An Ultra-Wideband-based Multi-UAV Localization System in GPS-denied environments

Thien Minh Nguyen, Abdul Hanif Zaini, Kexin Guo and Lihua Xie

ABSTRACT

This paper presents a technique for multi-UAV localization using ranging measurements from Two-Way Time-Of-Flight Ultra-Wideband (UWB) transceivers. In continuation with our previous work, the use of the Extended Kalman Filter (EKF) estimate is extended by fusing with other sensors to achieve a usable altitude estimate. Besides EKF, another method based on Non-linear Regression (NLR) is also developed to serve as an auxiliary localization to supplement the EKF. Experiments of autonomous flights are carried out to study the performance of these localization methods. Our success in supporting 4 UAVs demonstrates the capability of UWB for multi-UAV localization as a substitute for GPS in GPS-denied conditions. It can also serve as a cost-competitive alternative for very accurate yet expensive, highly centralized motion capture systems for indoor localization.

1 INTRODUCTION

Micro unmanned aerial vehicle (UAV) technology has plenty of potential and is seeing increasing application in recent years. Besides the well-reported military applications and the ubiquitous aerial photography consumer drones, UAVs are also being used for 3D mapping [1], aerial inspection [2], precision agriculture [3], as well as search and rescue [4]. Other noticeably interesting applications are also being explored such as deployment in indoor industrial environments [5], and even interception and capture of other UAVs [6].

Most of these applications rely heavily on the satellite-based localization system (Global Positioning System, GPS) which limits it to outdoor use. The system would thus fail to work indoors or in semi-cluttered urban environments and forests due to multipath and non-line-of-sight blockage. Hence, a satisfactory localization solution is desired to extend the use of UAV into complex environments.

The authors are with School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore 639798, Singapore. email: E150040@ntu.edu.sg (T.M. Nguyen); E150010@e.ntu.edu.sg (A.H.B. Zaini); GUOK0005@e.ntu.edu.sg (K. Guo); elhxie@ntu.edu.sg (L. Xie)

The research was partially supported by the ST Engineering - NTU Corporate Lab through the NRF corporate lab@university scheme.

For indoor localization, two infrastructure-dependent systems have attracted much attention from UAV research groups: a camera-based motion capture system, and a system that utilizes the existing wireless local area network. The camera-based motion capture system is commonly used by several research groups focusing on challenging autonomous aerial tasks such as formation, cooperative control and high speed maneuvers [7, 8]. Commercially-available motion capture systems are able to provide millimeter/degree accuracy and high update rates at upwards of 200Hz. However, the motion capture system is a highly centralized system utilizing many cameras connected to a single computer which then transmits the positions and orientations to the subjects within its view. It is also costly, has significant setup time, small sensing area, and can only be used indoors which limits use beyond research.

The second method utilizes the IEEE 802.11 standard usually referred to as WiFi. It has been used in commercial and residential buildings [9] and robot teams [10]. However, to date, no successful UAV flight using WiFi localization has been reported. This can be attributed to the unsatisfactory accuracy of WiFi-based localization methods for a demanding application such as UAV [11].

Recently, a third method for indoor localization has emerged which utilizes the pulse-based ultra-wide band (UWB) radio technology. The large bandwidth and pulse-based communication enables spectrum sharing and does not interfere with conventional signals. With the large bandwidth, this technology has the properties of strong multi-path resistance and, to some extent, wall-penetration, which enables accurate ranging via peer-to-peer two-way time of flight. With centimeter-level ranging accuracy, small size and light weight, low-power UWB modules can be applied for UAV localization [12]. UAV localization has also been achieved using time of arrival and time difference of arrival with one way UWB communications [13]. The work reported in this paper builds on the group's previous work in [12].

In this paper, PulsON 440 UWB modules from Time Domain [14] are installed on multiple quadcopters for localization in GPS-denied environments. Each UAV sends range requests to anchor nodes in a round-robin manner. Once a range measurement is retrieved, based on our proposed localization algorithm, position and velocity estimates will be updated and fed into the flight control unit of the quadcopters. A combination of Nonlinear Regression (NLR) and extended Kalman filter (EKF) localization methods is proposed where NLR can be used to initialize and reset the EKF estimates. Also, due

to the significance of the accuracy of anchors' coordinates for our localization estimation, a tedious procedure of measuring the anchors' coordinates must be conducted every time the operation sites are altered or the anchors are moved. Therefore, the proposed NLR method can also be used to assign the coordinates to the anchors. An autonomous formation flight test with a four-UAV team are also conducted and the data were studied to validate the performance of our proposed localization system,.

The content of this paper is organized as follows. In the first part we formulate the Extended Kalman Filter and Nonlinear Regression algorithms that use the distances to produce the locations of the agents. In the second part we describe our experiments where we fly 4 UAVs using the EKF estimates while running the other method simultaneously and summarize their performance. Finally, we conclude on the performance of the UWB-based localization system and its capability to support multiple UAVs as well as discuss issues that can be investigated for future works.

2 Localization Techniques

2.1 Extended Kalman Filter

In this paper the Extended Kalman Filter is based on our previous work [12]. Let p be the aircraft's position, $v = \dot{p}$ be its velocity, $w(t)$ is a white noise process ($w(t) \in \mathbb{R}^3$). The UAV's motion is modelled as a random acceleration second-order system as follows:

$$X = \begin{bmatrix} p \\ v \end{bmatrix}; \qquad \dot{X} = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix} X(t) + \begin{bmatrix} 0 \\ I \end{bmatrix} w(t)$$

The model above is discretized as:

$$X_k = \begin{bmatrix} p_k \\ v_k \end{bmatrix}; \qquad X_{k+1} = A_k X_k + w_k$$

where:
$A_k = \begin{bmatrix} I & \Delta t_k I \\ 0 & I \end{bmatrix}$ is a 6×6 discrete state transition matrix, Δt_k is the time difference between the last update and the current update. $\{w_k\}$ is a white noise sequence with its covariance derived in [12] as:

$$Q_k = \begin{bmatrix} \frac{1}{4}(\Delta t_k)^4 & \frac{1}{2}(\Delta t_k)^3 \\ \frac{1}{2}(\Delta t_k)^3 & (\Delta t_k)^2 \end{bmatrix} \otimes \mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2)$$

Q_k is a 6×6 covariance matrix for the random acceleration sequence. Here $\otimes$ is the Kronecker product and $\mathrm{diag}(\sigma_x^2, \sigma_y^2, \sigma_z^2)$ is a 3×3 diagonal matrix whose non-zero entries are the variance of the random acceleration on each direction.

We assume that the measurement is the distance from an anchor (whose position is denoted as p^a) to the UAV's position subject to a zero-mean Gaussian noise ν_k with variance R: $d_k = \|p_k - p^a\| + \nu_k$. The EKF algorithm will run through the following steps every time a new distance estimate is received:

$$\hat{X}^*_{k+1} = A_k \hat{X}_k \tag{1}$$
$$\hat{P}^*_{k+1} = A_k \hat{P}_k A_k^T + Q_k \tag{2}$$
$$\hat{d}_{k+1} = \left\| p^*_{k+1} - p^a \right\| \tag{3}$$
$$H_{k+1} = \left[\frac{1}{\hat{d}_{k+1}} (p^*_{k+1} - p^a)^T \middle| \mathbf{0}_{1\times 3} \right] \tag{4}$$
$$S_{k+1} = R + H_{k+1} \hat{P}^*_{k+1} H^T_{k+1} \tag{5}$$
$$K_{k+1} = \hat{P}^*_{k+1} H^T_{k+1} S^{-1}_{k+1} \tag{6}$$
$$\hat{X}_{k+1} = \hat{X}^*_{k+1} + K_{k+1} \left[d_{k+1} - \hat{d}_{k+1} \right] \tag{7}$$
$$\hat{P}_{k+1} = (I - K_{k+1} H_{k+1}) \hat{P}^*_{k+1} \tag{8}$$

The following Mahalanobis distance of each distance measurement is also calculated between step (5) and step (6):

$$D_M = \sqrt{(d_{k+1} - \hat{d}_{k+1})^T S^{-1}_{k+1} (d_{k+1} - \hat{d}_{k+1})}$$

If D_M is larger than a threshold then it is likely to be an outlier and we should skip fusing this measurement in step (7) and step (8).

In our previous work, the EKF estimate was used mainly as a 2D localization system. Therefore an independent laser-based range finding sensor was used for altitude estimate. Although the sensor is accurate, the UAV's altitude estimate is influenced by the terrain. In this work, we have combined the UWB altitude estimate with the low level sensors to achieve altitude control for UAV, independent from the variation of the terrain.

2.2 Substituting GPS in low level fusion

In our platforms, we utilize the high-level EKF estimate as the GPS input for the sensor fusion algorithm of the 3DR Pixhawk [15]. Our Pixhawk units run a customized version of the open-source PX4 [16] firmware and are set to use the 22-state EKF attitude and position estimator. In this 22-state EKF, the position, velocity, and attitude states are predicted by integration of acceleration and angular rate measurements from the IMU. In the correction step, GPS position measurement inputs are substituted and the relevant noise parameters are adjusted accordingly.

To ensure reliable and stable altitude estimation, the altitude data (received at a rate nearly 32Hz) are fused with the existing barometer-based altitude readings (sampled at 100Hz). The standard deviations of both measurements were found to be similar when held stationary at 18cm and 20cm respectively. However, the slower measurement rate of the noisy UWB-based altitude measurements does not provide the simultaneously smooth and responsive estimate required for altitude control. Nevertheless, this estimate still has the advantage that it is not subject to the variation of the ground level. As the barometer altitude reading is an absolute value

measured from sea level, an offset is subtracted from the barometer measurements. This offset is obtained by calculating the difference between barometer measurement and the EKF altitude estimate every 10 seconds as shown in Figure 1. This arrangement simultaneously compensates for barometer drift and dampens sudden changes in UWB-based altitude measurement.

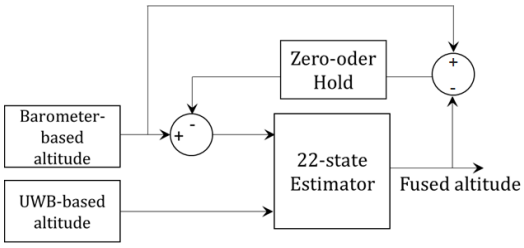

Figure 1: The method used for offsetting the barometer.

2.3 *Non-linear regression for coordinates assignment and localization*

Since our localization method requires setting up the anchors at the operation site, it usually would be very cumbersome, or sometimes impossible, for human to access the operation area to measure the coordinates of the anchors before the UAV team can be deployed. In this part we propose a method based on Non-linear regression to assign the coordinates to the anchors/UAVs on startup. This method can be implemented based on the communication capability of the UWB transceivers in the future.

Besides coordinates assignment, the algorithm can also be run in parallel with the EKF to serve as initializer and recovery value. A secondary localization technique is useful in this situation because there can be cases when the signal is lost for sometime (for example, 1 second), the step time Δt_k can be too large and cause the EKF to diverge. Also, since we assume a random acceleration model, when the UAV moves faster than the estimator can catch up, an over-conservative outlier rejection can cause it to stall completely. The NLR method does not require the past estimate unlike the EKF. Therefore, if enough distance measurements can be obtained in a short period, we can use the NLR to calculate the position and reset the EKF. In the Experiments section the performance of the NLR localization will be summarized and compared with the EKF.

Assume that we have a network of N agents, which can be either anchor or UAV, and each agent has its coordinates denoted as $p_i = \begin{bmatrix} x_i\ y_i\ z_i \end{bmatrix}^T$, $i = 1, 2...N$. An agent can have none, some or all of its coordinates known beforehand. In the context of a non-linear regression problem, the known coordinates will be in the set of error-free independent variables $\mathbf{x}$ and the unknown will be in the set of parameters β to be estimated. Regardless of their roles we will group these coordinates into one vector of length $3N$ as

$$\boldsymbol{p} = \begin{bmatrix} x_1\ y_1\ z_1\ x_2\ y_2\ z_2\ \dots\ x_N\ y_N\ z_N \end{bmatrix}^T$$

Let $d_{ij} = \|p_i - p_j\|$ be the distance between p_i and p_j, where $i, j = 1, 2, 3...N$ and $i \neq j$. Also since $d_{ij} = d_{ji}$ we only consider d_{ij} of $i < j$. The number of distances from all N agents in the network will be $K = N(N-1)/2$. Our non-linear model, denoted as $f(\mathbf{x}, \beta)$ or $f(\boldsymbol{p})$, will be a $K \times 1$ vector of all of the distances arranged in a particular way as follows:

$$\begin{bmatrix} d_{12} & d_{13}\ \dots\ d_{1N} & d_{23} & d_{24}\ \dots\ d_{2N}\ \dots & \dots\ d_{N-1,N} \end{bmatrix}^T$$

Our observed dependent variables y_{ij} are these distances subject to some error ϵ_{ij}, or $y_{ij} = d_{ij} + \epsilon_{ij}$.

To apply the Non-linear regression method, we start from some initial guess β_0 and seek to improve our estimate of β by the following recursive steps:

$$J_k = J_k(\mathbf{x}, \beta_k) = \left.\frac{\partial f(\mathbf{x}, \beta)}{\partial \beta}\right|_{\beta=\beta_k} \tag{9}$$

$$\Delta y = y - f(\mathbf{x}, \beta_k) \tag{10}$$

$$\Delta\beta = (J_k^T W J_k)^{-1} J_k^T W \Delta y \tag{11}$$

$$\beta_{k+1} = \beta_k + \Delta\beta \tag{12}$$

where $k = 0, 1, 2, 3...$ and W is a weight matrix to signify the comparative importance between the measurements.

During the initial phase when all agents are static W can be set as identity. In the second phase when some agents assume the role of anchors and some are the mobile nodes, the distances may not be acquired at the same time. The more recent a distance is, the more priority should be given to its residue Δy_i. Also, it should be noted that a time window should be applied to ignore the very old measurements. For example in a period from t_0 to t_m within our time window, we have measured the distances to m anchors at times t_1, $t_2, ...t_m$. W can be set as:

$$W = \frac{1}{\sum_{l=1}^{m} t_l - m t_0} \mathrm{diag}([\Delta t_1 \Delta t_2 \dots \Delta t_m])$$

where $\Delta t_l = t_l - t_0, l = 1, 2, ...m$.

To calculate the Jacobian J_k in equation (9) conveniently, we can employ a more expressive Jacobian $J_{\boldsymbol{p}} = \partial f(\mathbf{x}, \beta)/\partial \boldsymbol{p} = \partial f(\boldsymbol{p})/\partial \boldsymbol{p}$. $J_{\boldsymbol{p}}$ can be expressed as follows:

$$J_{\boldsymbol{p}} = \begin{bmatrix} \bar{J}_{12} & \bar{J}_{21} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \bar{J}_{13} & \mathbf{0} & \bar{J}_{31} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{J}_{1N} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \bar{J}_{N1} \\ \mathbf{0} & \bar{J}_{23} & \bar{J}_{32} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{0} & \bar{J}_{24} & \mathbf{0} & \bar{J}_{42} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \bar{J}_{2N} & \mathbf{0} & \mathbf{0} & \dots & \bar{J}_{N2} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \bar{J}_{N,N-1} \end{bmatrix} \tag{13}$$

where $\bar{J}_{ij} = \frac{(p_i - p_j)^T}{d_{ij}}$; $i, j \in \{1, 2, 3...N\}$ and $i \neq j$. $\mathbf{0}$ indicates a 1×3 block of zeros.

After we have substituted the values of $\mathbf{x}$ and β_k to (13), we can remove the columns corresponding to $\mathbf{x}$ to obtain J_k. For UAV localization, since only the UAVs are moving, all of the anchors' coordinates can be added to $\mathbf{x}$ so that the corresponding columns can be removed from $J_{\boldsymbol{p}}$. The same treatment can also be done to the rows corresponding to the distances between the anchors. Finally if the UAV only needs to localize itself then the final J_k matrix will be a $K^* \times 3$ matrix where K^* is the number of anchors.

Using NLR for coordinates assignment requires fixing some coordinates before estimating the others. The following lemma can provide several necessary conditions to assure the computational feasibility of the NLR algorithm.

Lemma 1. *Let l, m, n be the numbers of fixed x, y, z coordinates, respectively. The following are necessary conditions for J_k to have full rank:*

1. $(l + m + n) \geq 6$.
2. *The known coordinates must belong to at least three agents.*
3. $l \geq 1$, $m \geq 1$, $n \geq 1$.
4. *Any two in three numbers l, m, n must not be both equal to 1.*

Remark: Lemma 1 only specifies some conditions for the computational feasibility of the algorithm regarding the number of known coordinates. Some numerical instances of β_k may still result to loss of rank of J_k, especially when the agents are co-linear or coplanar. Therefore a check for infinity or NaN value in $\Delta\beta$ is still necessary. Also, since NLR is a gradient descent method, the solution may converge to a non-global optimum, therefore a good initial guess β_0 is also very important.

3 Experiments and Results

3.1 Testbed

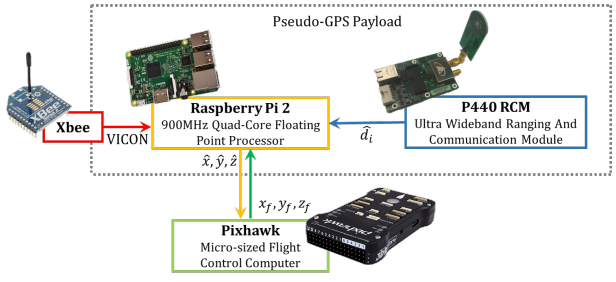

Figure 2: Main modules of the testbed.

In our experiment, the UWB transceivers in use are the P440 modules from Time Domain. There are four P440s acting as the anchor nodes and another four are mounted on the UAVs to range to the anchors. Each UAV has a high-level control board who hosts and directs the UWB modules to collect the distances to anchors.

The high level board then computes the location and sends the position estimates to the low-level board which fuses this data with its high-rate IMU and barometer sensors. The output is then sent back to the high level board at a lower rate for logging and off-line analysis.

3.2 Indoor autonomous flight

The indoor flight test was carried out using VICON as ground truth. 4 UAVs were loaded with predefined set points and set times. The set points were chosen to be on a $4m \times 4m$ square for the outer loop and an $2.5 \times 2.5m$ square for the inner loop. The anchors are positioned at the corners of a $6m \times 6m$ area with varying heights. After sending an RC signal to trigger the autonomous flight mode, all UAVs will register the trigger time. Each will then load the first set point from memory, fly to this point, stay there until the local clock exceeds the set time and then proceed to the next point. The ground truth was processed by a central computer and sent to each UAV to be stored locally. In Figure 5, all of the flight data were synchronized to the VICON system time by using time stamps and assuming that the clocks of P440, VICON, high-level and low-level controller boards have negligible drift during the experiment period.

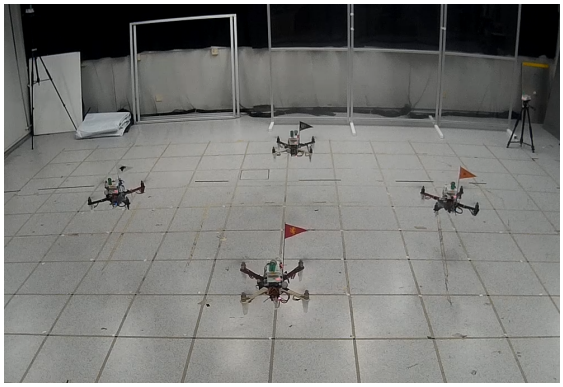

Figure 3: 4 UAVs at the closest set points (roughly $1.75m$ apart from each other) in the indoor test. Each square on the floor has an area of $0.5m \times 0.5m$

In Figure 4, it can be seen that x and y estimates from UWB-based localization methods have consistent accuracy across all UAVs, as the same level of error can be seen at the same area on the four plots. All of the RMS errors of the implemented methods are summarized in Table 1. For 2D localization analysis, which we only compare x and y estimates with VICON data, among all methods and UAVs, the maximum RMS error is 0.162m. The NLR method also notably

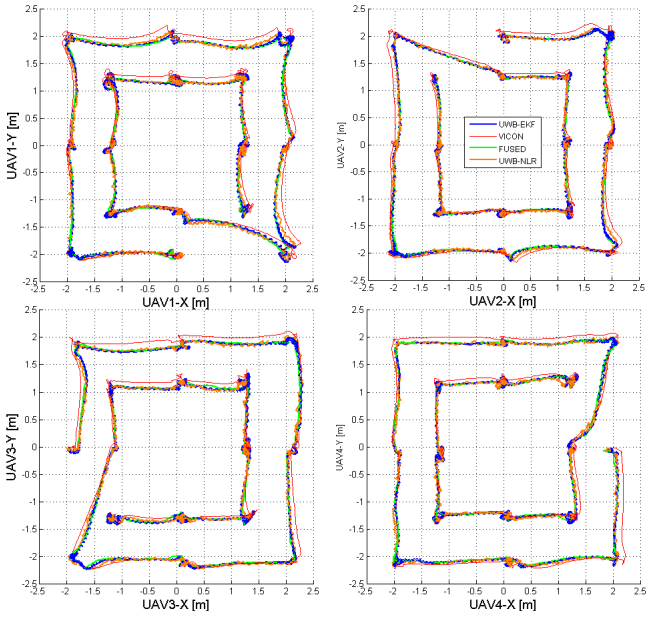

Figure 4: x and y estimates of 4 UAVs with VICON ground truth in red. EKF estimate is in blue, NLR orange and the EKF estimate fused with Pixhawk's sensors in green.

shows a slightly smaller error on all of the UAVs. The fused estimate appears to have a slightly larger error in comparison with the raw UWB-EKF estimate which can be explained because the data were down-sampled from 250Hz to 20Hz to ease the logging and analysis. The latency from transmitting the UWB-based estimate to px4 is also a factor of this larger error. This error is negligible for 2D movements of the UAVs.

For altitude estimate, the error is obviously more prominent than x and y estimates. The desired altitude is set at 1m for all the UAVs. The maximum RMS error is 0.354m from the fused estimate. It can be seen in Figure 5 that the error is location-wise consistent: any UAV will read the same altitude at the same place and the error level can act like an offset at that place. This effect thus does not forestall stabilization of altitude. On the other hand, it can be seen that the NLR altitude estimate has very large variation. It can suddenly "flip" at some points which can be explained as the algorithm converges to a value that is not globally optimal due to the imperfection of the distance as well as error in the declared position of the anchors.

Incompatible accuracy in the altitude estimate is a persistent issue in similar works [13]. For future work we may consider fusing external IMU data with the EKF/NLR estimates on the high level board to mitigate their respective shortcomings. If the application requires 2D localization only, x and y estimates of either method can be made into use. For 3D localization, the experiment shows that the EKF altitude estimate fused with Pixhawk's sensor is stable enough for flying. Though there is still significant altitude error, it is consistent among all UAVs and locations. This characteristics may offer a possibility for improvement in future works, for example the altitude error can be mapped out and compensated based on the 2D localization estimates.

UAV	EKF		FUSED		NLR	
	XY	**Z**	XY	**Z**	XY	**Z**
1	0.124	**0.353**	0.127	**0.354**	0.119	**0.346**
2	0.111	**0.343**	0.126	**0.349**	0.105	**0.327**
3	0.150	**0.335**	0.162	**0.338**	0.144	**0.336**
4	0.107	**0.348**	0.110	**0.347**	0.102	**0.341**

Table 1: RMS error of the indoor flight test for each UAV (all data are in m).

4 Conclusion

In this paper, the capability of the proposed technique for multi-UAV localization has been demonstrated by supporting up to 4 UAVs. A secondary localization method based on NLR was also investigated to work together with the EKF for initialization and reset. For 2D localization, both Extended Kalman Filter and Nonlinear Regression methods can provide sufficiently accurate estimate for UAVs. When altitude estimate is taken into account, the EKF estimate has been successfully tested for flying while the NLR method still has some problems with its altitude estimate. Nevertheless, the two methods are relatively independent and can both run in parallel to supplement each other.

References

[1] F. Nex and F. Remondino, "Uav for 3d mapping applications: a review," *Applied Geomatics*, vol. 6, no. 1, pp. 1–15, 2014.

[2] "Application of robotics in onshore oil and gas industry—a review part i," *Robotics and Autonomous Systems*, vol. 75, Part B, pp. 490 – 507, 2016.

[3] C. Zhang and J. M. Kovacs, "The application of small unmanned aerial systems for precision agriculture: a review," *Precision Agriculture*, vol. 13, no. 6, pp. 693–712, 2012.

[4] J. Qi, D. Song, H. Shang, N. Wang, C. Hua, C. Wu, X. Qi, and J. Han, "Search and rescue rotary-wing uav and its application to the lushan ms 7.0 earthquake," *Journal of Field Robotics*, vol. 33, no. 3, pp. 290–321, 2016.

[5] Y. Khosiawan and I. Nielsen, "A system of uav application in indoor environment," *Production & Manufacturing Research*, vol. 4, no. 1, pp. 2–22, 2016.

[6] B. Zhu, J. Xu, A. H. B. Zaini, and L. Xie, "A three-dimensional integrated guidance law for rotary uav interception," in *2016 12th IEEE International Confer-*

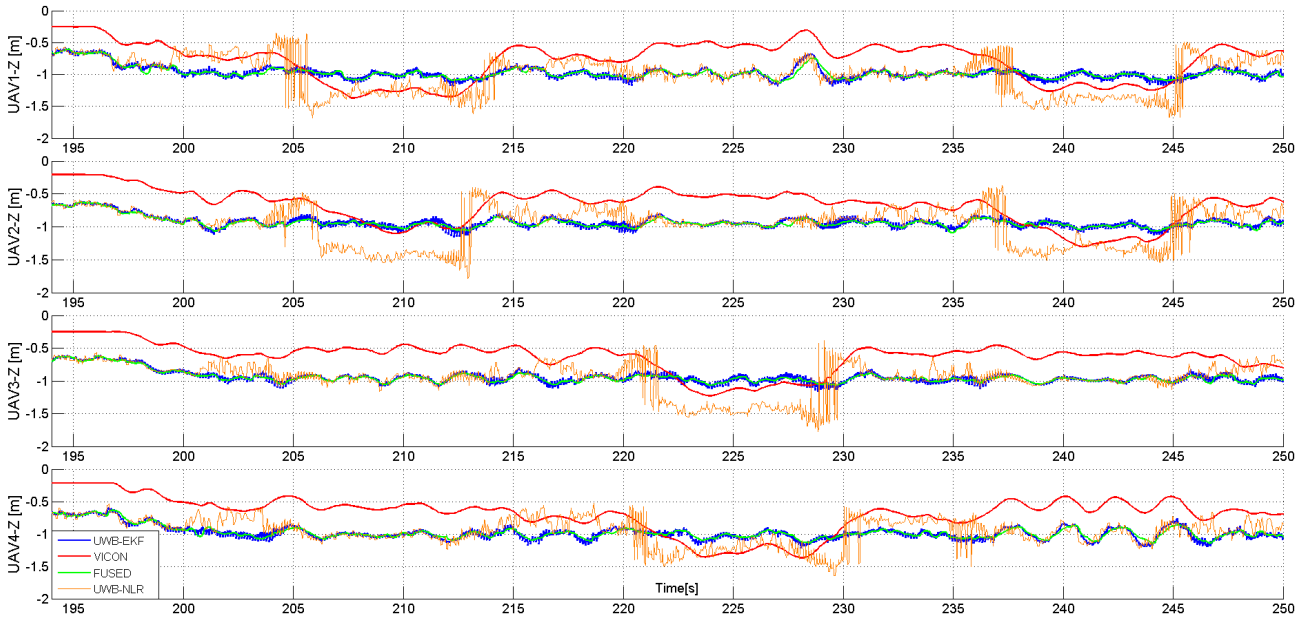

Figure 5: Concurrent z estimates of 4 UAVs for half of the flight test. The altitude is represented in NED convention, thus all values are negative. The ground-level measurement from UWB is 0.6m and the VICON height is about 0.25m. The UAVs take off at the 196 second and as the set value for altitude is -1m, the UWB-EKF and the fused estimates are maintained around this value by the control loop. For most of the time, the VICON altitude is lower than UWB-based estimate. However, for UAV 1, it can be seen that the error changes sign as it gets close to anchor 4 and anchor 2 at 206s to 214s and 238s to 245s respectively. These anchors are the ones with lower heights. The "flipping" effect of NLR estimates also occurs in these locations. The same effects are also observed on other UAVs.

ence on Control and Automation (ICCA), June 2016, pp. 726–731.

[7] A. Kushleyev, D. Mellinger, C. Powers, and V. Kumar, "Towards a swarm of agile micro quadrotors," *Autonomous Robots*, vol. 35, no. 4, pp. 287–300, 2013.

[8] 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*, 2014.

[9] B. Balaji, J. Xu, A. Nwokafor, R. Gupta, and Y. Agarwal, "Sentinel: occupancy based hvac actuation using existing wifi infrastructure within commercial buildings," in *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2013, p. 17.

[10] B. Benjamin, G. Erinc, and S. Carpin, "Real-time wifi localization of heterogeneous robot teams using an online random forest," *Autonomous Robots*, vol. 39, no. 2, pp. 155–167, 2015.

[11] Y. Sun, M. Liu, and M. Q. H. Meng, "Wifi signal strength-based robot indoor localization," in *Information and Automation (ICIA), 2014 IEEE International Conference on*, July 2014, pp. 250–256.

[12] K. Guo, Z. Qiu, C. Miao, A. H. Zaini, C.-L. Chen, W. Meng, and L. Xie, "Ultra-wideband-based localization for quadcopter navigation," *Unmanned Systems*, vol. 4, no. 01, pp. 23–34, 2016.

[13] A. Ledergerber, M. Hamer, and R. D'Andrea, "A robot self-localization system using one-way ultra-wideband communication," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, Sept 2015, pp. 3131–3137.

[14] Time Domain PulsON 440. [Online]. Available: http://www.timedomain.com/products/pulson-440/

[15] 3DR Pixhawk. [Online]. Available: https://store.3dr.com/products/3dr-pixhawk

[16] PX4 Development Guide. [Online]. Available: http://dev.px4.io/