A video presentation of this paper can be found at https://www.youtube.com/watch?v=s3eF84ucsdw

Using multiple short hops for multicopter navigation with only inertial sensors

Xiangyu Wu and Mark W. Mueller

***Abstract*— In certain challenging environments, such as inside buildings on fire, the main sensors (e.g. cameras, LiDARs and GPS systems) used for multicopter localization can become unavailable. Direct integration of the inertial navigation sensors (the accelerometer and rate gyroscope), is however unaffected by external disturbances, but the rapid error accumulation quickly makes a naive application of such a strategy feasible only for very short durations. In this work we propose a motion strategy for reducing the inertial navigation state estimation error of multicopters. The proposed strategy breaks a long duration flight into multiple short duration hops between which the vehicle remains stationary on the ground. When the vehicle is stationary, zero-velocity pseudo-measurements are introduced to an extended Kalman Filter to reduce the state estimation error. We perform experiments for closed-loop control of a multicopter for evaluation. The mean absolute position estimation error was 3.4% over a total flight distance of 5m in the experiments. The results showed a 80% reduction compared to the standard inertial navigation method without using this strategy. In addition, an additional experiment with total flight distance of 10m is conducted to demonstrate the ability of this method to navigate a multicopter in real-world environment. The final trajectory tracking error was 3% of the total flight distance.**

I. Introduction

Reliable and cost effective state estimation methods are critical for the operation of unmanned aerial vehicles (UAV). In laboratories, motion capture systems are often used to obtain very accurate state estimation of multicopters at high frequency [1]. While these systems are ideal for indoor research use, they are usually very expensive, inconvenient to set up and can only cover a small area in the order of several square meters [2]. For operations in open areas, Global Navigation Satellite System (GNSS) is widely used and can achieve localization accuracy of a few meters. In places where GNSS signal is weak or unavailable (e.g. indoor, underground or near tall buildings), radio beacons can be used to setup a localization networks [3] [4]. Such radio based localization systems can be created at a relatively low cost but the reliance on infrastructure makes them less flexible compared to only relying on on-board sensors.

Another category of UAV state estimation methods only relies on sensors on the vehicle itself such as cameras, LiDAR and Inertial Measurement Units (IMUs). One popular method in this category is simultaneous localization and mapping (SLAM) where the measurements from on-board sensors are often fused to build a map of the vehicle's surrounding environment and find the vehicle's location in the map [5] [6] [7]. Not dependent on any particular infrastructure, these methods are easy to deploy. In the recent past there have been significant progress in this area which enables them to transit into real-world applications [7]. On the other hand, these methods often require expensive sensors such as LiDAR and powerful computers and it is challenging to make them work robustly in challenging (e.g. featureless or dusty) environments.

The authors are with the High Performance Robotics Lab, Mechanical Engineering Department, University of California, Berkeley, CA 94703, USA. {wuxiangyu,mwm}@berkeley.edu

Fig. 1. Video sequences of a quadcopter doing a short hop flight. The state estimation error of inertial navigation can be reduced significantly by breaking a long time flight into multiple short time hop flights.

Inertial navigation is a potential solution under these challenging environments since the only sensor it requires is the IMU, which is usually unaffected by environments. Inertial state estimation has many applications in robotics state estimation. For example, [8] used inertial navigation for wheeled robots. By utilizing the information that the wheeled vehicle's lateral and vertical velocities are roughly zero in body frame and using a Convolutional Neural Network (CNN) for the IMU noise estimation, the method was able to achieve a position estimation error comparable to methods of using LiDAR or stereo vision. In addition, [9] [10] used inertial measurements with the aerodynamic modeling of the vehicle for the velocity and attitude estimation of multicopters.

A major challenge of inertial navigation is that consumer-level IMUs have large measurement noise which will result in fast error accumulation [11]. One technique to reduce inertial navigation error is detecting when the tracked object is stationary and add "zero-velocity pseudo measurements" to the state estimator during the stationary period. For example, [12] and [13] discussed the use of this technique to reduce pedestrain tracking error.

In this work, we propose a strategy for multicopter inertial navigation with only the accelerometer and rate gyroscope as sensors. In our proposed strategy, the multicopter moves by taking a series of short duration flights instead of a single long duration flight. Between the short flights, the vehicle remains stationary on the ground and we introduce zero-velocity pseudo measurements to an extended Kalman Filter (EKF) to reduce the state estimation error. Analytical

analysis of state estimation error of this method is given and experiments were done to evaluate its effectiveness.

The proposed method is especially helpful in challenging environments, because 1) it does not require any infrastructure and can be easily deployed; 2) rapid technological advancement of Micro-electromechanical system (MEMS) based IMUs have made them cheap, small and lightweight to be widely used on small multicopters; 3) it is unaffected by GPS-denied, limited visibility (e.g. smoky) or featureless environments. For example, when a multicopter is used to help firefighters to get information about a building on fire, this method can be used to navigate the vehicle to go through sections with dense smoke when other sensors (e.g. cameras and LiDARs) used for navigation are temporarily unavailable.

II. Multicopter modelling

In this section we define the reference frames, briefly introduce the dynamic model of a multicopter, and analyze the accelerometer and rate gyroscope error characteristics.

A. Multicopter dynamics

As shown in Fig. 2 an inertial frame I attached to the ground and a body frame B attached to the Center of Mass (COM) of the multicopter, are defined. The multicopter is defined as a rigid body with six degrees of freedom: three degrees of freedom from the linear translation $\boldsymbol{p}$ along the three axes of the inertial frame and three degrees of freedom from the three-axis rotation from the body frame to the inertial frame, described by an orthogonal rotation matrix $\boldsymbol{R}$. Denote the thrust produced by each propeller as $\boldsymbol{f}_i$, expressed in the vehicle's body frame. With linear velocity $\boldsymbol{v}$, linear acceleration $\boldsymbol{a}$ and the gravity acceleration $\boldsymbol{g}$, all expressed in the inertial frame, the translational dynamics of the vehicle is expressed as

$$\frac{d}{dt}\boldsymbol{p} = \boldsymbol{v} \tag{1}$$

$$\frac{d}{dt}\boldsymbol{v} = \boldsymbol{a} \tag{2}$$

$$m\boldsymbol{a} = m\boldsymbol{g} + \boldsymbol{R}\sum \boldsymbol{f}_i \tag{3}$$

Denote the angular velocity of the vehicle as $\boldsymbol{\omega} = (w_1, w_2, w_3)$ and torque produced by each propeller as $\boldsymbol{\tau}_i$, both expressed in the vehicle's body frame. The rotational dynamics of the vehicle is expressed as

$$\frac{d}{dt}\boldsymbol{R} = \boldsymbol{R}\,S(\boldsymbol{\omega}) \tag{4}$$

$$\boldsymbol{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} + \sum \boldsymbol{\tau}_i \tag{5}$$

where $S(\boldsymbol{\omega})$ is the skew-symmetric matrix form of the vector cross product such that

$$S(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{6}$$

A detailed description of multicopter dynamics can be found in e.g. [14] [15].

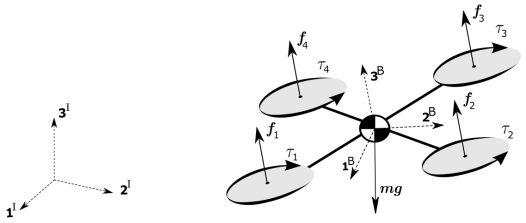

Fig. 2. Definition of reference frames. I represents the inertial reference frame and B represents the vehicle's body frame.

B. Sensor error characteristics

In this work we focus on MEMS IMUs consisting of a gyroscopes and an accelerometer, which are widely used on micro multicopters because of their small size and weight, low power consumption and low cost. The rate gyroscope measures angular velocity and the accelerometer measures proper acceleration (acceleration relative to a free-fall) of the vehicle, both measurements are expressed in the body frame B.

Measurement errors of MEMS IMUs usually include the following types [11]:

1) Bias: the offset of IMU measurements from the true values, which can be compensated by simply subtracting out from IMU outputs.
2) Thermo-mechanical white noise
3) Bias instability: the change in IMU bias.
4) Scale factor error and nonlinearity.

Among these four types of error, the thermo-mechanical white noise and uncorrected bias and scale factor error are usually the most significant errors [11]. In the proposed inertial navigation method, the bias and scale error are corrected by IMU calibration. The sensors are modeled as

$$\boldsymbol{\alpha} = \boldsymbol{R}^{-1}(\boldsymbol{a} - \boldsymbol{g}) + \boldsymbol{n}_\alpha \tag{7}$$

$$\boldsymbol{\gamma} = \boldsymbol{\omega} + \boldsymbol{n}_\gamma \tag{8}$$

where $\boldsymbol{\alpha}$ is the measurement of accelerometer and $\boldsymbol{n}_\alpha$ is accelerometer's measurement noise. Similarly, $\boldsymbol{\gamma}$ is the measurement of the rate gyroscope, and $\boldsymbol{n}_\gamma$ is rate gyroscope's noise. The variance of the noises are $\sigma_\alpha^2\boldsymbol{I}$ and $\sigma_\gamma^2\boldsymbol{I}$ for the acceleromenter and rate gyroscope respectively, both noises are assumed to be isotropic.

III. State estimation

An extended Kalman Filter (EKF) is used for the state estimation of the vehicle according to the method proposed in [16]. The estimator's state vector $\hat{\boldsymbol{\xi}}$ consists of 9 elements:

$$\hat{\boldsymbol{\xi}} = \begin{bmatrix} \hat{\boldsymbol{p}} \\ \hat{\boldsymbol{v}} \\ \hat{\boldsymbol{\delta}} \end{bmatrix} \tag{9}$$

where $\hat{\boldsymbol{p}}$ is vehicle's estimated position, $\hat{\boldsymbol{v}}$ is vehicle's estimated velocity and $\hat{\boldsymbol{\delta}}$ is estimated three-dimensional attitude

error with respect to the reference orientation $\hat{\boldsymbol{R}}_{ref}$. The estimated orientation of the vehicle is represented by

$$\hat{\boldsymbol{R}} = \hat{\boldsymbol{R}}_{\text{ref}}(t) \exp\left(S\left(\hat{\boldsymbol{\delta}}\right)\right) \tag{10}$$

where $\exp(\cdot)$ is matrix exponential so that $\exp S\left(\hat{\boldsymbol{\delta}}\right)$ represents rotation matrix.

A. Prediction

During the prediction step, the estimator does not use dynamics equations of the vehicle, but uses measurements from the accelerometer and rate gyroscope instead. Thus, it does not require knowing any dynamics parameters of the vehicle. The prediction step follows the following difference equations

$$\begin{aligned}\hat{\boldsymbol{\xi}}_p(t+\Delta t) &= \hat{\boldsymbol{\xi}}(t) + \begin{bmatrix} \frac{d}{dt}\hat{\boldsymbol{p}}(t) \\ \frac{d}{dt}\hat{\boldsymbol{v}}(t) \\ \frac{d}{dt}\hat{\boldsymbol{\delta}}(t) \end{bmatrix} \Delta t \\ &= \begin{bmatrix} \hat{\boldsymbol{p}}(t) + \hat{\boldsymbol{v}}(t)\Delta t \\ \hat{\boldsymbol{v}} + \left(\left(\hat{\boldsymbol{R}}_{\text{ref}}(t) \exp\left(S\left(\hat{\boldsymbol{\delta}}(t)\right)\right)\right)^{-1} \boldsymbol{\alpha}(t) + \boldsymbol{g} \right) \Delta t \\ \boldsymbol{\delta}(t) + \left(\boldsymbol{\gamma}(t) - \frac{1}{2} S(\boldsymbol{\gamma}(t))\, \boldsymbol{\delta}(t)\right) \Delta t \end{bmatrix} \end{aligned} \tag{11}$$

The estimated variance is given by

$$\boldsymbol{P}_{\xi\xi,p}(t+\Delta t) = \boldsymbol{A}(t)\boldsymbol{P}_{\xi\xi}(t)\boldsymbol{A}(t)^T + \boldsymbol{Q} \tag{12}$$

where

$$\begin{aligned}\boldsymbol{A}(t) &= \begin{bmatrix} \boldsymbol{I} & \boldsymbol{I}\Delta t & 0 \\ 0 & \boldsymbol{I} & S\left(\hat{\boldsymbol{R}}_{\text{ref}}(t)^{-1}\boldsymbol{\alpha}(t)\right)\Delta t \\ 0 & 0 & \boldsymbol{I} - \frac{1}{2} S(\boldsymbol{\gamma}(t))\, \Delta t \end{bmatrix} \\ \boldsymbol{Q} &= \text{diag}\left[0, \sigma_\alpha^2 \boldsymbol{I}, \sigma_\gamma^2 \boldsymbol{I}\right] \end{aligned}$$

The attitude error $\boldsymbol{\delta}$ is then set to zero after each prediction, by updating the reference attitude $\boldsymbol{R}_{\text{ref}}$ and covariance matrix. The details of this update can be found in [17].

B. Zero-velocity update

When the vehicle stays still on the ground, the knowledge that the vehicle is not moving can be utilized to improve state estimation performance. The proposed zero-velocity detector is based on rate gyroscope only for computational simplicity, since [18] has shown that using both measurements from the accelerometer and rate gyroscope only gives marginal performance improvement compared to using rate gyroscope's measurements only. The zero velocity detector has two tuning parameters, $N_{\text{threshold}}$ and $\gamma_{\text{threshold}}$. If the magnitude of the rate gyroscope's measurement is below $\gamma_{\text{threshold}}$ for more than $N_{\text{threshold}}$ continuous time steps, the vehicle is considered stationary and it is considered to be moving otherwise. For the hardware we use, $N_{\text{threshold}}$ was tuned to be 20 and $\gamma_{\text{threshold}}$ was tuned to be 0.2 rad/s. A demonstration of the zero velocity detector is shown in Fig. 3.

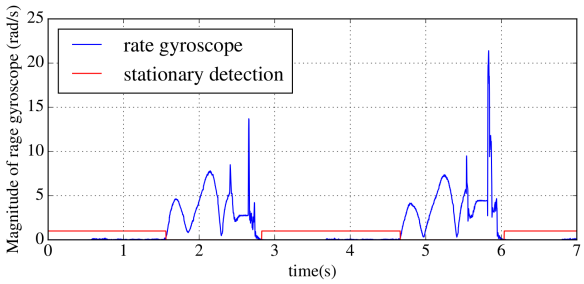

Fig. 3. A demonstration of the implemented zero velocity detector based on the magnitude of rate gyroscope measurements. The value of zero velocity detection state is either 1 (stationary) or 0 (moving). The vehicle is "hopping" from 1.55s to 2.75s and from 4.7s to 6.0s. It remains stationary on the ground for the rest of the time.

When the vehicle is detected to be stationary, zero-velocity updates as pseudo measurements are introduced to the state estimator. The observation matrix H is

$$\boldsymbol{H} = \left[0, \boldsymbol{I}_{3\times 3}, 0\right]$$

For notational convenience, we set $t \leftarrow t + \Delta t$. Follow the standard EKF formalism and we can get

$$\boldsymbol{K}(t) = \boldsymbol{P}_{\xi\xi,p_+}(t)\boldsymbol{H}^T \left(\boldsymbol{H}\boldsymbol{P}_{\xi\xi,p_+}(t)\boldsymbol{H}^T\right)^{-1} \tag{13}$$

$$\hat{\boldsymbol{\xi}}_m(t) = \hat{\boldsymbol{\xi}}_{p_+}(t) + \boldsymbol{K}(t)\left(-\hat{\boldsymbol{v}}_{p_+}(t)\right) \tag{14}$$

$$\boldsymbol{P}_{\xi\xi,m}(t) = \left(\boldsymbol{I} - \boldsymbol{K}(t)\boldsymbol{H}\right)\boldsymbol{P}_{\xi\xi,p_+} \tag{15}$$

$$\hat{\boldsymbol{R}}_{\text{ref}\,m}(t) = \hat{\boldsymbol{R}}_{\text{ref}\,p_+}(t) \tag{16}$$

Note that the zero-velocity pseudo measurements has zero variance. The $\hat{\boldsymbol{\delta}}_m(t)$ is then reset to zero by updating the reference attitude $\boldsymbol{R}_{\text{ref}}$ and covariance matrix. The details of this update can be found in [17].

IV. Multicopter Inertial Navigation

In this section, we provide a simplified analysis of the state estimation error of inertial navigation and based on the analysis propose a special flight motion to the state estimation error.

A. Error analysis

State estimation by directly integrating the measurements from IMU would cause the estimation error to grow rapidly. To illustrate rapid growth of error, a simplified analysis of the position estimation error on one direction is given below. Linearize the attitude at hovering, where the proper acceleration is equal to $-\boldsymbol{g}$. We define $\hat{\boldsymbol{\xi}} = \left[p, v, \theta\right]^\top$, where p, v and θ represent position, velocity and attitude respectively, from (1), (2), (7) and (8), we can get

$$\frac{d}{dt}\hat{\boldsymbol{\xi}}(t) = \boldsymbol{A}\hat{\boldsymbol{\xi}}(t) + \begin{bmatrix} 0 \\ \sigma_\alpha \\ \sigma_\gamma \end{bmatrix} \tag{17}$$

$$\frac{d}{dt}\boldsymbol{P}(t) = \boldsymbol{A}\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A}^\top + \boldsymbol{Q} \tag{18}$$

where

$$\boldsymbol{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & g \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{Q} = \mathrm{diag}\big[0, \sigma_\alpha^2, \sigma_\gamma^2\big]$$

We define the initial variance of the states to be $\boldsymbol{P}(0) = \mathrm{diag}[P_{pp}(0), P_{vv}(0), P_{\theta\theta}(0)]$. The variance of the states at time t are given by

$$P_{\theta\theta}(t) = \sigma_\gamma^2 t + P_{\theta\theta}(0) \tag{19}$$

$$P_{vv}(t) = \frac{\sigma_\gamma^2 g^2}{3} t^3 + g^2 P_{\theta\theta}(0) t^2 + \sigma_\alpha^2 t + P_{vv}(0) \tag{20}$$

$$P_{pp}(t) = \frac{\sigma_\gamma^2 g^2}{20} t^5 + \frac{g^2 P_{\theta\theta}(0)}{4} t^4 + \frac{\sigma_\alpha^2}{3} t^3 + P_{vv}(0) t^2 + P_{pp}(0) \tag{21}$$

which shows that after time t the additive noise from the IMU causes the variance of position grow on the order of t^5.

B. Motion planning

The simplified error analysis in section IV-A points towards a motion planning strategy to reduce state estimation variance: planning a path with many short duration "hops". Between the hops the vehicle remains stationary on the ground and we can partially reset the state uncertainty. A single flight of duration t is broken into N hops of time t/N. After each hop the vehicle stays stationary and the estimation variance of velocity estimation is set to zero. So the velocity estimation variance at the beginning of each hop flight is zero. From (19) - (21), the estimation variance of θ at the end of short flight i ($i \in \{1, 2, ..., N\}$) is given by

$$P_{\theta\theta}\left(\frac{i}{N} t\right) = P_{\theta\theta}(0) + \sigma_\gamma^2 \frac{i}{N} t \tag{22}$$

and estimation variance of position has the following relationship

$$P_{pp}\left(\frac{i+1}{N} t\right) = \frac{\sigma_\gamma^2 g^2 t^5}{20 N^5} + \frac{g^2 P_{\theta\theta}(\frac{i}{N} t) t^4}{4N^4} + \frac{\sigma_\alpha^2 t^3}{3N^3} + P_{pp}(\frac{i}{N} t) \quad (i \geq 1) \tag{23}$$

$$P_{pp}\left(\frac{t}{N}\right) = \frac{\sigma_\gamma^2 g^2 t^5}{20 N^5} + \frac{g^2 P_{\theta\theta}(0) t^4}{4N^4} + \frac{\sigma_\alpha^2 t^3}{3N^3} + \frac{P_{vv}(0) t^2}{N^2} + P_{pp}(0) \quad (i = 0) \tag{24}$$

From (22) - (24) the position estimation variance after N short motions is

$$P_{pp}(t) = \frac{(5N-3)\sigma_\gamma^2 g^2 t^5}{40 N^4} + \frac{g^2 P_{\theta\theta}(0) t^4}{4N^3} + \frac{\sigma_\alpha^2 t^3}{3N^2} + \frac{P_{vv}(0) t^2}{N^2} + P_{pp}(0) \tag{25}$$

The effect of this motion planning strategy can be seen by comparing (25) to (21). Assume, for example, the initial condition of the system is perfectly known, and the system has a perfect rate gyroscope, such that the state estimation uncertainty only comes from the the accelerometer, the final position estimation variance is reduced by a factor of N^{-2} when t is large.

In Section III, we use an extended Kalman Filter for state estimation. The zero-velocity pseudo measurement would provide a even greater reduction in position estimation variance compared to the error analysis in this section because of the correlation between velocity and position as well as the correlation between velocity and attitude.

For trajectory planning of each hop, we use the method proposed in [19], a trajectory generation method which minimizes the jerk (third derivative of position) of the multicopter given the initial and desired final states. The solution of minimum-jerk trajectory is provided in closed form and is computationally inexpensive, which makes it suitable to be implemented on embedded flight controllers. In addition, the method verifies if the planned trajectory satisfies the vehicle's actuation constraints (e.g. maximum thrust of the motor) and does not collide with known planar obstacles (e.g. the ground).

The vehicle's initial velocity and acceleration are zero because of the stationary state between the hops. The final velocity of the hop trajectory is zero such that the vehicle stops moving after each hop. Between each hop, the vehicle remains stationary on the ground, and the zero-velocity pseudo measurements are introduced to reduce the estimation variance of the state estimator. Although a shorter hop duration is helpful to reduce state estimation error, too aggressive trajectories will make the trajectory tracking difficult. As a result, the trade-off between the aggressiveness of the hop trajectory and trajectory tracking should be considered when choosing the hop time. The height of the hop trajectory can be adjusted by changing the final acceleration on the vertical direction.

V. Experimental evaluation

A. Experimental setup

Experiments were conducted to evaluate the performance of the proposed inertial navigation method, and can be seen in the video attachment of this paper. A custom-built quadcopter, as shown in Fig. 4 was used in the experiments. The quadcopter weighs 165g including the battery. The distance between the hubs of two diagnal motors is 117.6mm and the propeller is 76.2mm in diameter. A small analog camera is installed on the vehicle for video streaming.

A Crazyflie 2.0 [20] running a modified version of PX4 firmware was used as a low-level flight controller for the quadcopter. It is equipped with a consumer-level MEMS IMU (InvenSense MPU-9250) with a measurement frequency of 500Hz. The trajectory tracking flight controller and the proposed inertial state estimator ran on this microcontroller at 500Hz.

Fig. 4. The quadcopter used in the experiments.

The experiments were done in an indoor flight space of size $7 \times 6 \times 5$m. A commercial motion capture system, which provides high-accuracy, high-rate state information, was used during the experiments to provide ground truth of the vehicle's states.

B. Performance evaluation

1) The proposed strategy: The proposed inertial state estimator's performance was quantified by comparing with state estimation from the motion capture system. Note that during the experiments, the motion capture system is only used for ground truth, not for control of the vehicle. In the experiments, the vehicle was commanded to fly in a constant direction for 5 meters in 5 hops, each hop had length of 1m and took 1 second. There was a two-second interval between two hops, when the vehicle is stationary on the ground. The state estimator introduced in Section III was used to estimate the state of the vehicle and a cascaded PD controller was used for trajectory tracking.

The experiment was repeated eight times. The state estimation error of the proposed inertial state estimator is shown in Fig. 5. At the end of the flight, the root mean square error (RMSE) for position estimation was 0.13m, 0.12m and 0.03m for the downrange direction (flight direction), crossrange direction (perpendicular to the flight direction and in the horizontal plane) and vertical direction, respectively. The mean absolute position estimation error was 0.17m, which was 3.4% of the total flight distance. The trajectory of the vehicle (measured by the motion capture system), is compared with the reference trajectory in Fig. 6. At the end of the flight, the RMSE of position tracking was 0.19m and 0.06m for the crossrange and downrange direction. The position tracking error was zero for the vertical direction because the floor in the flight space is horizontal and flat. The mean absolute tracking error was 0.16m, which was 3.2% of the total flight distance.

2) Long duration flights as comparison: Experiments with a long duration flight instead of multiple short-duration hops are conducted to compare with the proposed strategy. In the experiments, the vehicle was commanded to fly in a constant direction for 5 meters in 5 seconds. The motion capture system was used during the flight for state estimation for control because inertial navigation for 5 seconds would

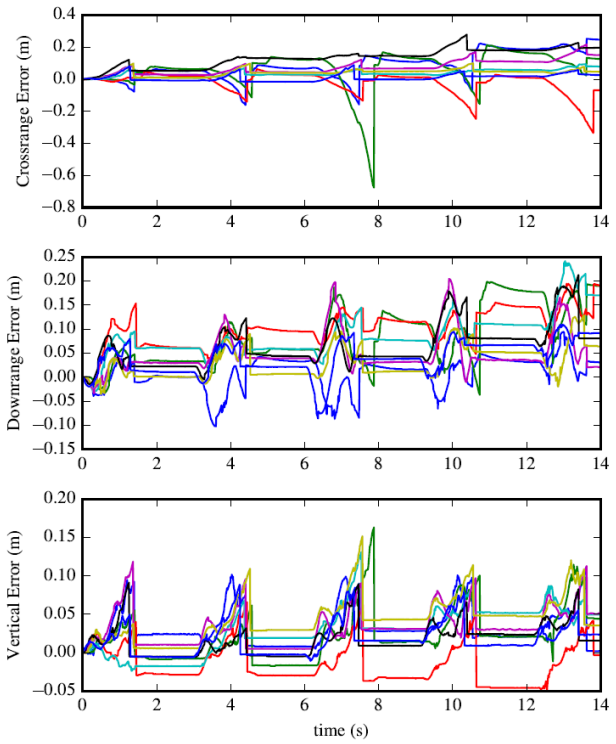

Fig. 5. Position estimation of the vehicle, using only the rate gyroscope and accelerometer. No knowledge of the environment, except that the floor is not moving, is assumed. The effect of zero velocity measurement update could be seen at around 1.5s, 4.5s, 7.5s, 10.5s and 13.5s for reducing the estimation error.

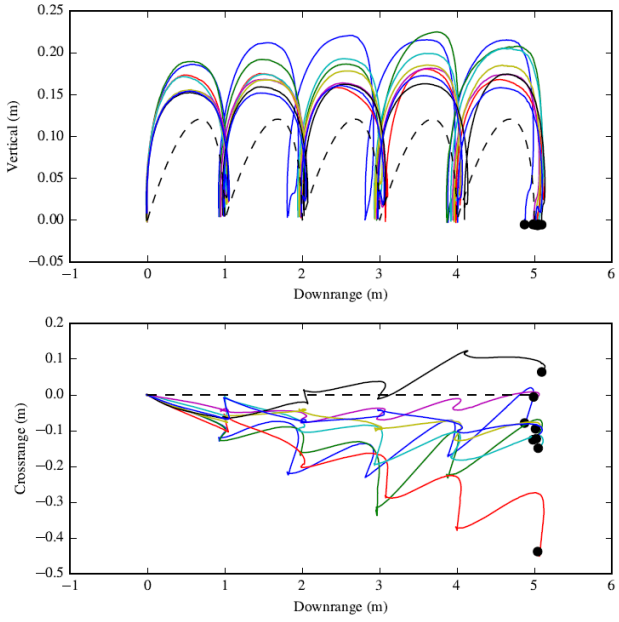

Fig. 6. Closed-loop control of the vehicle using the proposed inertial navigation estimator. Trajectories of 8 separate flights are shown in solid lines with different color and the final position of each flight is marked by a solid circle. The reference trajectory is shown as a black dashed line.

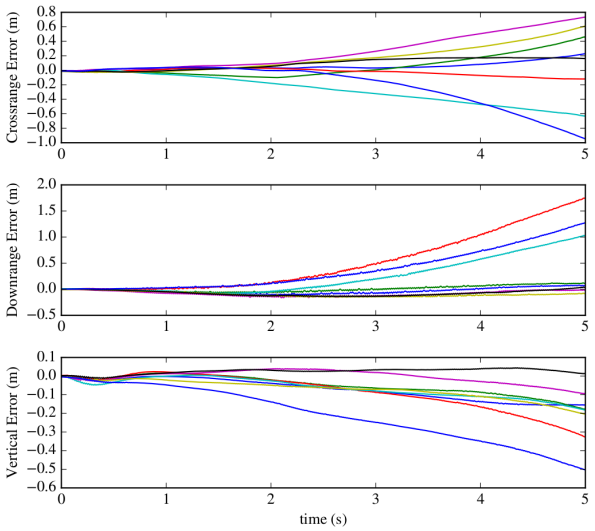

Fig. 7. Position estimation error of the vehicle for a single long duration flight without using the proposed strategy.

give a large state estimation error and can crash the vehicle. After the experiments, we run the inertial navigation off-board using the collected IMU data. We then compare the position estimation from inertial navigation with position estimation from the motion capture system, which is used as ground truth. The experiment was repeated eight times and the state estimation error is shown in Fig. 7. At the end of the flight, the root mean square error (RMSE) for position estimation was 0.85m, 0.56m and 0.25m for the downrange, crossrange and vertical direction, respectively. The mean absolute position estimation error was 0.88m, which is 17.6% of the total flight distance, and is 4.2 times larger than the proposed strategy. The position estimation error of these two methods are compared in table I.

TABLE I

POSITION ESTIMATION ERROR COMPARISON

position estimation error	proposed strategy	long-distance flight
downrange RMSE	0.13m	0.85m
crossrange RMSE	0.12m	0.56m
vertical RMSE	0.03m	0.25m
mean absolute error	0.17m	0.88m

C. Motivating example

In addition to the experiments in the flight space, an additional experiment was conducted where the vehicle flew a longer distance around a corner. The experiment environment is shown in Fig. 8. This experiment demonstrates the ability of the proposed inertial navigation strategy to navigate a multicopter in real-world environment. The vehicle first flew forward for 6m in 6 hops and then made a left turn and flew 4m in 4 hops. Each hop was 1 second. The proposed strategy successfully navigates the vehicle and final position tracking error was about 0.3m, which is about 3% of the total flight distance, a similar result as before.

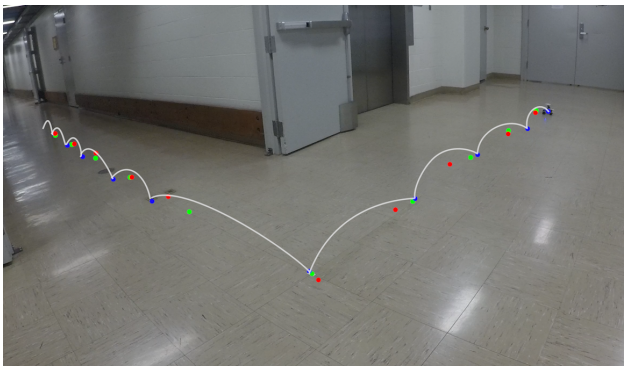

Fig. 8. The vehicle flew 10m in 10 hops around a corner. The green dots mark the target positions, the blue dots mark the actual position of the vehicle after each hop, and the red dots mark the estimated positions. The white curves represents the actual flight path.

VI. Conclusion and Future Work

In this work, an inertial navigation strategy for multicopters was proposed. The proposed method is based only on measurements from the on-board accelerometer and rate gyroscope and is especially suitable for challenging environments where other sensors are unavailable. An error analysis of the state estimation error of inertial navigation was introduced and based on this analysis a motion planning method of breaking a long time flight into multiple short time flight steps was proposed to reduce the estimation error.

Indoor experiments were repeated multiple times to evaluate the performance of this state estimation method, using a standard quadcopter equipped with a consumer-level IMU. The state estimator was used for closed-loop control of the multicopter. The experiments showed that the mean absolute position estimation error of the proposed state estimator at the end of the flight was 0.17m for translation of 5m, which was 3.4% of the total distance, and the final mean absolute trajectory tracking error was 0.16m, which was 3.2% of the total distance.

For future work, we would like to investigate the optimal trajectory for each flight step. Intuitively, a trajectory with large acceleration initially (when uncertainty in orientation estimation is low) and with decreasing acceleration as time increases (due to increasing uncertainty in the vehicle's orientation), could possibly achieve the minimum inertial navigation error. We also plan to design a multicopter with robust mechanical structure to implement the proposed inertial navigation method, and evaluate the method's performance in complex, challenging environments.

Acknowledgements

This work was supported by the Defense Advanced Research Projects Agency (DARPA) Subterranean Challenge. The experimental testbed at the HiPeRLab is the result of contributions of many people, a full list of which can be found at `hiperlab.berkeley.edu/members/`.

REFERENCES

[1] S. Lupashin, M. Hehn, M. W. Mueller, A. P. Schoellig, M. Sherback, and R. D'Andrea, "A platform for aerial robotics research and demonstration: The flying machine arena," *Mechatronics*, vol. 24, no. 1, pp. 41 – 54, 2014.

[2] M. Field, D. Stirling, F. Naghdy, and Z. Pan, "Motion capture in robotics review," in *2009 IEEE International Conference on Control and Automation*, Dec 2009, pp. 1697–1702.

[3] M. W. Mueller, M. Hamer, and R. D'Andrea, "Fusing ultra-wideband range measurements with accelerometers and rate gyroscopes for quadrocopter state estimation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1730–1736.

[4] S. Fahandezh-Saadi and M. W. Mueller, "Optimal measurement selection algorithm and estimator for ultra-wideband symmetric ranging localization," *CoRR*, vol. abs/1804.09773, 2018.

[5] A. R. Vidal, H. Rebecq, T. Horstschaefer, and D. Scaramuzza, "Ultimate slam? combining events, images, and imu for robust visual slam in hdr and high-speed scenarios," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 994–1001, April 2018.

[6] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, "Fusion of imu and vision for absolute scale estimation in monocular slam," *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1, pp. 287–299, Jan 2011.

[7] C. Cadena, L. Carlone, H. Carrillo, Y. Latif, D. Scaramuzza, J. Neira, I. D. Reid, and J. J. Leonard, "Simultaneous localization and mapping: Present, future, and the robust-perception age," *CoRR*, vol. abs/1606.05830, 2016.

[8] M. Brossard, A. Barrau, and S. Bonnabel, "AI-IMU dead-reckoning," *CoRR*, vol. abs/1904.06064, 2019. [Online]. Available: http://arxiv.org/abs/1904.06064

[9] G. Allibert, D. Abeywardena, M. Bangura, and R. Mahony, "Estimating body-fixed frame velocity and attitude from inertial measurements for a quadrotor vehicle," in *2014 IEEE Conference on Control Applications (CCA)*, Oct 2014, pp. 978–983.

[10] J. Svacha, K. Mohta, M. Watterson, G. Loianno, and V. Kumar, "Inertial velocity and attitude estimation for quadrotors," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Oct 2018, pp. 1–9.

[11] O. J. Woodman, "An introduction to inertial navigation," University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-696, Aug. 2007.

[12] I. Skog, P. Handel, J. Nilsson, and J. Rantakokko, "Zero-velocity detectionâĂŤan algorithm evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 11, pp. 2657–2666, Nov 2010.

[13] E. Foxlin, "Pedestrian tracking with shoe-mounted inertial sensors," *IEEE Computer Graphics and Applications*, vol. 25, no. 6, pp. 38–46, Nov 2005.

[14] M. W. Mueller, "A dynamics-agnostic state estimator for unmanned aerial vehicles using ultra-wideband radios," 2018.

[15] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, Sep. 2012.

[16] M. W. Mueller, M. Hehn, and D. Raffaello, "Covariance correction step for kalman filtering with an attitude," *Journal of Guidance, Control, and Dynamics*, vol. 40, no. 9, pp. 2301–2306, Nov 2016.

[17] M. W. Mueller, "A dynamics-agnostic state estimator for unmanned aerial vehicles using ultra-wideband radios," 2018.

[18] I. Skog, P. Handel, J. Nilsson, and J. Rantakokko, "Zero-velocity detection – an algorithm evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 11, pp. 2657–2666, Nov 2010.

[19] M. W. Mueller, M. Hehn, and R. D'Andrea, "A computationally efficient motion primitive for quadrocopter trajectory generation," *IEEE Transactions on Robotics*, vol. 31, no. 6, pp. 1294–1310, 12 2015.

[20] Bitcraze. (2018) Crazyflie 2.0. [Online]. Available: www.bitcraze.io/crazyflie-2