage regulation for each sensor, and provides an interface to communicate sensor data to the carrier board. Further, the Jetson TX2 is configured as a Wi-Fi hotspot so that it can directly communicate with a notebook connected to the base module. The rover module is powered by a $14.8\,\mathrm{V}$ Li-Po battery, and all the components are enclosed in a water-proof container for safe recovery in ocean environments.

The base module consists of the same sensors as the rover module, excluding the Jetson TX2 computing module. The base GPS sensor sends corrections to the rover RTK GPS module through the telemetry unit, while providing high accuracy (error standard deviation in the range of $1\,\mathrm{cm}$) relative position data with a latency of $0.4\,\mathrm{s}$ at a low frequency. The measurements from the sensors in the base module are transmitted to a notebook connected through USB. Further, the base computer is connected to the Wi-Fi hotspot on the rover for data log. The estimator is executed real-time on the Jetson TX2 on the rover module, and the states estimated by the proposed estimator and rover sensor data are transmitted to the base computer over Wi-Fi for real-time data monitoring.

B. Software Implementation

All of estimation, communication, and data log are implemented using a multi-threaded C++ program. A separate thread is developed for each of the following tasks: reading from each sensor (IMU and GPS), communicating with the base computer for receiving commands and sending data for real-time data monitoring, and generating control inputs for the UAV. When new measurements are available, the corresponding thread runs the required prediction and the correction steps.

The prediction and the attitude correction using IMU measurements were run in real-time. However, the proposed forward propagation for the delayed GPS correction requires the

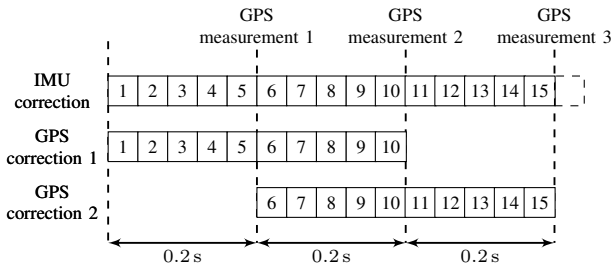


Fig. 4. Schematic of the data saving structure: Each block represents the data saved between the delayed GPS measurements. The time period between each GPS measurement is $0.2\,\mathrm{s}$. The measurement delay of $0.4\,\mathrm{s}$ is spread over two measurement cycles.

IMU data, the estimated attitude, and the covariance over the delayed time period. Since IMU runs at $200\,\mathrm{Hz}$, saving all the corresponding IMU data for the delay of $0.4\,\mathrm{s}$, and using them to perform the forward propagation are not feasible for real-time implementation. Instead, every eighth IMU data is saved so that approximately five IMU data points are available between each GPS measurements update. In other words, the IMU measurements are sub-sampled to $25\,\mathrm{Hz}$ over the propagation required for GPS corrections.

Further, the GPS sensor runs at $5\,\mathrm{Hz}$, which means that the time period between two measurements is $0.2\,\mathrm{s}$. The sensor has approximately a $0.4\,\mathrm{s}$ delay, which is translated into a delay spread over two GPS measurement cycles as illustrated in Figure 4.

Here, each block in the IMU correction line represents the five data points saved during the IMU corrections between two GPS measurements. Let the start of the estimator denoted by the block 1. The first GPS measurement is acquired at the fifth step. But, this measurement is disregarded as it corresponds to a state before the estimator started running.

The second GPS measurement is acquired at the 10-th instance, and this corresponds to the actual state $0.4\,\mathrm{s}$ prior at the first block. This measurement is used to correct the state at the block 1, the corrected state is propagated to the current time using the data saved in blocks 1 to 10.

Then, the third GPS measurement is acquired at 15-th instance, and the same calculations performed for the second GPS measurement must be repeated using the data saved at blocks 6 to 15. However, the states in blocks 6 to 10 are already updated using the second GPS measurement, and those corrected states must be used for correction with the third GPS measurement. In other words, when the delayed GPS correction was run with the second GPS measurement, the updates to the states at blocks 6 to 10 must be saved and re-used for the GPS correction with the third GPS measurement.

Taking this into consideration, the proposed estimator is implemented in C++ utilizing standard vectors that can dynamically allocate its size, which is helpful when there are unexpected additional delays or drops in the GPS measurement so that the corrected state should be propagated more than $0.4\,\mathrm{s}$. The pseudocode for the implementation is presented in Algorithm 1. The procedure "IMU Callback" includes running the attitude prediction and correction, and position

Algorithm 1 Delayed GPS Correction

1: **procedure** IMUCALLBACK($R_{\mathrm{IMU}}, a_{\mathrm{IMU}}, \Omega_{\mathrm{IMU}}$)
2: $x, v, b_a, R,\ P \leftarrow$ prediction($a_{\mathrm{IMU}}, \Omega_{\mathrm{IMU}}$)
3: $R,\ P \leftarrow$ attitude_correction(R_{IMU})
4: $counter \leftarrow counter + 1$
5: $r \leftarrow counter \bmod 8$
6: **if** $r == 0$ **then**
7: $q \leftarrow x, v, R, b_a, P, a_{\mathrm{IMU}}, \Omega_{\mathrm{IMU}}$
8: Q.enqueue(q)
9: $n_{\mathrm{IMU}} \leftarrow n_{\mathrm{IMU}} + 1$
10: $counter \leftarrow 0$
11: **return**
1: **procedure** GPSCALLBACK($x_{\mathrm{GPS}}, v_{\mathrm{GPS}}$)
2: $x_D, v_D, R_D, b_{a_D}, P_D \leftarrow Q$.dequeue()
3: $x_D, v_D, b_{a_D}, P_D \leftarrow$ position_correction($x_{\mathrm{GPS}}, v_{\mathrm{GPS}}$)
4: $n \leftarrow length(Q)$
5: **for** $i \leftarrow 1, n$ **do**
6: $a_{\mathrm{IMU}_D} \leftarrow Q$.dequeue()
7: $x_D, v_D, b_{a_D}, P_D \leftarrow$ forward_propagation(a_{IMU_D})
8: **if** $i > n - n_{\mathrm{IMU}}$ **then**
9: $q \leftarrow x_D, v_D, b_{a_D}, R_D, P_D, a_{\mathrm{IMU}_D}, \Omega_{\mathrm{IMU}_D}$
10: Q.enqueue(q)
11: $x, v, b_a, P \leftarrow x_D, v_D, b_{a_D}, P_D$
12: $n_{\mathrm{IMU}} \leftarrow 0$
13: **return**

(a) Ship (b) Octocopter

Fig. 5. Experimental setup in ocean environment: (a) USNA YP700 as seen from a camera attached to the UAV during a flight test in Chesapeake Bay (b) octocopter attached with the rover module hovering over the ship deck of YP700

prediction, while saving data required for the delayed position correction. The procedure "GPS Callback" implements the delayed correction and forward propagation. Further, lines 8-10 in this procedure are used for saving data of the latest measurement cycle for the next position correction.

C. Flight Experiments in Ocean Environments

The proposed estimator was tested in outdoor flight experiments in an ocean environment. The base module was fixed on a United States Naval Academy (USNA) research vessel YP700, and the rover module was attached to a custom made research UAV (see Figure 5). The UAV was controlled manually behind the ship by a human RC pilot, while the ship was sailing at a constant velocity in the Chesapeake Bay. The pilot utilized the attitude hold mode of another on-board flight control system that does not require any position measurement. This system is independent of the NVIDIA Jetson TX2 that

executed the proposed estimation scheme. The objective of the experiment is to observe how the proposed estimator performs under strong wind conditions that has an adverse effect on the acceleration measurements. The data collected in this experiment are presented in Figure 6, where black lines show the sensor measurements, and the red lines represent the states estimated by the proposed estimator.

The trajectory of the UAV with respect to the ship is shown in Figure 6-(a). In this plot, $(0, 0)$ corresponds to the position where the base GPS antenna was mounted. The negative x_1 direction is toward the bow of the ship from the GPS antenna, and the positive x_2 direction is the port side of the ship. The UAV took off from the ship deck in the aft of the ship, and it was flying back and forth along positive x_1 direction until $450\,\mathrm{s}$ such that the trajectory remains inside of the turbulent ship air wake. After that, the UAV flew back and forth while also moving towards the port side.

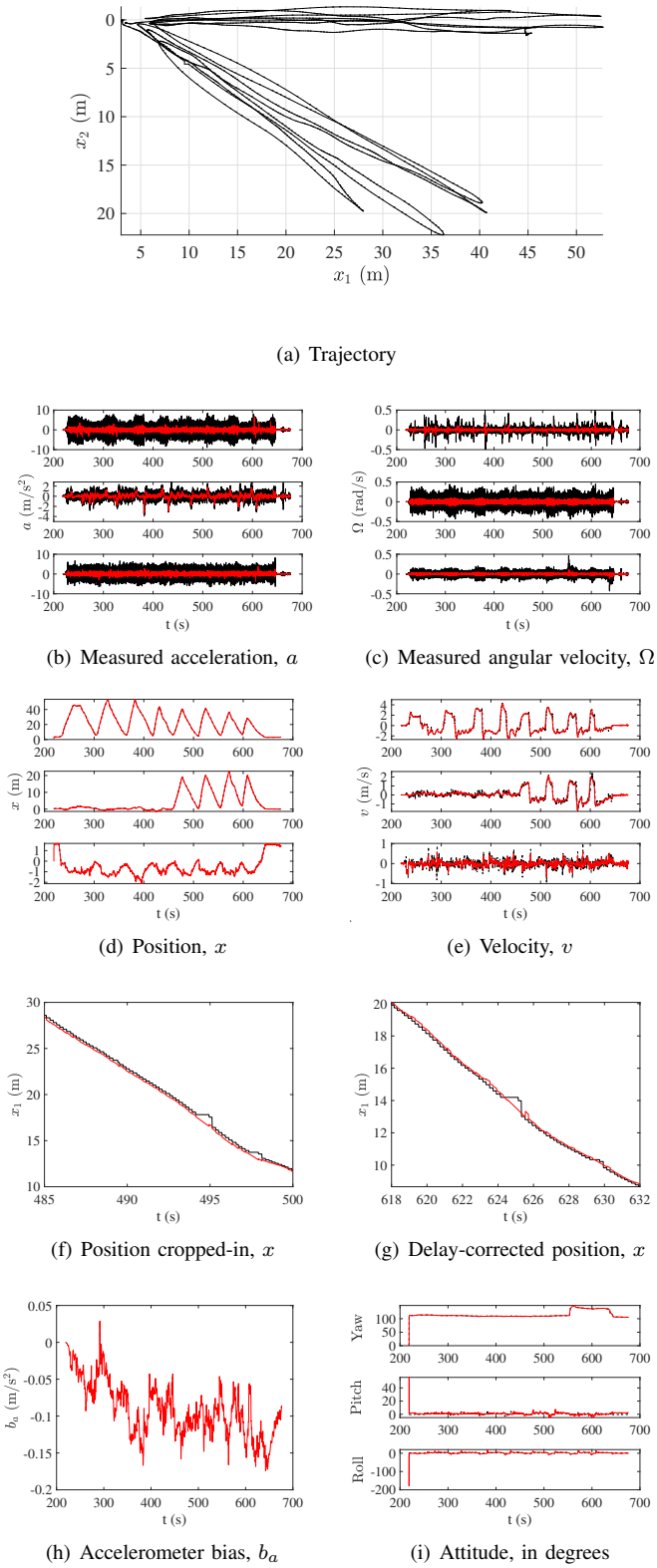
The 3-axis acceleration and 3-axis angular velocity measured by IMU are plotted in Figure 6-(b), (c). Due to the vibration from the motors and the effects of the turbulence from the ship air wake, these measurements are corrupted by high frequency noise. To avoid that, both the measurements are first filtered through low-pass Butterworth filters. These filtered measurements (shown as red lines) are then used in the proposed estimator.

It should be noted that filtering IMU measurements with Butterworth filters causes a delay. However, with the vibrations incurred by the motors/propellers, it is difficult to operate the UAV with unfiltered IMU measurements without performance degradation even with high accurate position measurements provided by motion capture systems. Therefore, we tuned filter parameters to achieve the best compromise between the maximum vibration rejection and minimal lag, determined with extensive hardware experiments [20]. The corresponding delay caused by filtering is in the level of 20 milliseconds, and it does not have any significant effect on the state estimation.

The estimated position and the velocity using the proposed filter are illustrated in Figure 6-(d),(e). Since it is difficult to visualize the estimated states given the axis scales, Figure 6-(f) shows a zoomed in view of the position plot along x_1-axis between $485\,\mathrm{s}$ and $500\,\mathrm{s}$. The selected region also includes two instances where GPS measurements were dropped due to transmission losses at approximately $494\,\mathrm{s}$ and $497\,\mathrm{s}$. Even at the presence of these additional delays in the GPS measurements, the proposed estimator performed well.

Additionally, a cropped in view of the position plot along x_2-axis between $618\,\mathrm{s}$ and $632\,\mathrm{s}$ is shown in Figure 6-(g). This also includes two instances where the GPS measurements had additional delays approximately at $624\,\mathrm{s}$ and $629\,\mathrm{s}$. Here, the position estimated by the proposed method is plotted with the GPS measurements corrected for the time delay, which moves both of them to the same time horizon. It can be observed that the estimated position follows the measurement as if there were no delay.

Further, the estimated accelerometer bias is shown in Figure 6-(h), and the attitude estimated by the proposed method is shown in Figure 6-(i). The yaw-pitch-roll angles were calculated during the post processing, from the estimated attitudes



(a) Trajectory

(b) Measured acceleration, a

(c) Measured angular velocity, Ω

(d) Position, x

(e) Velocity, v

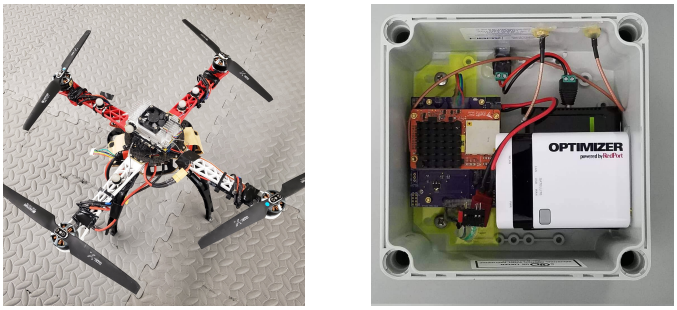
(f) Position cropped-in, x

(g) Delay-corrected position, x

(h) Accelerometer bias, b_a

(i) Attitude, in degrees

Fig. 6. Results for flight experiments in ocean environment: red solid lines show the estimation from the proposed method, black lines show the measurements.



(a) (b)

Fig. 7. Outdoor autonomous flight hardware: (a) the custom built quadrotor at the Flight Dynamics and Control Lab at GWU (b) components of the base module

Fig. 8. The quadrotor UAV during an autonomous flight, using the states estimated by the proposed estimator, inside the GWU outdoor UAV testing area.

represented by rotation matrices. It shows the convergence of the attitude within few milliseconds.

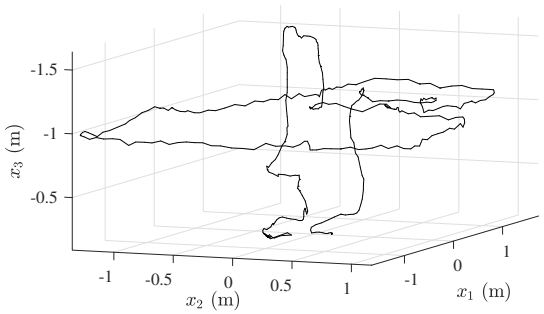
D. Autonomous Flight

Next, outdoor autonomous flight tests were performed utilizing the proposed estimator. [2] Another quadrotor UAV built at the Flight Dynamics and Control Lab at GWU (see Figure 7-(a)) was used as the testing platform. The rover module with the Jetson TX2 computing module was attached to the UAV. In addition to the IMU and the RTK GPS sensors, the Jetson TX2 at the rover module is connected to MikroKopter BL-Ctrl v2 ESC units over I2C, to control four 700 kV T-Motor brushless DC motors with 10×4.7 propellers.

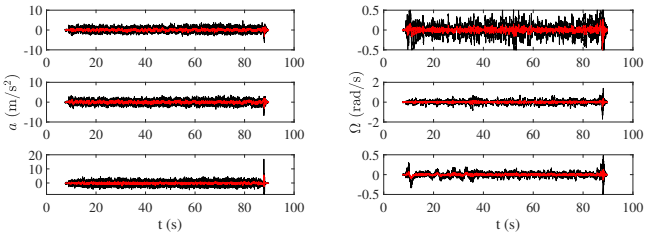
The states estimated by the proposed method were used to control the UAV using the geometric controller proposed in [21]. The measurement error covariance estimated by the sensors are used as the sensor noise values in the estimator. Both the estimation and the control were performed onboard in real-time through the Jetson TX2 computing module running Ubuntu 16.04 with JetPack 3.3.

The data saved during a flight test are presented in Figure 9. The red solid lines represent the estimated data, black lines show the sensor measurements, and the blue lines show the desired commands used for the autonomous navigation. The

[2]A video of one of the flight experiments can be found at https://youtu.be/PfuGb5yhlLQ.

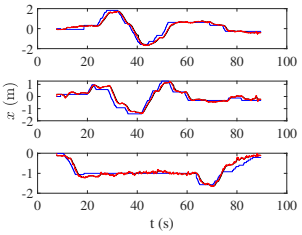


(a) Trajectory

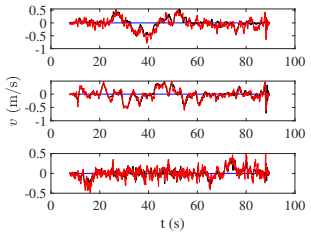


(b) Measured acceleration, a

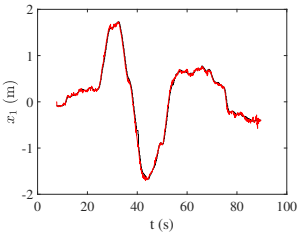
(c) Measured angular velocity, Ω



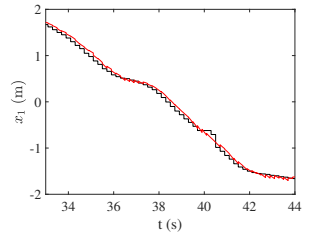
(d) Position, x



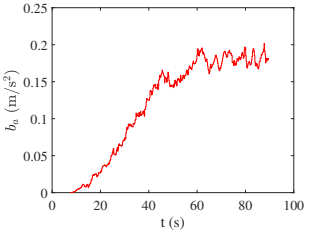
(e) Velocity, v



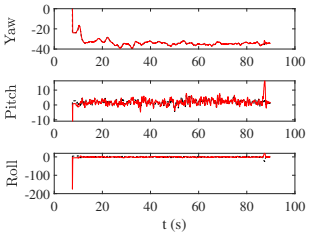
(f) Delay corrected position, x



(g) Cropped in delay corrected position, x



(h) Accelerometer bias, b_a



(i) Attitude, in degrees

Fig. 9. Results for autonomous flight: red solid lines show the estimation from the proposed method, black lines show the measurements, blue lines show the desired commands

desired position commands are generated by key strokes at the base computer, which represent a shift of the hovering position by $0.1\,\mathrm{m}$ along each axis of the inertial frame. These are transmitted to the UAV over Wi-Fi. The trajectory of the flight from the takeoff to the landing is shown in Figure 9-(a). The UAV was first set to take-off to a given height, then it was moved making an approximate rectangle, followed by the command to land.

Similar with the previous test, both the acceleration and the angular velocity measurements are filtered through separate low-pass Butterworth filters. The black lines in Figure 9-(b),(c) show the actual sensor measurement, and the red lines show the filtered measurements, which were used in the estimator.

The position and the velocity estimated are shown in Figure 9-(d), (e). The estimated states closely follow the measured states. Further, the UAV follows the position command sent to the UAV through the base computer shown as the blue solid line.

For better visualization, the position along x_1 axis is plotted separately in Figure 9-(f). The black solid line is the delay-compensated RTK GPS measurement, where the red line is the estimated position. It can be observed that the estimated position closely follows the GPS measurement.

Further, Figure 9-(g) shows the same plot as in Figure 9-(f), but cropped in between $33\,\mathrm{s}$ and $44\,\mathrm{s}$ and the GPS measurements corrected for the delay. This section is chosen since it includes an instance where the GPS measurements has an additional delay at approximately $40\,\mathrm{s}$. Even in the presence of this additional delay, the estimation did not diverge, and did not adversely affect the performance of the autonomous flight.

The estimated accelerometer bias is shown in Figure 9-(h), and attitude during the flight is presented in Figure 9-(i). The IMU attitude measurements are high frequency non-delayed measurements, and it can be observed that the estimates converge to the measured values during the first few milliseconds.

Quadrotor is not inherently stable, and as such, the availability of precise real-time position estimation at a higher rate is critical for autonomous flight. The proposed estimation scheme achieves it successfully with a low-cost RTK GPS unit that has a nontrivial latency, even in the presence of occasional GPS measurement drops. Through multiple flight tests, we experienced that the quadrotor could not fly in a stable manner or it even crashed without the proposed estimator compensating the delay.

VI. Conclusions

This paper presents an estimation scheme to integrate IMU measurements and delayed RTK GPS measurements. Assuming that the delay period is given, the prior state coincidental to the delayed GPS measurements is corrected, before being propagated to the current time with the saved IMU measurements. In particular, the proposed method with subsampling IMU measurements yields an excellent performance comparable to the ideal case without delay, without requiring excessive computational load. The efficacy of the proposed scheme has been verified with benchmark numerical studies and successful outdoor flight experiments. Future directions include considering unknown time-delay via adaptive filtering approaches.

References

[1] B. Townsend and P. Fenton, "A practical approach to the reduction of pseudorange multipath errors in a L1 GPS receiver," in *Proceedings of the 7th International Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, UT, USA*. Citeseer, 1994, pp. 20–23.

[2] *Piksi Multi Product Summary v1.2.2*, Swift Navigation, 2020. [Online]. Available: https://www.swiftnav.com/resource-files/Piksi%20Multi/v1.2.2/Specification/Piksi%20Multi%20Product%20Summary%20v1.2.2.pdf

[3] M. S. Grewal, L. R. Weill, and A. P. Andrews, *Global positioning systems, inertial navigation, and integration*. John Wiley & Sons, 2007.

[4] E. Nebot, S. Sukkarieh, and H. Durrant-Whyte, "Inertial navigation aided with GPS information," in *Proceedings Fourth Annual Conference on Mechatronics and Machine Vision in Practice*. IEEE, 1997, pp. 169–174.

[5] J. Wendel and G. F. Trommer, "Tightly coupled GPS/INS integration for missile applications," *Aerospace Science and Technology*, vol. 8, no. 7, pp. 627–634, 2004.

[6] F. Caron, E. Duflos, D. Pomorski, and P. Vanheeghe, "GPS/IMU data fusion using multisensor Kalman filtering: introduction of contextual aspects," *Information fusion*, vol. 7, no. 2, pp. 221–230, 2006.

[7] A. M. Hasan, K. Samsudin, A. R. Ramli, R. Azmir, and S. Ismaeel, "A review of navigation systems (integration and algorithms)," *Australian journal of basic and applied sciences*, vol. 3, no. 2, pp. 943–959, 2009.

[8] Y. Li, J. Wang, C. Rizos, P. Mumford, and W. Ding, "Low-cost tightly coupled GPS/INS integration based on a nonlinear Kalman filtering design," in *Proceedings of ION National Technical Meeting*, 2006, pp. 18–20.

[9] M. George and S. Sukkarieh, "Tightly coupled INS/GPS with bias estimation for UAV applications," in *Proceedings of Australiasian Conference on Robotics and Automation (ACRA)*, 2005.

[10] R. Hirokawa and T. Ebinuma, "A low-cost tightly coupled GPS/INS for small UAVs augmented with multiple GPS antennas," *Navigation*, vol. 56, no. 1, pp. 35–44, 2009.

[11] M. B. Rhudy, Y. Gu, and M. Napolitano, "Low-cost loosely-coupled dual GPS/INS for attitude estimation with application to a small UAV," in *AIAA Guidance, Navigation, and Control (GNC) Conference*, 2013, p. 4957.

[12] S. C. Thomopoulos, "Decentralized filtering and control in the presence of delays: Discrete-time and continuous-time case," *Information Sciences*, vol. 81, no. 1-2, pp. 133–153, November 1994.

[13] H. L. Alexander, "State estimation for distributed systems with sensing delay," in *Data Structures and Target Classification*, V. Libby, Ed. SPIE, August 1991.

[14] T. Larsen, N. Andersen, O. Ravn, and N. Poulsen, "Incorporation of time delayed measurements in a discrete-time Kalman filter," in *Proceedings of the 37th IEEE Conference on Decision and Control (Cat. No.98CH36171)*. IEEE, 1998.

[15] R. van der Merwe, E. Wan, and S. Julier, "Sigma-point Kalman filters for nonlinear estimation and sensor-fusion: Applications to integrated navigation," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, August 2004.

[16] S. Julier and J. Uhlmann, "Fusion of time delayed measurements with uncertain time delays," in *Proceedings of the 2005, American Control Conference, 2005.* IEEE, 2005.

[17] W. L. Brogan, *Modern control theory*. Prentice Hall, 1991.

[18] E. Hairer, C. Lubich, and G. Wanner, *Geometric numerical integration*, ser. Springer Series in Computational Mechanics 31. Springer, 2000.

[19] A. Bryson and Y. Ho, *Applied Optimal Control*. Hemisphere Publishing Corporation, 1975.

[20] M. Bisheban, "Geometric control of a quadrotor unmanned aerial vehicle in wind field (PhD thesis)," 2018.

[21] K. Gamagedara, M. Bisheban, E. Kaufman, and T. Lee, "Geometric controls of a quadrotor UAV with decoupled yaw control," in *2019 American Control Conference (ACC)*, July 2019, pp. 3285–3290.

Kanishke Gamagedara is currently pursuing his Ph.D. degree at the George Washington University, where he also received his master's degree in Mechanical and Aerospace Engineering in 2018. He received his bachelor's degree at University of Peradeniya, Sri Lanka, in Mechanical Engineering in 2015. His research interests are mainly focused on geometric control of unmanned aerial systems.

Taeyoung Lee is a professor of the Department of Mechanical and Aerospace Engineering at the George Washington University. He received his doctoral degree in Aerospace Engineering and his master's degree in Mathematics at the University of Michigan in 2008. His research interests include geometric mechanics and control with applications to complex aerospace systems.

Murray Snyder Biography text here.

LIST OF FIGURES