

# Efficient Probabilistic Localization for Autonomous Indoor Airships using Sonar, Air Flow, and IMU Sensors

Jörg Müller

Wolfram Burgard

*Department of Computer Science, University of Freiburg, Germany, {muellerj, burgard}@informatik.uni-freiburg.de*

## Abstract

In recent years, autonomous miniature airships have gained increased interest in the robotics community. Whereas their advantage lies in their abilities to move safely and to hover for extended periods of time, they at the same time are challenging as their payload is strictly limited and as their complex second-order kinematics makes the prediction of their pose and velocity through physical simulation difficult and imprecise. In this paper, we consider the problem of particle filter based online localization for a miniature blimp with lightweight ultrasound and air flow sensors as well as an IMU. We present probabilistic models dedicated to the special characteristics of the miniature and lightweight sensors applied on our blimp. Furthermore, we introduce an efficient odometry motion model based on the measurements of air flow sensors and an IMU which is less computationally demanding compared to the standard physical simulation-based control motion model. In experiments with a real blimp in a complex indoor environment, our approach has proven to allow accurate and reliable online localization of a miniature blimp and requires an order of magnitude fewer particles compared to the localization based on the standard control motion model. Furthermore, we demonstrate the substantial improvements in terms of localization accuracy when taking into account the temporal correlation of the air flow measurements in our novel odometry motion model.

*keywords:* blimp, localization, autonomous navigation, sonar, IMU

## 1 Introduction

Miniature airships as autonomous mobile systems for indoor navigation have several desirable features as they can navigate safely and with low noise in their three-dimensional environments. At the same time, their low power consumption makes them well-suited for long-term operation tasks. These features facilitate a wide range of applications including environmental monitoring, surveillance, disaster scenarios, communication, and advertising even in the presence of people, e.g., in public spaces. To efficiently navigate in such applications, the robots need to be able to perform several tasks such as path planning and closed loop control that additionally require the ability to accurately estimate the position. In the context of miniature blimps, however, localization is challenging as such platforms are restricted to lightweight and small on-board sensors like miniature sonars or microelectromechanical systems (MEMS) based IMUs or thermal air flow sensors [29, 30, 44]. Additionally, the

motion prediction of airships typically relies on a physical simulation of the accelerations acting on the vehicle, e.g., thrust and drag, which is computationally expensive and induces a high amount of uncertainty, even when adaptive motion models [28] are used.

In this paper, we consider the problem of online localization for a miniature blimp in indoor environments through probabilistic state estimation with a particle filter. Our blimp, which is depicted in Fig. 8, has an effective payload of approximately 150 grams and is equipped with five ultrasound sensors, three thermal air flow sensors as well as an IMU for navigation. Due to their wide opening angle the sonar sensors typically provide highly ambiguous information. We therefore present a sensor model that is dedicated to the characteristics of small-sized ultrasound sensors. We furthermore present a novel sensor model for the thermal air flow sensors and develop an efficient flow odometry motion model for accurate motion prediction in the particle filter. In contrast to the standard simulation-based control motion model, our odometry motion model requires an order of magnitude fewer particles for localization and is less computationally demanding because it does not rely on complex physical simulations. Therefore it enables accurate online localization. We present the results obtained in experiments carried out with a real robotic blimp in a complex indoor environment. The results demonstrate that our approach to online localization outperforms the standard particle filter localization based on the control motion model applied in many state-of-the-art autonomous blimp navigation systems [13, 21, 45, 28].

This paper is organized as follows. After formulating the problem in the following Section and a brief description of particle filter localization in Section 3 we present the novel probabilistic sensor models in Section 4 to 6. We then summarize the standard airship control motion model in Section 7 and introduce our odometry motion model based on air flow sensors in Section 8. In Section 9 we demonstrate the capabilities of our approach in experiments with a real robotic blimp. In Section 10 we then present a detailed discussion of related work.

## 2 Problem Formulation

In this paper we consider the problem of recursively estimating the state  $\mathbf{x}$  of a miniature airship. As the airship has second-order system dynamics and can move freely in three-dimensional space, we define its state as

$$\mathbf{x} = [\mathbf{p}^T, \mathbf{q}^T, \mathbf{v}^T, \boldsymbol{\omega}^T]^T \quad (1)$$

consisting of the position  $\mathbf{p} = [x, y, z]^T$ , the orientation  $\mathbf{q} = [q_0, q_1, q_2, q_3]^T$  represented by a unit quaternion [8], the translational velocity  $\mathbf{v} = [v_x, v_y, v_z]^T$ , and the angular velocity  $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^T$ . Additionally, we define the translational acceleration  $\mathbf{a} = \dot{\mathbf{v}} = [a_x, a_y, a_z]^T$  and the angular acceleration  $\boldsymbol{\alpha} = \dot{\boldsymbol{\omega}} = [\alpha_x, \alpha_y, \alpha_z]^T$ . The position and orientation are expressed in the global frame of reference  $\mathcal{F}_g$  (with the  $z$ -axis pointing upwards). The velocities, accelerations, forces, and torques are expressed in the body-fixed frame of reference  $\mathcal{F}_b$ . The origin of the body-fixed frame  $\mathcal{F}_b$  is the center of buoyancy of the airship with the  $x$ -axis pointing forward and the  $z$ -axis pointing upwards. For evaluation and model learning purposes we define the ground truth state

$$\mathbf{x}^* = [\mathbf{p}^{*T}, \mathbf{q}^{*T}, \mathbf{v}^{*T}, \boldsymbol{\omega}^{*T}, \mathbf{a}^{*T}, \boldsymbol{\alpha}^{*T}]^T, \quad (2)$$

which also includes the translational and angular acceleration of the robot.

Throughout this paper we assume that the trajectory  $\mathbf{x}_{1:T} = \mathbf{x}_1, \dots, \mathbf{x}_T$  is discretized into  $T$  time steps. The input data for state estimation are the applied control commands  $\mathbf{u}_{1:T}$  and the sensor data  $\mathbf{z}_{1:T}$ . In particular, the sensor data  $\mathbf{z} = \{z_{S,1}, \dots, z_{S,\ell}, z_{F,1}, \dots, z_{F,m}, \mathbf{z}_I\}$  of our airship consists of the measurements of  $\ell$  sonar sensors,  $m$  air flow sensors, and an IMU indicated by the indices S, F, and I, respectively.

### 3 Vanilla Particle Filter Localization

For solving the localization problem we follow the probabilistic approach and apply the recursive Bayesian filtering scheme [39]. The key idea of this approach is to recursively update the posterior probability density  $p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t})$  of the state  $\mathbf{x}_t$  at time  $t$  conditioned on all sensor data  $\mathbf{z}_{1:t}$  and control commands  $\mathbf{u}_{1:t}$  up to time  $t$ . For the implementation of the Bayes filtering scheme we apply particle filtering [9] which approximates the posterior  $p(\mathbf{x}_t | \mathbf{z}_{1:t}, \mathbf{u}_{1:t})$  by a set of  $N$  particles. Each particle corresponds to a state hypothesis  $\mathbf{x}^{[i]}$  weighted by the so-called importance weight  $w^{[i]}$ . The recursive update of the particle filter is performed according to the following three steps [7, 9, 14]:

1. *Sampling*: In the *prediction step*, we propagate each particle by drawing a successor state from the proposal distribution. As it is common practice in mobile robot localization, we use the probabilistic motion model  $p(\mathbf{x}_t^{[i]} | \mathbf{x}_{t-1}^{[i]}, \mathbf{u}_t)$  given the control command  $\mathbf{u}_t$  as the proposal distribution.
2. *Importance Weighting*: In the *correction step*, the importance weight of each particle is updated according to the importance sampling principle. As we use the motion model as the proposal distribution, the weight is updated by  $w_t^{[i]} \propto w_{t-1}^{[i]} p(\mathbf{z}_t | \mathbf{x}_t^{[i]})$  according to the sensor model.
3. *Resampling*: In the *resampling step*, we draw a new generation of particles (with replacement) such that each particle is selected with a probability that is proportional to its weight.

For an efficient implementation of the particle filter, we use low-variance resampling [39] and omit the resampling step until the effective number of particles [24] drops below half the number of particles.

Under the Markov assumption, the measurements of the individual sensors are conditionally independent given the state of the system [39], so that we can express the measurement likelihood as

$$p(\mathbf{z} | \mathbf{x}) = \left( \prod_{i=1}^{\ell} p(z_{S,i} | \mathbf{x}) \right) \left( \prod_{j=1}^m p(z_{F,j} | \mathbf{x}) \right) p(\mathbf{z}_I | \mathbf{x}) \quad (3)$$

for our miniature blimp equipped with sonar sensors, air flow sensors, and an IMU. In the following four sections of this paper we describe the probabilistic models for the corresponding sensors and for the motion of miniature indoor airships. Furthermore, we introduce our novel air flow odometry motion model for computing efficient and accurate proposal distributions in the particle filter.

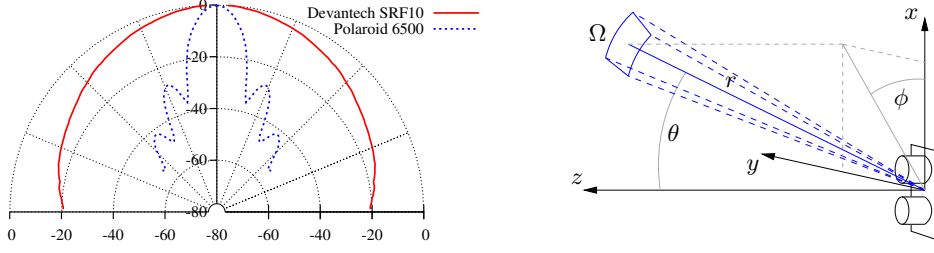


Figure 1: Left: The intensity pattern of the Devantech SRF10 miniature sonar sensor compared to the one of the popular Polaroid 6500 sensor. The units are decibel and are normalized to the maximum intensity. Right: The spherical coordinate system used for modeling the sensor behavior. An object is seen by the sensor in distance  $r$ , azimuth angle  $\phi$ , and zenith angle  $\theta$ . In this way, the dihedral angle  $\Omega$  is covered.

## 4 Sonar Sensor Model

The probabilistic sensor model  $p(\mathbf{z} \mid \mathbf{x})$  plays a crucial role in the correction step of the particle filter and its proper design is essential for accurate state estimates. For sonar sensors it defines the likelihood of the scalar distance measurement  $r \in \mathbb{R}$  given the state  $\mathbf{x}$  of the system including the information about the environment typically given by a map. In the following, we summarize our model introduced in [30] which physically models the characteristics of small-size sonar sensors with large opening angles.

A measurement starts with the generation and transmission of an ultrasound signal. The signal propagates spherically through the space and, after it got reflected by objects in the environment, the corresponding echo is typically detected by the receiver unit of the sensor. As soon as the received and amplified signal exceeds a given threshold, the measurement procedure is terminated and the distance  $r = \frac{c\Delta t}{2}$  is calculated based on the time of flight  $\Delta t$  and the constant velocity  $c$  of sound in air.

For very small transmitters with a diameter in the same order of magnitude as the wavelength, the signal is hardly focused and has lower intensity at its boundary area. In our approach, we model this behavior by calculating a probability distribution of triggering a measurement by modeling the received signal over the elapsed time  $\Delta t$ . The emitted signal intensity (power per area)  $I$  is depicted in Fig. 1 and depends on the zenith angle  $\theta$ . Due to the symmetry of ultrasonic membranes it does not depend on the azimuth angle  $\phi$ .

We determine the set of objects potentially reflecting the signal by a discrete set of ray-casting operations on the map such that the entire visible hemisphere is covered. The incident signal power of an object, which is seen by the sensor in distance  $r$  and zenith angle  $\theta$  and which corresponds to the dihedral angle  $\Omega$ , is

$$P(r, \theta, \Omega) = I(\theta) D(r) \Omega . \quad (4)$$

In contrast to Moravec [26], we physically model the damping  $D(r)$  of ultrasound in air and the intensity change with  $\frac{1}{r^2}$  which is taken into account in  $\Omega$ .

A proportion  $P_R(r)$  of this signal power of all objects in distance  $r$  is reflected back to the sensor and it can be approximated by a Gaussian

$$p(P_R(r) \mid \mathbf{x}) \approx \mathcal{N}((P_R(r); \alpha P_{\max}(r, \mathbf{x}), \alpha (1 - \alpha) P_{\max}(r, \mathbf{x})) . \quad (5)$$

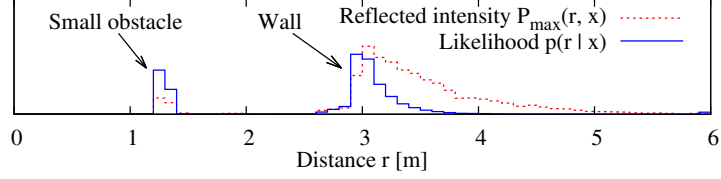


Figure 2: An example of the maximum power  $P_{\max}$  that would be received if all objects in distance  $r$  reflect the ultrasound signal towards the receiver and the corresponding measurement likelihood.

The mean and variance of this Gaussian depend on the reflection proportion  $\alpha$  and the maximum power  $P_{\max}$  that would be received if all objects in distance  $r$  reflect the ultrasound signal towards the receiver. An example of  $P_{\max}$  and the corresponding likelihood is shown in Fig. 2.

The threshold circuit causes the sensor to measure the shortest distance, out of which the signal that is received and amplified by  $g$  exceeds a given threshold  $P_E$ . Consequently, the measurement probability

$$p'(r_i | \mathbf{x}) = p(g \cdot P_R(r_i) > P_E | \mathbf{x}) \cdot \left(1 - \sum_{j < i} p'(r_j | \mathbf{x})\right) \quad (6)$$

is the product of the probability that the amplified signal exceeds the threshold and the probability that the measurement procedure not already has been terminated. Thereby, we discretize the measured distances into  $r_0, \dots, r_M$  similar to Moravec *et al.* [26].

Additionally, dynamic, unmapped objects could influence the measurement which is modeled by a small probability  $\beta$ . Furthermore, the sensor could fail with probability  $\gamma$  and generate uniformly distributed measurements which results in the measurement likelihood

$$p(r_i | \mathbf{x}) = (1 - \gamma) \cdot ((1 - \beta) p'(g \cdot P_R(r_i) > P_E | \mathbf{x}) + \beta) \cdot \left(1 - \sum_{j < i} p(r_j | \mathbf{x})\right) + \gamma \cdot p_{\text{uniform}}(r_i) \quad (7)$$

In the training stage we learn the model parameters  $\alpha, \beta, \gamma$ , and  $P_E$  by maximizing the joint log likelihood

$$\log p(z_{1:T} | \mathbf{x}_{1:T}^*, \alpha, \beta, \gamma, P_E) = \sum_{t=1}^T \log p(z_t | \mathbf{x}_t^*, \alpha, \beta, \gamma, P_E) \quad (8)$$

of the sonar measurement data  $z_{1:T}$  given the ground truth trajectory  $\mathbf{x}_{1:T}^*$  of the airship. The resulting likelihood is shown in Fig. 2. Furthermore, we precalculate the set of potentially reflecting objects of the static map to avoid executing computationally expensive ray-casting operations during localization.

## 5 Air Flow Sensor Model

Often, air flow sensors are applied for velocity or wind speed estimation on UAVs [12, 40]. Most of these flow sensors have in common that their one-dimensional measurement value  $z \in \mathbb{R}$  depends on the air velocity  $v_z$  along the fixed measurement axis of the sensor. In the following, we summarize our probabilistic sensor model originally introduced in [29]. It models the measurement likelihood  $p(z | \mathbf{x})$  of air flow sensors with a fixed measurement axis.

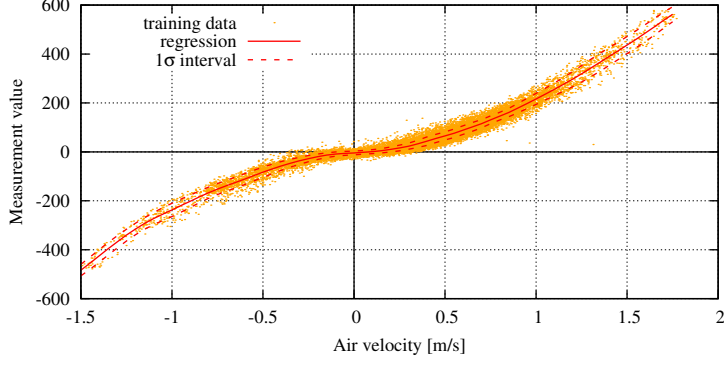


Figure 3: The local linear regression on the air flow sensor training data generated from about 20 min of operation. The regression on the measurement noise is represented by the  $1\sigma$  interval.

We model the measurement principle by assuming Gaussian noise in the heteroscedastic measurement process

$$p(z \mid \mathbf{x}) = \mathcal{N}(z; h(v_z(\mathbf{x})), \sigma(v_z(\mathbf{x}))^2), \quad (9)$$

where  $h$  is a strictly monotonic increasing function of the velocity of the air sweeping over the sensor. The variance  $\sigma^2$  typically depends on the sensor characteristics as well as the air velocity. In indoor navigation scenarios, we assume the air to be static (no wind). Depending on the translational and rotational velocity of the blimp the velocity of the air sweeping over the sensor along its measurement axis  $\mathbf{n}_z$  is

$$v_z(\mathbf{x}) = (\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_z) \cdot \mathbf{n}_z. \quad (10)$$

For the implementation of the model described above, we learn  $h : \mathbb{R} \rightarrow \mathbb{R}$  and  $\sigma : \mathbb{R} \rightarrow \mathbb{R}$  from a set of training data using local linear regression. This approach (see Fig. 3 for an application) is a non-parametric regression method that can correctly model the heteroscedastic uncertainty and has good extrapolation properties.

Whereas local linear regression provides an accurate regression of the measurement function, its main drawback lies in its computational complexity. A single function call requires  $O(M)$  where  $M$  is the typically large number of training data points. To obtain an efficient approximation, we discretize the function and store a dense grid containing its values and variance so that approximative function calls are in  $O(1)$  with linear interpolation between the nearest neighbors. This approximation of the regression furthermore allows us to calculate the inverse  $h^{-1}$  of the strictly monotonic increasing measurement function through a binary search in the function values of the model approximation. Including linear interpolation, this results in a complexity of  $O(\log n)$  where  $n$  is the number of grid points of the discrete approximation.

## 6 IMU Sensor Model

Inertial sensors are a well studied topic and most commercially available IMUs are equipped with accelerometers, gyroscopes, and magnetometers on three axes. Furthermore, they are usually equipped with a low-power processor which carries out sensor calibration and data fusion using a variant of the Kalman filter. In this paper, we assume that the IMU provides an accurate orientation estimate in a quaternion  $\mathbf{z}_I$ .

We model the measurement likelihood of the orientation estimates by assuming that

$$\mathbf{z}_1 = \mathbf{q} \odot \tilde{\mathbf{q}}(\gamma_1, \gamma_2, \gamma_3) \quad \text{with} \quad \gamma_i \sim \mathcal{N}(0, \sigma_1^2), \quad (i = 1, 2, 3) \quad (11)$$

where  $\mathbf{q}$  is the orientation contained in the state vector  $\mathbf{x}$  and  $\odot$  is the quaternion product. Further,  $\tilde{\mathbf{q}}(\Delta\phi, \Delta\theta, \Delta\psi)$  is the quaternion representing the incremental rotation  $\Delta\phi$ ,  $\Delta\theta$ , and  $\Delta\psi$  around all three axes and  $\sigma_1$  is the standard deviation of the measurement errors  $\gamma_1, \gamma_2, \gamma_3$  of the IMU. In a first order Taylor approximation of  $\tilde{\mathbf{q}}$  this results in the probabilistic sensor model

$$p(\mathbf{z}_1 | \mathbf{x}) = \mathcal{N}([\mathbf{0}, I_{3 \times 3}] (\mathbf{q}^{-1} \odot \mathbf{z}_1); \mathbf{0}, \Sigma_1) \quad (12)$$

with  $\Sigma_1 = \frac{\sigma_1^2}{\sqrt{2}} I_{3 \times 3}$ . Here,  $\mathbf{q}^{-1}$  is the inverse of  $\mathbf{q}$  [8] and  $I_{3 \times 3}$  is the three-dimensional identity matrix.

## 7 Physical Simulation-based Motion Model for Miniature Airships

The probabilistic motion model  $p(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{u}_t)$  is the core component of the prediction step in the Monte Carlo state estimation process. In the following, we summarize our control motion model introduced in [29]. We first derive a deterministic model by considering the control commands and the underlying physics of the motion of miniature airships. In the physical model of airships, the Newton-Euler equation of motion

$$M [\mathbf{a}^T, \boldsymbol{\alpha}^T]^T = \mathfrak{F}_{\text{external}}(\mathbf{x}) + \mathfrak{F}_{\text{fictitious}}(\mathbf{x}) \quad (13)$$

couple the acceleration to the force and torque  $\mathfrak{F} = [\mathbf{F}^T, \boldsymbol{\tau}^T]^T$  through the inertia matrix  $M$ . In particular, the external forces and torques are the buoyancy and gravity  $\mathfrak{F}_{\text{bg}}$ , the drag of the hull  $\mathfrak{F}_{\text{D,h}}$  and the fins  $\mathfrak{F}_{\text{D,f}}$ , and the propulsion of the rotors  $\mathfrak{F}_{\text{r}}$ . Additionally, fictitious forces  $\mathfrak{F}_{\text{fictitious}}$  have to be taken into account as (13) is defined with respect to the body-fixed frame  $\mathcal{F}_{\text{b}}$ .

We solve the second-order differential equation (13) for the full 12-dimensional state  $\mathbf{x} = [\mathbf{p}^T, \mathbf{q}^T, \mathbf{v}^T, \boldsymbol{\omega}^T]^T$  of the vanilla particle filter through numerical integration assuming constant acceleration during each time step. We estimate the model parameters, which are hard to identify individually, from data recorded during the operation of the real airship.

In a second step, we extend the deterministic model by a statistical identification of the sources of uncertainty. From the remaining errors in the acceleration estimates in the parameter identification, we can estimate the covariance of the acceleration errors. This implicitly defines the probabilistic model needed in the prediction step of the particle filter by error propagation through numerical integration.

## 8 Efficient Particle Filter Localization with an Odometry Motion Model

For ground robots, often odometry motion models are applied for localization. These models integrate the measurements of wheel rotations, an approach that is typically more accurate than predicting and integrating the motor accelerations based on the control commands [7, 22, 25]. Unfortunately, most robotic airships are not equipped with appropriate motion sensors. They therefore have to rely on a physical motion simulation based

on controls, which suffers from large errors [21, 30], even when adaptive motion models [28] are used. In this section, we transfer the idea of using odometry measurements together with probabilistic motion models to flying vehicles equipped with an IMU and three or more air flow sensors. Thereby, we explicitly consider the measurement uncertainty of the odometry sensors when drawing particles from the odometry motion model in the particle filter.

When using the odometry motion model, which is a first-order model, the state vector of the particle filter can be reduced to the pose  $\mathbf{x} = [\mathbf{p}^T, \mathbf{q}^T]$  which drastically decreases the dimensionality of the filtered state from twelve to six. In our efficient implementation with the odometry motion model, the proposal distribution of the particle filter  $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, z_{F,1}, \dots, z_{F,m}, \mathbf{z}_I)$  is based on the IMU and air flow measurements and consequently the likelihood

$$p(z_{S,1}, \dots, z_{S,\ell} \mid \mathbf{x}) = \prod_{i=1}^{\ell} p(z_{S,i} \mid \mathbf{x}) \quad (14)$$

takes into account the sonar measurements and is used for weighing the particles .

In the remainder of this section we assume  $\mathbf{u}$  to represent the odometry measurements  $z_{F,1}, \dots, z_{F,m}, \mathbf{z}_I$ . This is a common practice in the context of odometry motion models for wheeled robots [39]. In our case, the odometry measurements are those from the IMU and all air flow sensors. Consequently, in this context,  $\mathbf{z}$  represents the remaining measurements which are only the measurements of all sonar sensors.

## 8.1 IMU and Air Flow Sensor Odometry

For dead reckoning odometry of flying vehicles, the full six-dimensional velocity consisting of  $\mathbf{v}$  and  $\boldsymbol{\omega}$  is required. Whereas the rotational part  $\boldsymbol{\omega}$  is directly provided by the IMU, the translational part  $\mathbf{v}$  cannot be obtained in a straightforward way for flying vehicles.

In the following, we assume the vehicle to be equipped with an IMU and  $m \geq 3$  air flow sensors which are mounted at the positions  $\mathbf{r}_1, \dots, \mathbf{r}_m$  and whose measurement axes are  $\mathbf{n}_1, \dots, \mathbf{n}_m$ . For a compact representation we combine the measurements  $z_1, \dots, z_m$  of all air flow sensors and formulate the joint measurement function

$$\begin{bmatrix} z_1 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} h((\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_1) \cdot \mathbf{n}_1) + \varepsilon_1 \\ \vdots \\ h((\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_m) \cdot \mathbf{n}_m) + \varepsilon_m \end{bmatrix} \quad (15)$$

according to (9) and (10) with  $\varepsilon_i \sim \mathcal{N}(0, \sigma((\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_i) \cdot \mathbf{n}_i)^2)$ . We solve (15) for  $\mathbf{v}$  by applying the inverse



measurement function  $h^{-1}$  and obtain

$$\begin{bmatrix} h^{-1}(z_1 - \varepsilon_1) \\ \vdots \\ h^{-1}(z_m - \varepsilon_m) \end{bmatrix} = \begin{bmatrix} (\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_1) \cdot \mathbf{n}_1 \\ \vdots \\ (\mathbf{v} + \boldsymbol{\omega} \times \mathbf{r}_m) \cdot \mathbf{n}_m \end{bmatrix} \quad (16)$$

$$= \begin{bmatrix} \mathbf{v} \cdot \mathbf{n}_1 \\ \vdots \\ \mathbf{v} \cdot \mathbf{n}_m \end{bmatrix} + \begin{bmatrix} (\boldsymbol{\omega} \times \mathbf{r}_1) \cdot \mathbf{n}_1 \\ \vdots \\ (\boldsymbol{\omega} \times \mathbf{r}_m) \cdot \mathbf{n}_m \end{bmatrix} \quad (17)$$

$$= [\mathbf{n}_1, \dots, \mathbf{n}_m]^T \mathbf{v} + [\mathbf{r}_1 \times \mathbf{n}_1, \dots, \mathbf{r}_m \times \mathbf{n}_m]^T \boldsymbol{\omega} \quad (18)$$

$$= A\mathbf{v} + B\boldsymbol{\omega} \quad (19)$$

with the constant matrices  $A = [\mathbf{n}_1, \dots, \mathbf{n}_m]^T$  and  $B = [\mathbf{r}_1 \times \mathbf{n}_1, \dots, \mathbf{r}_m \times \mathbf{n}_m]^T$  depending on the arrangement of the air flow sensors on the vehicle. We invert  $A$  using the left pseudo-inverse  $A_{\text{left}}^{-1} = (A^T A)^{-1} A^T$  which requires  $\text{rank}(A) = 3$ . This can be guaranteed by at least three air flow sensors having linearly independent measurement axes. We obtain

$$\mathbf{v} = A_{\text{left}}^{-1} \left( \begin{bmatrix} h^{-1}(z_1 - \varepsilon_1) \\ \vdots \\ h^{-1}(z_m - \varepsilon_m) \end{bmatrix} - B\boldsymbol{\omega} \right), \quad (20)$$

which is the least-squares solution for  $\mathbf{v}$  of the over-constrained equation system (19) in case of  $m > 3$  [41]. In case of  $m = 3$  it is the exact solution and the left pseudo-inverse is equal to the inverse  $A^{-1}$ .

In the odometry motion model, we obtain the rotational velocity  $\boldsymbol{\omega}_I = \boldsymbol{\omega} + \boldsymbol{\delta}$  from the IMU. The measurement error  $\boldsymbol{\delta} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\boldsymbol{\omega}, I})$  is modeled as zero mean Gaussian noise so that the resulting velocity estimates are

$$\boldsymbol{\omega} = \boldsymbol{\omega}_I - \boldsymbol{\delta} \quad (21)$$

and

$$\mathbf{v} = A_{\text{left}}^{-1} \left( \begin{bmatrix} h^{-1}(z_1 - \varepsilon_1) \\ \vdots \\ h^{-1}(z_m - \varepsilon_m) \end{bmatrix} - B(\boldsymbol{\omega}_I - \boldsymbol{\delta}) \right). \quad (22)$$

In the particle filter, we utilize the dead reckoning odometry motion model as the proposal distribution in the prediction step. Thereby, we sample the measurement errors  $\varepsilon_1, \dots, \varepsilon_m$  and  $\boldsymbol{\delta}$  from zero mean Gaussian distributions where the covariances can be identified from recorded measurement data with ground truth information. Subsequently, we propagate the errors through (22) and compute the position hypothesis of each particle from the resulting velocities by numerical integration. Additionally, we exploit the orientation estimated by the on-board sensor data fusion of the IMU by sampling the orientation hypothesis  $\mathbf{q}^{[i]}$  of each particle directly from the IMU sensor model described in Section 6.

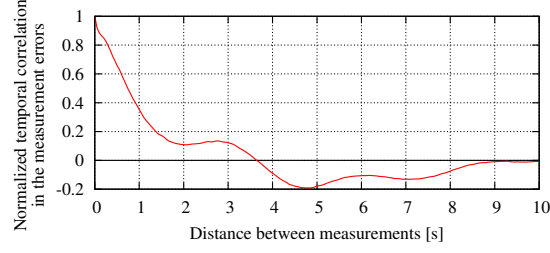


Figure 4: The temporal correlation in the measurement errors of the air flow sensor that senses the forward velocity of the blimp. Each measurement error is computed as the deviation between the actual measurement and the prediction of our air flow sensor model given the ground truth velocity information.

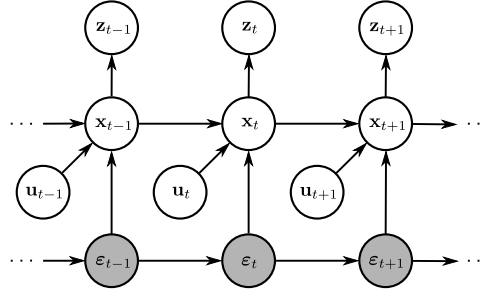


Figure 5: The extended Dynamic Bayesian Network for localization of a mobile robot taking into account the temporal correlation in the odometry measurement errors. It characterizes the evolution of controls  $\mathbf{u}$ , states  $\mathbf{x}$ , measurements  $\mathbf{z}$ , and measurement errors of the odometry data  $\epsilon$ .

## 8.2 Odometry Data with Temporally Correlated Measurement Errors

One fundamental assumption of the Bayes filter is the Markov assumption which states that the random variables of the measurements and the motion of the robot are conditionally independent given the state of the system [39]. However, as shown in Fig. 4, the measurement errors of an air flow sensor and therefore the corresponding random variables are temporally correlated. This effect is caused by turbulences and the motion of the surrounding air, which is displaced by the vehicle and partially accelerated with it. Although Bayes filters have been found to be surprisingly robust to violations of the Markov assumption [39], we explicitly take the correlation into account in the filtering process. To achieve this, we combine the measurement errors of all air flow sensors together in the vector  $\epsilon = [\epsilon_1, \dots, \epsilon_m]^T$  and explicitly consider their history as illustrated in Fig. 5. In particular, we estimate the history of errors together with the state of the robot and extend the posterior of the particle filter to

$$p(\mathbf{x}_{1:t}, \epsilon_{1:t} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) = \eta p(\mathbf{z}_t \mid \mathbf{x}_{1:t}, \mathbf{z}_{1:t-1}, \epsilon_{1:t}, \mathbf{u}_{1:t}) p(\mathbf{x}_{1:t}, \epsilon_{1:t} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \quad (23)$$

where  $\eta$  is a normalizer. We factorize the second conditional probability twice and obtain

$$p(\mathbf{x}_{1:t}, \epsilon_{1:t} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) = p(\mathbf{x}_t, \epsilon_t \mid \mathbf{x}_{1:t-1}, \epsilon_{1:t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_{1:t-1}, \epsilon_{1:t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \quad (24)$$

$$= p(\mathbf{x}_t \mid \mathbf{x}_{1:t-1}, \epsilon_{1:t}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\epsilon_t \mid \mathbf{x}_{1:t-1}, \epsilon_{1:t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) \quad (25)$$

$$p(\mathbf{x}_{1:t-1}, \epsilon_{1:t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) .$$

Under the Markov assumption in the extended network shown in Fig. 5, (23) together with (25) can be simplified to

$$\begin{aligned} p(\mathbf{x}_{1:t}, \boldsymbol{\varepsilon}_{1:t} \mid \mathbf{z}_{1:t}, \mathbf{u}_{1:t}) \\ = \eta p(\mathbf{z}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \boldsymbol{\varepsilon}_t, \mathbf{u}_t) p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1}, \mathbf{x}_{1:t-1}, \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t}) p(\mathbf{x}_{1:t-1}, \boldsymbol{\varepsilon}_{1:t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}) \end{aligned} \quad (26)$$

$$= \eta p(\mathbf{z}_t \mid \mathbf{x}_t) p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \boldsymbol{\varepsilon}_t, \mathbf{u}_t) p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1}) p(\mathbf{x}_{1:t-1}, \boldsymbol{\varepsilon}_{1:t-1} \mid \mathbf{z}_{1:t-1}, \mathbf{u}_{1:t-1}). \quad (27)$$

Here,  $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \boldsymbol{\varepsilon}_t, \mathbf{u}_t)$  is the odometry motion model as described in Section 8.1 conditioned on the air flow measurement error. Note that in this context  $\mathbf{u}$  represents the air flow and IMU odometry measurement data. The term  $p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1})$  is the air flow measurement error transition model and (27) follows from d-separation on the dynamic Bayes network (Fig. 5). Learning this full high-dimensional probability density function would require a large amount of data and is prone to overfitting. Therefore, we apply the joint Gaussian approximation

$$\mathbf{e}_{1:t} \sim \mathcal{N}(\mathbf{0}, \Sigma_{\boldsymbol{\varepsilon}}) \quad \text{with} \quad \mathbf{e}_{1:t} = [\boldsymbol{\varepsilon}_1^T, \dots, \boldsymbol{\varepsilon}_t^T]^T \quad \text{and} \quad \Sigma_{\boldsymbol{\varepsilon}} = \text{Cov}(\mathbf{e}_{1:t}). \quad (28)$$

In the prediction step of the particle filter we sample from the odometry motion model  $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \boldsymbol{\varepsilon}_t, \mathbf{u}_t)$   $p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1})$  as the proposal distribution by first sampling  $\boldsymbol{\varepsilon}_t$  from the joint Gaussian

$$p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1}) = \mathcal{N}(\Sigma_{21}\Sigma_{11}^{-1}\mathbf{e}_{1:t-1}, \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}) \quad (29)$$

with the covariance

$$\Sigma_{\boldsymbol{\varepsilon}} = \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix} \quad (30)$$

so that  $\Sigma_{11} = \text{Cov}(\mathbf{e}_{1:t-1})$  and  $\Sigma_{22} = \text{Cov}(\boldsymbol{\varepsilon}_t)$  [10]. In the second step, we sample the motion of the vehicle from  $p(\mathbf{x}_t \mid \mathbf{x}_{t-1}, \boldsymbol{\varepsilon}_t, \mathbf{u}_t)$  conditioned on the (temporally correlated) measurement error  $\boldsymbol{\varepsilon}_t$  of the air flow sensors.

The method described above accounts for the temporal correlation in the air flow measurement errors in the particle filter. However, in practice, amending each particle with the full history of the measurement errors results in the computational complexity  $O(TN)$  of each resampling step where  $T$  is the length of the trajectory and  $N$  is the number of particles. As can be seen in Fig. 4, in practice, the correlation of the measurement errors is limited to a few seconds so that one can safely limit the size of the history to a constant value  $h$  depending on the temporal correlation and the measurement frequency. This results in the approximation

$$p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{1:t-1}) \approx p(\boldsymbol{\varepsilon}_t \mid \boldsymbol{\varepsilon}_{t-h:t-1}) \quad (31)$$

so that the resampling step of the particle filter is in  $O(N)$  as in its standard implementation.

## 9 Experiments

We evaluated our approach and compared it to existing airship localization techniques in experiments carried out in a large indoor environment. We created a complex maze-like environment with a size of approximately  $10\text{ m} \times 10\text{ m}$  as shown in Fig. 6. The environment was mapped using a laser range finder and modeled using the



Figure 6: Our miniature blimp operating in the maze-like indoor environment.

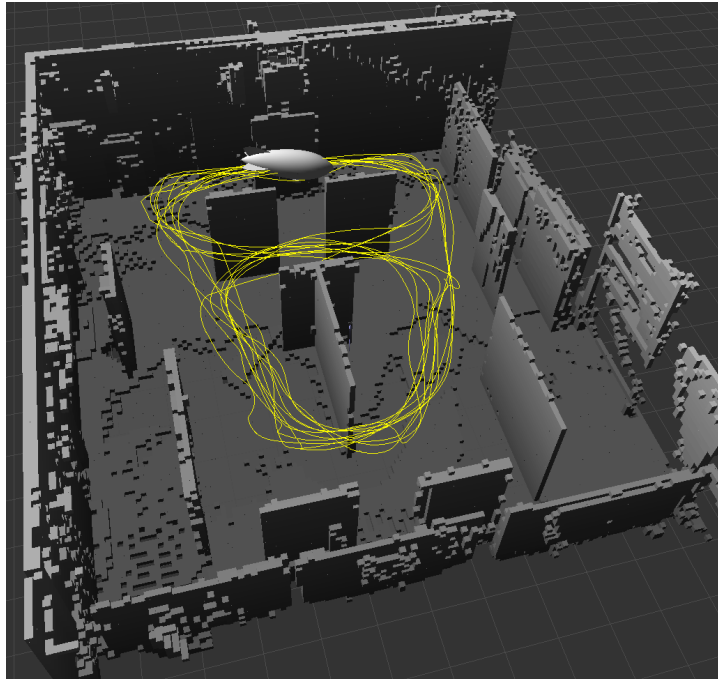


Figure 7: The OctoMap representation of the indoor environment in which the blimp was manually operated during the experiments. The trajectory of the blimp has a length of 276.6 m and is depicted in yellow. The blimp itself is shown at its final position on the trajectory.

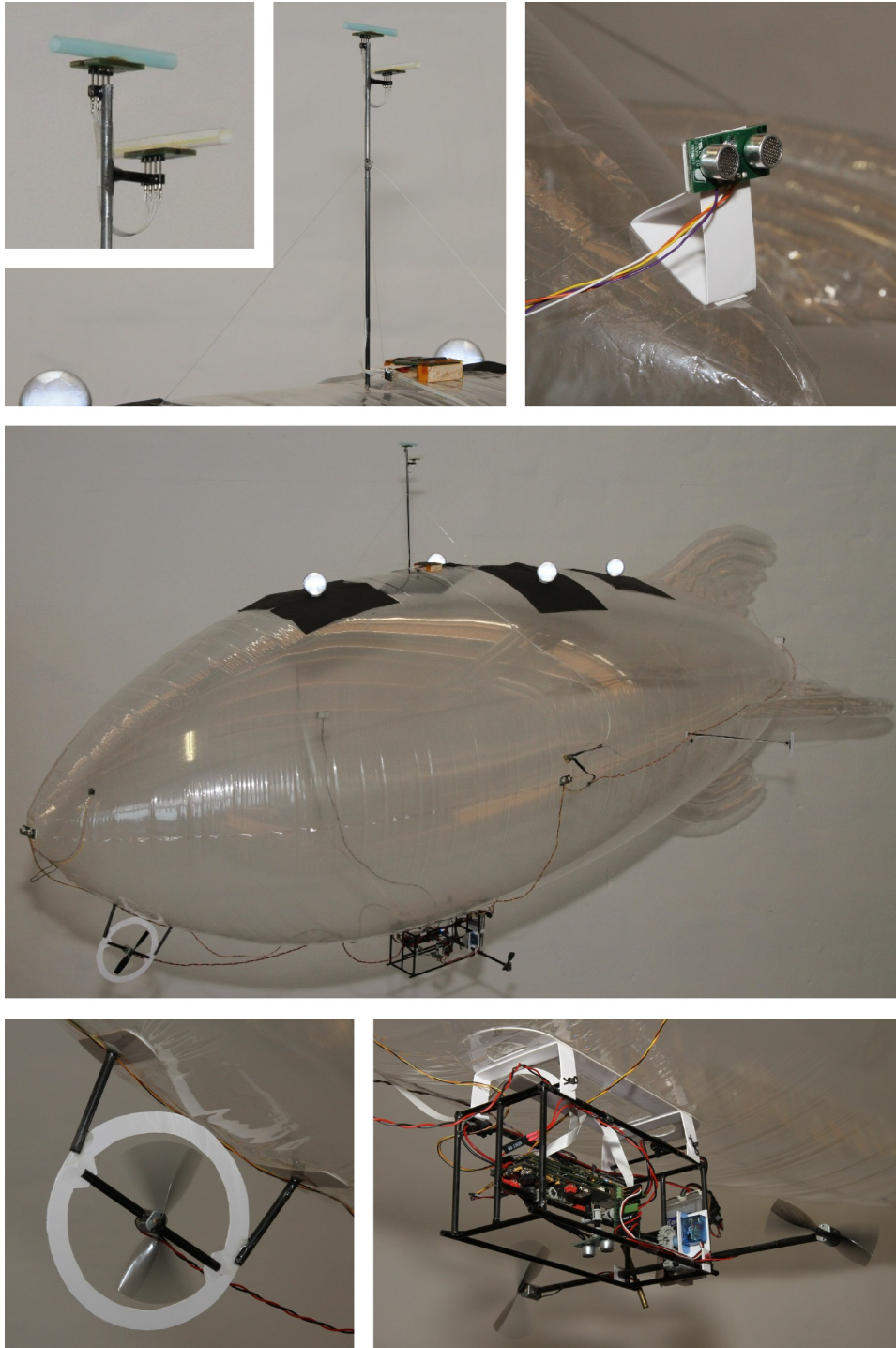


Figure 8: Our miniature blimp is equipped with three air flow sensors mounted on poles. Two of them are mounted at the top of the blimp and a third sensor is mounted laterally. Additionally, an IMU is placed at the top of the airship, four sonar sensors are mounted at the hull, and one sonar is integrated into the gondola. The blimp is actuated by three propellers, where two of them can be pivoted together and a third one provides thrust for yaw rotation.

OctoMap framework [43], which provides a tree-based map structure representing the occupancy of 3D volume elements in a hierarchical fashion. The resulting map with a resolution of 10 cm is shown in Fig. 7.

Our blimp, depicted in Fig. 8, is 2.10 m long and has a payload of about 150 grams for carrying sensors, a computer, and controlling electronics. It is actuated by two main propellers that pivot together, providing thrust in the forward/backward and upward/downward directions. A third propeller is mounted laterally at the front of the blimp for yaw rotation. For autonomous operation our blimp is equipped with a Gumstix computer and the control electronics developed by Rottmann *et al.* [31]. To localize itself, our blimp is equipped with five Devantech SRF10 miniature sonar sensors, each of them weighing 3.3 grams. As their membrane has a diameter of approximately 8.5 mm and they emit an ultrasound signal with a wavelength of 8.5 mm, their signal is hardly focused and often provides ambiguous information. Four sonar sensors are mounted horizontally at the front, back, left hand, and right hand side of the hull. The fifth sensor is integrated into the gondola pointing downwards for providing height measurements. Additionally, our blimp is equipped with an IMU [35] that weighs 8.8 grams. The IMU sensor data is fused in an extended Kalman filter on the processor integrated into the IMU and provides accurate attitude and heading estimates. For accurate velocity measurements, the airship is equipped with three SDP600 differential pressure sensors from Sensirion AG, Stäfa, Switzerland, operated here as thermal air flow sensors. Each air flow sensor weighs less than 1 gram and is mounted on a pole at 20 cm distance to the hull to reduce the influence of the surrounding air accompanying the blimp. As shown in Fig. 8, two flow sensors are mounted at the top of the airship to measure the velocity in the  $x$  and  $y$  direction. The third sensor is mounted laterally to measure the velocity in the  $z$  direction.

We implemented two variants of our novel approach as well as two state-of-the-art approaches to airship localization:

1. *Vanilla*: The vanilla implementation of the particle filter with the standard physical simulation-based control motion model.
2. *Vanilla+MP*: The vanilla implementation extended by the simultaneous estimation of the air drag parameters of the control motion model which are hard to obtain accurately from calibration experiments [28].
3. *Flow-IMU-Odometry*: Our novel localization approach with the odometry motion model using air flow and IMU measurements *without* taking into account the temporal correlation of the air flow measurements.
4. *Flow-IMU-Odometry+TC*: Our novel localization approach with the odometry motion model using air flow and IMU measurements taking into account the temporal correlation of the air flow measurements as described in Section 8.2.

As a reference, we obtain ground truth states of the blimp from a MotionAnalysis optical motion capture (MoCap) system with nine Raptor-E cameras. It provides accurate pose estimates at 100 Hz so that the ground truth velocities and accelerations can be obtained from a quadratic regression on the trajectory. Additionally, the MoCap system allows to precisely measure the mounting positions and orientations of all sensors. In the preparation of the experiment we learned the parameters of all models from the MoCap reference trajectory and

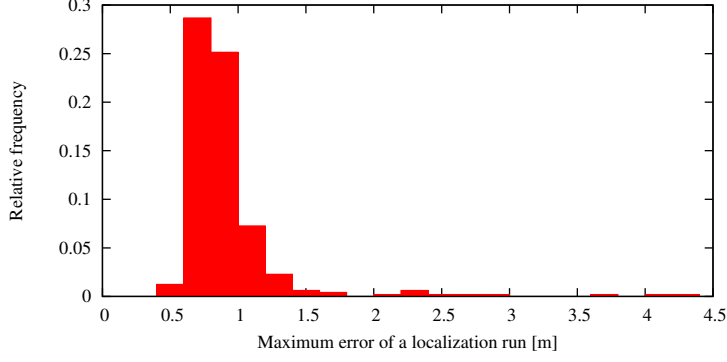


Figure 9: The relative frequency of the maximum localization error of all localization runs of all experiments. In 32 % of the runs, the particle filter lost track of the blimp and the maximum error exceeded 4.5 m, which is not shown in the plot.

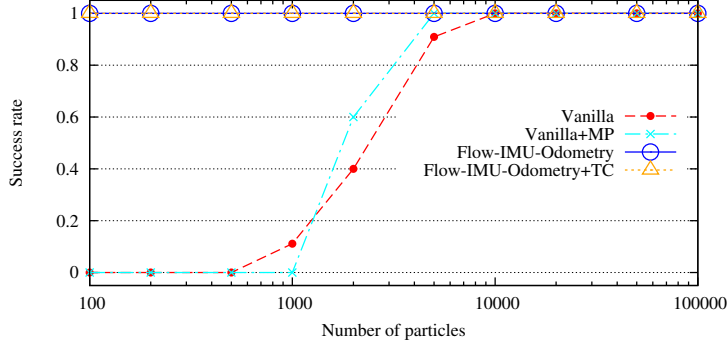


Figure 10: The success rate of the individual localization approaches.

together with the real sensor and control data recorded during manually controlled flight experiments. Neither during the model training, nor during the localization experiments there were any dynamic, unmapped obstacles.

In an experiment we manually controlled the blimp for 10 min through the maze-like environment. The collision-free trajectory had a length of 276.6 m and is depicted in Fig. 7. Despite the fact that the IMU relies on magnetic measurements, the RMS error of the orientation estimates was  $3.22^\circ$  during the experiment in the indoor environment. We evaluate and compare the localization methods on the sensor and control data recorded during operation. As a measure of localization error we use the Euclidean distance between the ground truth position and the position estimate of the particle filter, which is the weighted average of all particles. We evaluated the maximum error of each individual run of our experiments and found two groups of results as shown in Fig. 9: Good runs with a maximum error lower than 2 meters and outliers with high errors (higher than 2 meters). Therefore, we consider a localization run as successful when the position estimate of the filter never deviates more than 2 m from the ground truth position during the whole run. For each successful localization run we evaluate the root mean square (RMS) of the three-dimensional position error over the whole trajectory.

### Offline Comparison

In an offline experiment we evaluated all localization methods with a varying number of particles. The success rate of the individual methods is depicted in Fig. 10. As can be seen, the flow odometry approach facilitates

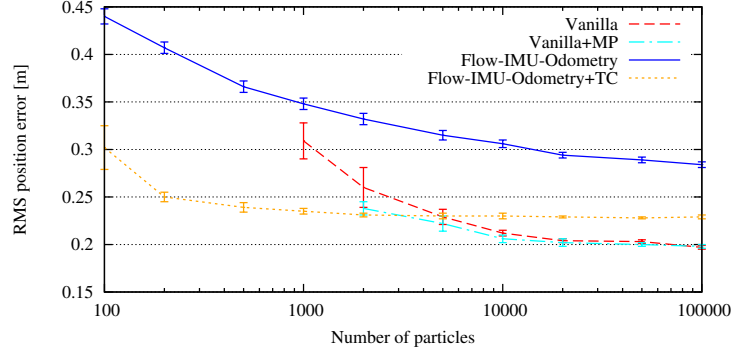


Figure 11: The average RMS localization error of the individual localization approaches. The error bars indicate the 5 % confidence intervals over ten successful runs.

a reliable localization even when a low number of particles is used. This is due to the fact that in this method the velocities are directly measured by the odometry sensors, namely the IMU and the air flow sensors, so that they do not need to be estimated in the particle filter. This decreases the dimensionality of the state estimation problem and therefore fundamentally reduces the number of particles needed to densely represent the area of high likelihood in the posterior about the pose of the vehicle.

The average RMS position errors are depicted in Fig. 11. The vanilla implementations generate significantly lower position errors when using a huge number of particles. This is caused by the fact that they integrate all sensor information and, in contrast to our novel approaches, additionally take into account the information about the control commands sent to the rotors. The simultaneous estimation of the motion model parameters (Vanilla+MP) has proven to outperform the vanilla implementation when only very sparse sensor information is available [28]. Here, much more sensor data is available, i.e., in addition, IMU and air flow measurements, so that there is no significant difference in the RMS position error for 100,000 particles. A further result of this experiment is the significantly lower localization error of the odometry motion model when taking into account the temporal correlation of the air flow measurements. This is mainly due to the slow turbulences of the air accompanying the blimp causing slowly varying systematic measurement errors.

### Online Comparison

For autonomous operation, the localization algorithm should be able to provide accurate state estimates during operation for motion planning and closed-loop control. Therefore, we compared our implementation of all localization methods listed above with respect to online operation. All localization methods were executed on an Intel® Atom™ N270 1.6 GHz with 1 GB RAM in a single thread. This processor could potentially be carried by the blimp and is considerably faster than the Gumstix computer, especially in floating point calculations. In this setting, we determined the maximum number of particles that just enables the particle filter running online, i.e., processing the data as fast as it was generated by sensors and actuators. The physical simulations of the control motion model of the vanilla implementations are computationally demanding so that their maximum number of particles is 107 (Vanilla) and 105 (Vanilla+MP), which is far from enabling a reliable online localization. As the odometry motion model is computationally modest compared to the control motion model and the dimensionality



of the state space is reduced, our novel implementations clearly outperform the vanilla implementations in terms of efficiency and allow a reliable online localization on a low-power embedded computer with 564 (Flow-IMU-Odometry) and 455 (Flow-IMU-Odometry+TC) particles, respectively.

## 10 Related Work

In the past, several authors have considered UAVs. For example, Kantor *et al.* [19], Hada *et al.* [16], and Hygounenc *et al.* [18] developed airships with several kilograms of payload and utilized them for surveillance, data collection, or rescue mission coordination tasks. Especially in the context of quadrotor helicopters, several researchers considered online localization or SLAM with on-board sensors in GPS-denied environments [2, 15, 17, 34, 42]. Furthermore, the problem of localizing fixed-wing vehicles with on-board sensors has been successfully addressed [3]. The relatively high payload of these systems, however, allows them to carry more powerful sensors, e.g., laser range finders, and also facilitates more extensive on-board computations compared to miniature blimps.

In the context of blimp platforms that are of similar scale to the ones targeted in this paper there also has been carried out work on localization or even SLAM with cameras. Whereas cameras provide rich information, their images typically cannot be processed on the embedded computers installed on miniature airships [1, 21, 36]. Accordingly, our method relies on alternative lightweight and power-saving on-board sensors, namely sonar, air flow and IMU whose low-dimensional measurement output can be processed even with very limited computational power.

Before laser scanners became available for installation on mobile robots, ultrasound sensors were popular for estimating the distance to objects in the environment of a robot. Originally, robots were equipped with arrays of Polaroid ultrasound sensors that had, compared to the sensors installed on our blimp, a relatively small opening angle. In the literature, several approaches for modeling the behavior of such ultrasound sensors can be found [26, 27, 4]. However, most of these models have been designed for two-dimensional occupancy grid maps only and also do not specifically model the intensity decrease of the sound cone while it propagates. Thrun [38] proposed an approach to occupancy grid mapping that considers multiple objects in the sound cone. However, this approach utilizes a simplified sensor model. Schroeter *et al.* [33] directly learn the likelihood function from data collected with a mobile robot, which is an approach similar to the one described by Thrun *et al.* [39]. Compared to these approaches, our technique seeks to physically model the sensor and explicitly takes into account the potential reflections of objects. Physical models have also been considered by Leonard and Durrant-Whyte [23] and Tardos *et al.* [37]. In contrast to our method, their approach relies on the assumption of certain types of geometric objects such as planes and edges.

Corresponding to the odometry sensors most wheeled robot platforms are equipped with, one can use airspeed sensors on flying vehicles for state estimation or control. For example, Fei *et al.* [12] and Tokutake *et al.* [40] utilize thermal flow sensors on the wings of small unmanned aircrafts for the detection of flight parameters including the airspeed. For attitude estimation, Euston *et al.* [11] fuse IMU and airspeed measurements of a UAV. Furthermore, a popular application of airspeed sensors on UAVs is the combination with GPS or optical flow

for flight stabilization [20] or wind estimation [6]. In this context, also approaches to calibrate the scaling of an airspeed sensor have been developed [5, 32].

However, all approaches for state estimation or control of robots based on flow sensors mentioned above apply maximum likelihood state estimation or control based on the calibrated output of one or multiple identical sensors. In contrast, we explicitly represent a posterior about the state of the system and model the uncertainty of the measurements and of the motion of the robot for probabilistic state estimation. Thus, our approach can seamlessly integrate arbitrary sensors.

Besides airspeed sensing there are other techniques for the prediction of incremental movements of UAVs. Most localization and control approaches for miniature blimps rely on physical simulation-based control motion models which are computationally demanding and require the tedious calibration of several parameters [21, 45, 13, 18, 28]. Furthermore, the motion of UAVs can be predicted based on the acceleration and angular rate measurement of an IMU [3]. However, the acceleration of a blimp during navigation is typically low compared to the gravity which results in a poor signal to noise ratio of the acceleration measurements. In contrast to these approaches, our method closely follows the odometry motion model applied on most ground robots [7, 39]. We transfer the principle of dead-reckoning odometry to our blimp and combine the translational velocity information of air flow sensors with the rotational velocity estimated by an IMU. In contrast to the common localization approaches of wheeled robots, where the uncertainty of the precise wheel odometry sensors is approximated by considering Gaussian noise on the integrated 2D movement [39], we model the measurement uncertainty of the flow sensors and the IMU individually and propagate the uncertainty through the measurement equations.

## 11 Conclusions

In this paper, we presented a novel approach to probabilistic online localization for a miniature blimp equipped with lightweight ultrasound and air flow sensors as well as an IMU. For robust state estimation in a particle filter, we presented probabilistic models explicitly taking into account the special characteristics and uncertainties of the miniature and lightweight sensors applied on our blimp. Furthermore, we introduced an efficient odometry motion model using the measurements of the air flow sensors and the IMU which is less computationally demanding than the standard physical simulation-based control motion model.

Our approach has been implemented and thoroughly tested on a real blimp in a complex indoor environment. In all experiments, our approach has been proven to allow accurate and reliable online localization of a miniature blimp and to outperform the particle filter localization based on the standard control motion model. As the odometry motion model provides accurate measurements of the velocity of the blimp, the dimensionality of the state space in the filter is decreased and therefore the number of particles required for a reliable localization is reduced by one order of magnitude. Additionally, we demonstrate significant improvements in terms of localization accuracy by taking into account the temporal correlation of the air flow measurements in our novel odometry motion model.

## Acknowledgments

This work has partly been supported by the German Research Foundation (DFG) within the Research Training Group 1103. The authors would like to thank Maximilian Beinhofer, Oliver Paul, and Jan Peters for valuable suggestions and fruitful discussions. Sensirion AG is gracefully acknowledged for providing their flow sensor chips.

## References

- [1] A. Angeli, D. Filliat, S. Doncieux, and J.-A. Meyer. 2D simultaneous localization and mapping for micro aerial vehicles. In *European Micro Aerial Vehicles (EMAV)*, 2006.
- [2] A. Bachrach, S. Prentice, R. He, and N. Roy. RANGE - Robust autonomous navigation in GPS-denied environments. *Journal of Field Robotics*, 28(5):644–666, 2011.
- [3] A. Bry, A. Bachrach, and N. Roy. State estimation for aggressive flight in GPS-denied environments using onboard sensing. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1–8, 2012.
- [4] W. Burgard, D. Fox, D. Hennig, and T. Schmidt. Estimating the absolute position of a mobile robot using position probability grids. In *Proc. of the National Conf. on Artificial Intelligence (AAAI)*, pages 896–901, 1996.
- [5] H. Chao and Y.Q. Chen. Surface wind profile measurement using multiple small unmanned aerial vehicles. In *Proc. of the American Control Conf.*, pages 4133–4138, 2010.
- [6] A. Cho, J. Kim, S. Lee, and C. Kee. Wind estimation and airspeed calibration using a UAV with a single-antenna GPS receiver and pitot tube. *IEEE Transactions on Aerospace and Electronic Systems*, 47(1):109–117, 2011.
- [7] F. Dellaert, D. Fox, W. Burgard, and S. Thrun. Monte Carlo localization for mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1322–1328, 1999.
- [8] J. Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. Technical report, Stanford University, 2006.
- [9] A. Doucet, N. de Freitas, and N. Gordan. *Sequential Monte-Carlo Methods in Practice*. Springer Verlag, 2001.
- [10] M.L. Eaton. *Multivariate Statistics: A Vector Space Approach*. Wiley, 1983.
- [11] M. Euston, P. Coote, R. Mahony, J. Kim, and T. Hamel. A complementary filter for attitude estimation of a fixed-wing UAV. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 340–345, 2008.

- [12] H. Fei, R. Zhu, Z. Zhou, and J. Wang. Aircraft flight parameter detection based on a neural network using multiple hot-film flow speed sensors. *Smart Materials and Structures*, 16(4):1239–1245, 2007.
- [13] S.B.V. Gomes and J.G. Jr. Ramos. Airship dynamic modeling for autonomous operation. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3462–3467, 1998.
- [14] G. Grisetti, C. Stachniss, and W. Burgard. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE Transactions on Robotics*, 23(1):34–46, 2007.
- [15] S. Grzonka, G. Grisetti, and W. Burgard. A fully autonomous indoor quadrotor. *IEEE Transactions on Robotics*, 8(1):90–100, 2012.
- [16] Q. Hada, K. Kawabata, H. Kaetsu, and H. Asama. Autonomous blimp system for aerial infrastructure. In *Proc. of the Int. Conf. on Ubiquitous Robots and Ambient Intelligence*, 2005.
- [17] R. He, S. Prentice, and N. Roy. Planning in information space for a quadrotor helicopter in a GPS-denied environments. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1814–1820, 2008.
- [18] E. Hygounenc, I-K. Jung, P. Soueres, and S. Lacroix. The autonomous blimp project at LAAS/CNRS: Achievements in flight control and terrain mapping. *Int. Journal of Robotics Research*, 23(4):473–511, 2004.
- [19] G. Kantor, D. Wettergreen, J.P. Ostrowski, and S. Singh. Collection of environmental data from an airship platform. In *Proc. of the SPIE Conf. on Sensor Fusion and Decentralized Control in Robotic Systems*, pages 76–83, 2001.
- [20] J. Keshavan and J.S. Humpert. MAV stability augmentation using weighted outputs from distributed hair sensor arrays. In *Proc. of the American Control Conference*, pages 4445–4450, 2010.
- [21] J. Ko, D.J. Klein, D. Fox, and D. Haehnel. GP-UKF: Unscented Kalman filters with Gaussian process prediction and observation models. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1901–1907, 2007.
- [22] S. Lenser and M. Veloso. Sensor resetting localization for poorly modelled mobile robots. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 1225–1232, 2000.
- [23] J.J. Leonard and H.F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, Boston, 1992.
- [24] J.S. Liu. Metropolized independent sampling with comparisons to rejection sampling and importance sampling. *Statistics and Computing*, 6(2):113–119, 1996.
- [25] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit. FastSLAM: A factored solution to simultaneous localization and mapping. In *Proc. of the National Conf. on Artificial Intelligence (AAAI)*, pages 593–598, 2002.

- [26] H.P. Moravec. Sensor fusion in certainty grids for mobile robots. *AI Magazine*, 9(2):61–74, 1988.
- [27] H.P. Moravec and A.E. Elfes. High resolution maps from wide angle sonar. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 116–121, 1985.
- [28] J. Müller, C. Gonsior, and W. Burgard. Improved monte carlo localization of autonomous robots through simultaneous estimation of motion model parameters. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 2604–2609, 2010.
- [29] J. Müller, O. Paul, and W. Burgard. Probabilistic velocity estimation for autonomous miniature airships using thermal air flow sensors. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 39–44, 2012.
- [30] J. Müller, A. Rottmann, and W. Burgard. A probabilistic sonar sensor model for robust localization of a small-size blimp in indoor environments using a particle filter. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 3589–3594, 2009.
- [31] A. Rottmann, M. Sippel, T. Ziterell, W. Burgard, L. Reindl, and C. Scholl. Towards an experimental autonomous blimp platform. In *Proc. of the European Conf. on Mobile Robots (ECMR)*, 2007.
- [32] A.J. Rutkowski, M.M. Miller, R.D. Quinn, and M.A. Willis. Egomotion estimation with optic flow and air velocity sensors. *Biological Cybernetics*, 104(6):351–367, 2011.
- [33] C. Schroeter, A. Koenig, H.-J. Boehme, and H.-M. Gross. Multi-sensor Monte-Carlo-localization combining omni-vision and sonar range sensors. In *Proc. of the European Conf. on Mobile Robots (ECMR)*, pages 164–169, 2005.
- [34] S. Shen, M. Nathan, and V. Kumar. Autonomous multi-floor indoor navigation with a computationally constrained MAV. In *Proc. of the IEEE Int. Conf. on Robotics & Automation (ICRA)*, pages 20–25, 2011.
- [35] M. Sippel, A. Abduhl-Majeed, W. Kuntz, and L. Reindl. Enhancing accuracy of an indoor radar by the implementation of a quaternion- and unscented Kalman filter-based lightweight, planar, strapdown IMU. In *Proc. of the European Navigation Conf. (ENC-GNSS)*, 2008.
- [36] B. Steder, G. Grisetti, S. Grzonka, C. Stachniss, A. Rottmann, and W. Burgard. Learning maps in 3d using attitude and noisy vision sensors. In *Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 644–649, 2007.
- [37] J.D. Tardós, J. Neira, P.M. Newman, and J.J. Leonard. Robust mapping and localization in indoor environments using sonar data. *Int. Journal of Robotics Research*, 21(4):311–330, 2002.
- [38] S. Thrun. Learning occupancy grid maps with forward sensor models. *Journal of Autonomous Robots*, 15(2):111–127, 2003.
- [39] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. MIT Press, 2005.

- [40] H. Tokutake, S. Sunada, and J. Fujinaga. Attitude control of a small UAV using a flow sensor system. *Journal of System Design and Dynamics*, 5(1):1–16, 2011.
- [41] L. Wasserman. *All of Nonparametric Statistics*. Springer, 2006.
- [42] S. Weiss, D. Scaramuzza, and R. Siegwart. Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics*, 28(6):854–874, 2011.
- [43] K.M. Wurm, A. Hornung, M. Bennewitz, C. Stachniss, and W. Burgard. OctoMap: A probabilistic, flexible, and compact 3d map representation for robotic systems. In *Proc. of the ICRA 2010 Workshop on Best Practice in 3D Perception and Modeling for Mobile Manipulation*, 2010.
- [44] G. Wyeth and I. Barron. An autonomous blimp. In *Proc. of the IEEE Int. Conf. on Field and Service Robotics (FSR)*, pages 464–470, 1997.
- [45] J.C. Zufferey, A. Guanella, A. Beyeler, and D. Floreano. Flying over the reality gap: From simulated to real indoor airships. *Journal of Autonomous Robots*, 21(3):243–254, 2006.