Towards Autonomous Indoor Flights Using Wireless Sensor Network Based Localization

Juergen Eckert*, Reinhard German* and Falko Dressler†
*Dept. of Computer Science, University of Erlangen, Germany
†Institute of Computer Science, University of Innsbruck, Austria
{juergen.eckert,german}@cs.fau.de, falko.dressler@uibk.ac.at

***Abstract*—Indoor hovering objects such as quadrotors need to continuously be controlled to hold their position in space. For real autonomous flight of these copters, the necessary position control including all related information transfers have to be provided in a fully decentralized and autonomous manner. We discuss challenges related to flight control based on our Autonomous Localization Framework (ALF), which provides scalable and decentralized localization in GPS-denied areas. Using a sensor network based on the IEEE 802.15.4 communication protocol, continuous position maintenance is feasible but, unfortunately, in no way stable. Therefore, we introduce a sensor array, which reduces the system dynamics and allows a robust position control of the platform.**

I. Introduction

Six degrees of freedom need to be controlled continuously for motion in the three-dimensional space. The usual fall back mechanism for unsolvable problems in the field of robotics is simply to stop. This, of course, can not be applied during the flight of an Unmanned Aerial Vehicle (UAV). In this paper, we are focusing on a UAV subclass, the so called vertical take-off and landing (VTOL) devices. They are most suited for our indoor usage scenario. Here, however, not much space for maneuvering is available, which forces the platform to mostly remain in the most unstable condition: hovering. Serious self-caused air turbulences additionally affect the flight stability.

Copters (classical, coaxial, quad, etc) are basically controlled the same way. The forward/back and left/right axes get indirectly controlled by pitch and roll, respectively. Applying a constant angle, different from zero, to an axis will cause the platform to move into this direction with increasing speed. This reduces the amount of independent degrees to four: pitch, roll, yaw, and altitude (also indirectly controlled by thrust). Even if it would be possible to perfectly control the pitch and roll axes, the system would still move according to Newton's first law of motion. Moreover, each disturbance in the air results in an on-board unpredictable and unmeasurable movement. To counteract such involuntary movement, a guidance system is needed to continuously control the position – otherwise the system would quickly crash into walls or other obstacles.

Indoor position controlling approaches for flying objects based using localization systems [1] or Simultaneous Localization and Mapping (SLAM) [2] already exist. However, they either depend on a highly accurate and expensive localization system that needs to be deployed manually; or they need a high amount of (mostly off-board) computing power. In the scope of this paper, we rely on our ALF framework, which, to the best of our knowledge, is the first platform fulfilling all the requirements for a real autonomous indoor flying system; including the necessary sensor deployment and information acquisition.

II. Autonomous Localization Framework

Our Autonomous Localization Framework (ALF) has been designed to fulfill the task of providing a zero-effort accurate localization system for autonomously flying robots [3]. However, it is not exclusively restricted to this. The framework satisfies several requirements, low cost being probably the most important one. This is the main reason why we not built-up a SLAM approach (usually expensive laser distance scanners would be needed). Furthermore, the framework should be able to operate with minimal computational power.

ALF has been designed as a decentralized system without any synchronization or global knowledge and a finite amount of energy. The typical operation scenario is depicted in Figure 1. Sensor nodes need to be capable of detecting and communicating with their direct neighbors; no special routing information or topology is required. For localization, at least the distance to the neighbors needs to be measurable. As we desire an accurate localization system, we do not rely on rather vague Received Signal Strength Indicator (RSSI) measurements (although it would be possible [4]). Our ground platform [5] is equipped with an ultrasound based Time of Flight (ToF) measuring hardware. It reports distances with an observational error of less than ± 2 cm. A byproduct is a rough estimation of the angle to a neighbor ($\pm 45°$), which is sub sequentially also used but not required. The mobility is a key feature of a truly autonomous system but not required for ALF.

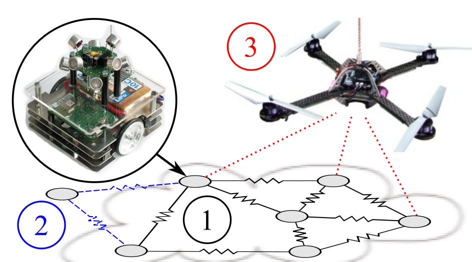


Fig. 1. Autonomous Localization Framework scenario [3]

First (step 1 in Figure 1) each node needs to find a “convenient” place, which is a trade-off between area coverage, accuracy, and cost. As no global knowledge exists, each node needs to find this independently from the states of its neighbors. A local decision table based approach is used. Afterwards (step 2), the node localizes itself according to an already existing reference grid [6]. Using this information as an initial position, the node joins the reference network and enters the maintenance and correction process, which is based on our Advanced Mass-Spring-Relaxation (advMSR) approach [7]. Again, only local information are used.

In this process, as typical for indoor scenarios, many Non Line of Sight (NLOS) measurements may happen. In order not to corrupt the system those need to be detected and blacklisted [3]. The algorithm was designed so that it reaches convergence. At any point in time, nodes are allowed to join or leave the network. Reason for leaving might be unsatisfied system parameters or the depletion of the battery. Finally (step 3), the nodes and the generated position information are used as a reference grid for providing a localization support for customers [6], such as the autonomously flying copters. The trilaturation-based scheme was specially designed to fulfill real-time requirements.

III. System Accuracy

We evaluated the system performance in a lab experiment: Nine nodes were placed on the floor to build up a coordinate system. We simulated a customer by a sensing device on a stick (fixed altitude) mounted on a toy train. Two different experiments were evaluated: Line of Sight (LOS) and NLOS measurements. For the latter one, obstacles were placed into the testbed. These obstacles introduced incorrect NLOS measurements with a probability of $P_{\text{NLOS}} \approx 30\,\%$ (this value is commonly reported in the literature [8]).

Due to the lack of a more accurate localization system, the error of the generated absolute position was not directly measurable. However, Figure 2 shows the altitude error of the experiment. All the results are shown in form of boxplots: The thick line represents the median, the rectangular boxes contains 50 % of the measurements, the boundaries are indicating the 25 % and 75 % quantiles. The circles show statistical outliers. The dashed red lines are indicating the hardware sensing accuracy of ±2 cm. It can be seen that at least 50 % of the measurements of both experiments are within that range. A maximum error of ±20 cm and ±40 cm can be given for the LOS and for the NLOS experiment, respectively.

IV. Communication

As stated before, ALF relies on simple direct neighbor communication. When a customer wants to know its position it triggers an active ultra sound measurement. The sound pulse is received by the ground nodes. This flight duration needs to be reported back, using the radio, to conclude to a position.

We used SunSpot nodes in our experiments. These systems operate an IEEE 802.15.4 [9] compatible Chipcon CC2420 interface for wireless communication. The primary design goals

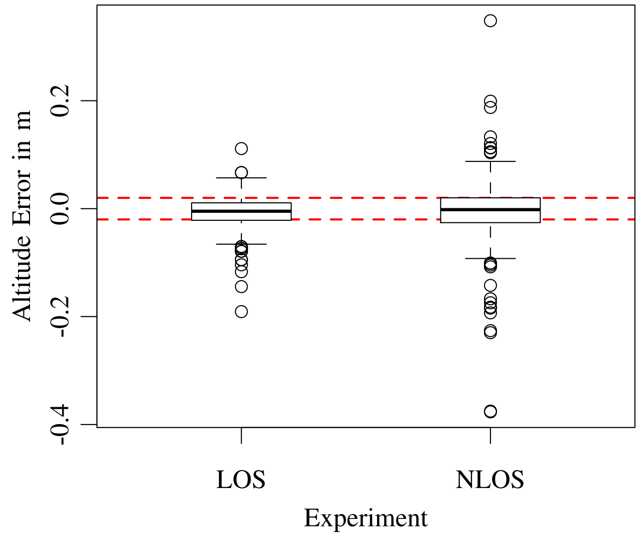


Fig. 2. Customer localization accuracy

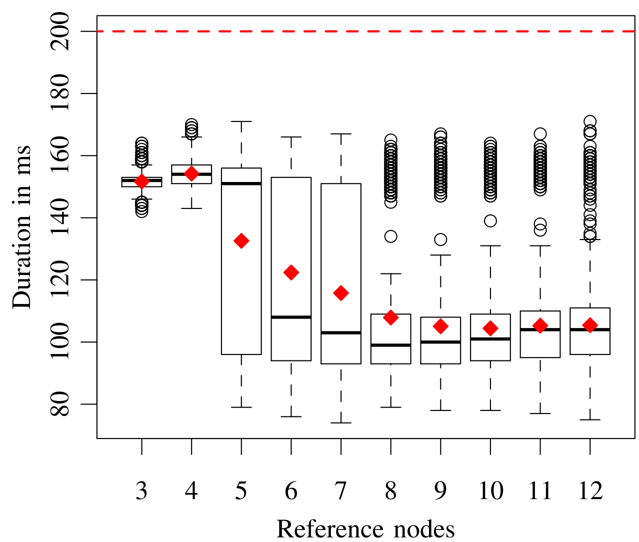


Fig. 3. Communication Latency

of this interface were energy efficiency and scalability. High latency and low bit rates were accepted, but low-latency support has been worked on for this protocol [10], [11], but completely different low latency MAC approaches might be even more suitable [12]). However, we intended to stick to the default setup to show the general applicability of ALF. We showed that a custom agent based application layer protocol can collect the information in time [6]. It performs significantly better than a pure broadcast based approach.

In an experiment we placed three to twelve ground nodes around one customer. We measured the required time for collecting enough measurement tuples to robustly determine the customer’s position. The results are plotted in Figure 3. Three is the minimum number of reference nodes required for

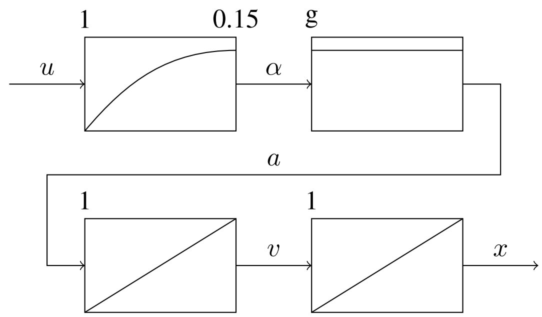


Fig. 4. System model of a copter

trilaturation. The red dashed line shows the upper border of a time slot, which must not be exceeded by all means. The red rhombi are showing the mean values. It can be seen that the slot boarder is never crossed and that the duration significantly decreases for five and more nodes. This is because we stop the collection process prematurely, because five tuples are enough to robustly compute a position [6]. On the average, it takes 100 ms to 160 ms to complete the data collection. No outliers exceeded the 200 ms border.

V. Control Theory

From a control theory point of view, we approximated the system of the roll and nick axes as depicted in Figure 4. Both axes can be controlled independently from each other. The system input u is tilt information, given in radian. The platform tries to reach this value by tilting. Due to inertia this takes some time (150 ms on our platform). We approximated this using a first order lag element. Now, considering that it hovers in the air $|F_{weight\ force}| = |F_{lift}|$ and only very small angles are applied, it is sufficient to multiply the previous outcome with the gravitational acceleration g to get an acceleration into the direction of the axis. By double integrating the value a position x is obtained, which, can be measured by ALF.

This system could be stabilized by a classical PD controller. Due to the previously mentioned communication latency the control frequency is fairly low (5 Hz). However, results reported in the literature indicate that under optimal conditions stabilization is still doable [1]. By moving from the classical continuous to a time discrete perspective and by utilizing a state-space position controller, we tried to overcome the tremendous delay time of the communication channel. Still, the total delay time of up to 195 ms (including measurement, information transport and computation) made the system very susceptible.

The slightest measurement or communication error results in an overshoot, potentially followed by a crash. We see two possible solutions to this problem: (a) A significant reduction of the communication latency (e.g., a different MAC layer [12]). (b) A reduction of the system dynamic of the customer. We chose to rely on the second option following our discussions ob the communication hardware options.

VI. Optical Sensor

In order to reduce system dynamics and to have a temporary fall back mechanism, we currently experiment with an optical flow sensor. These sensors are very low cost, low power, and light-weight, typically also installed in computer mice, with an adjusted lens. They have already been identified as suitable for this application [13]. Usually this sensor is facing downwards to the floor and controls the speed on the forward/back and left/right axes. Before the velocity control can do its work, two more things are required: The altitude, which is rather simple to obtain, is needed to derive the moved distance.

More important and but even more difficult to estimate is the nick and roll angle on the copter. As the platform tilts to move and the sensor is rigidly mounted to the frame, it detects a relatively high movement into the inverse direction of the actual movement (see Figure 5(b)). This is counter-productive and needs to be compensated by the knowledge of the current slant. However, this is very time critical and the angles need to be very accurate, especially for large altitudes. Therefore, we are planing to exploit a unique feature of indoor flights: Rooms not only have floors but also ceilings. Thus, we use one down and one up facing sensor. This way the angle of the platform can be neglected.

Figure 5(a) depicts one axis of a quadrotor hovering at time t. The current tilting α_t usually needs to be known, but the absolute value is not important. It additionally knows the distances to the floor d_f and to the ceiling d_c. If the room height h is known, $d_c = h - d_f$ needs not to be measured. In Figure 5(b), the quadrotor has a different tilting α_{t+1} and has also moved a distance Δx. However, the up facing sensors reports the distance Δx_c and the down facing one reports Δx_f. *NB*: For improved visualization the values are already normalized according to the resolution of the sensor and the distance to the surface.

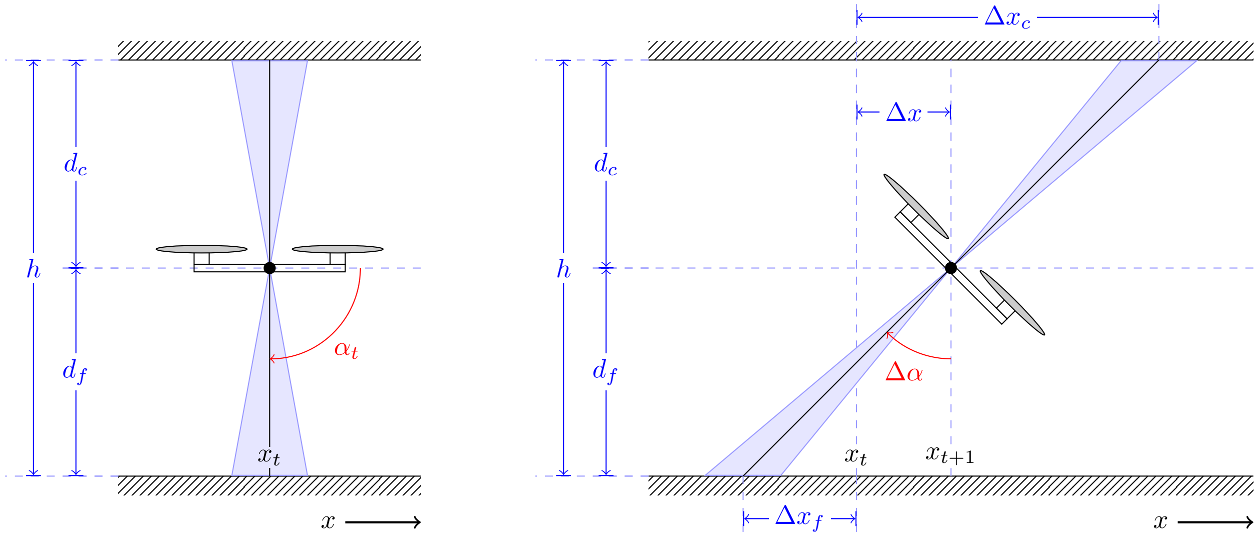
The common approach is now just to use the down facing sensor, which might report a movement into the negative direction as depicted in Figure 5(b). An inclination correction can be calculated as

$$\Delta x = \Delta x_f - d_f \tan \Delta\alpha, \qquad (1)$$

where $\Delta\alpha = \alpha_{t+1} - \alpha_t$.

It can be seen that for large d_f accurate tilting information is mandatory and needs to be measured with a high resolution. Unfortunately, tilting information in general is very hard to obtain on such dynamic systems. The relative distance movement Δx is normally in the range of a few millimeters (due to high sampling rates), whereas the average flight height d_f is usually in the range of 1 m to 2 m. That clearly shows the error-proneness of this approach. *NB*: It is assumed that the platform is only doing minor inclinations, i.e., α_t is always nearly orthogonal to the floor (see Figure 5(a)).

We are about to circumvent the evaluation of the tilting angle by placing an additional sensor on top of the platform. Using the intercept theorem gives us the following angle free



(a) Time t

(b) Time $t+1$

Fig. 5. Quadrotor movement

equation for the platform movement Δx:

$$\frac{\Delta x_c - \Delta x}{\Delta x - \Delta x_f} = \frac{d_c}{d_f} \tag{2}$$

In case of an undetectable surface (no contrasts) on one sensor, the fall back to the others in combination with the standard angle based approach (Equation 1) provides a second safety stage.

VII. Conclusion

We briefly introduced a scalable, real-time capable localization framework using sensor network technology. It is fully aware of scenario driven limitations such as energy and computing power boundedness. Additionally it does not require synchronization, routing information or any other global knowledge. As shown it is mostly ($P \approx 50\,\%$) as accurate as the utilized sensing hardware.

However, due to measurement outliers in combination with the communication time lag we were not yet able to continuously stabilize the quadrotor. After reducing the system dynamic of the copter by the use of the optical flow sensor, the Autonomous Localization Framework will be able to stabilize the platform permanently.

References

[1] D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger, and D. Rus, "Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz," in *IEEE International Conference on Robotics and Automation (ICRA 2007)*. Roma, Italy: IEEE, April 2007, pp. 361 –366.

[2] S. Hochdorfer and C. Schlegel, "6 DoF SLAM using a ToF Camera: The Challenge of a Continuously Growing Number of Landmarks," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2010)*. Taipei, Twaiwan: IEEE, October 2010, pp. 3981–3986.

[3] J. Eckert, R. German, and F. Dressler, "ALF: An Autonomous Localization Framework for Self-Localization in Indoor Environments," in *7th IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS 2011)*. Barcelona, Spain: IEEE, June 2011.

[4] K. Chintalapudi, A. P. Iyer, and V. N. Padmanabhan, "Indoor Localization Without the Pain," in *16th ACM International Conference on Mobile Computing and Networking (MobiCom 2010)*. Chicago, IL: ACM, September 2010, pp. 173–184.

[5] J. Eckert, K. Koeker, P. Caliebe, F. Dressler, and R. German, "Self-localization Capable Mobile Sensor Nodes," in *IEEE International Conference on Technologies for Practical Robot Applications (TePRA 2009)*. Woburn, MA: IEEE, November 2009, pp. 224–229.

[6] J. Eckert, R. German, and F. Dressler, "An Indoor Localization Framework for Four-rotor Flying Robots Using Low-power Sensor Nodes," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 2, pp. 336–344, February 2011.

[7] J. Eckert, F. Villanueva, R. German, and F. Dressler, "Distributed Mass-Spring-Relaxation for Anchor-free Self-localization in Sensor and Actor Networks," in *20th International Conference on Computer Communication Networks (ICCCN 2011)*. Maui, HI: IEEE, July/August 2011.

[8] R. Casas, A. Marco, J. Guerrero, and J. Falco, "Robust Estimator for Non-Line-of-Sight Error Mitigation in Indoor Localization," *EURASIP Journal on Applied Signal Processing*, vol. 2006, pp. 1–8, 2006.

[9] "Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low Rate Wireless Personal Area Networks (WPANs)," IEEE, IEEE Standard 802.15.4-2006, 2006.

[10] F. Chen, R. German, and F. Dressler, "Towards IEEE 802.15.4e: A Study of Performance Aspects," in *8th IEEE International Conference on Pervasive Computing and Communications (PERCOM 2010), 2nd IEEE International Workshop on Information Quality and Quality of Service for Pervasive Computing (IQ2S 2010)*. Mannheim, Germany: IEEE, March 2010, pp. 68–73.

[11] A. Koubaa, M. Alves, B. Nefzi, and Y.-Q. Song, "Improving the IEEE 802.15.4 Slotted CSMA/CA MAC for Time-Critical Events in Wireless Sensor Networks," in *5th International Workshop on Real-Time Networks (RTN 2006)*, Dresden, Germany, July 2006.

[12] D. Carlson and A. Terzis, "Flip-MAC: A Density-Adaptive Contention-Reduction Protocol for Efficient Any-to-One Communication," in *7th IEEE/ACM International Conference on Distributed Computing in Sensor Systems (DCOSS 2011)*. Barcelona, Spain: IEEE, June 2011.

[13] S. Lange, N. Sunderhauf, and P. Protzel, "A vision based onboard approach for landing and position control of an autonomous multirotor UAV in GPS-denied environments," in *IEEE International Conference on Robotics and Automation (ICRA 2009)*, Kobe, Japan, June 2009, pp. 1–6.