A Sigma-Point Kalman Filter for remote sensing of updrafts in autonomous soaring

Martin Stolle, Yoko Watanabe and Carsten Döll

Abstract Autonomous soaring is a promising approach to augment the endurance of small UAVs. Most of the existing work on this field relies on accelerometers and/or GPS receivers to sense thermals in the proximity of the vehicle. However, thermal updrafts are often visually indicated by cumulus clouds that are well characterized by their sharp baselines. This paper focuses on a cloud mapping algorithm which estimates the 3D position of cumulus clouds. Using the meteorological fact of a uniform cloud base altitude a state-constrained sigma-point Kalman filter (SCSPKF) is developed. A method of using the resulting cloud map and its uncertainty in the path planning task to realize a soaring flight to a given wayoint is presented as a perspective of this work.

1 Introduction

Accelerated by a breakthrough in micro electromecanical systems (MEMS), small UAVs and the role they play in our society, be it military or civil, have grown in importance in the near past. However, their utility is still restricted due to small payload capacities as well as poor endurance and small operational ranges. One existing idea to overcome these still predominant drawbacks, is to apply flight control and guidance algorithms for soaring flight [1, 2, 3]. The soaring flight makes use of updrafts to lift the UAV and hence to reduce the transported mass dedicated to energy (battery or fuel). Moreover, soaring UAVs operate silently which clearly is a benefit for military purposes.

Martin Stolle · Yoko Watanabe · Carsten Döll

Department of Systems Control and Flight Dynamics (DCSD), The French Aerospace Laboratory (ONERA),

2 avenue Edouard Belin, 31055 Toulouse Cedex 4, France

e-mail: martin.stolle@onera.fr, yoko.watanabe@onera.fr, carsten.doell@onera.fr

Generally speaking, soaring flight combines all kind of techniques to keep an unpowered aircraft airborne. Dynamic soaring for instance is a technique where the vehicle harnesses energy from horizontal wind gradients. In thermal soaring energy is gained by relying on uprising currents of air. These buoyant plumes of rising air result from gradients in the earth's surface heating and can reach heights of up to 4000 m above ground according to [4]. In cross-country soaring, gliders fly beyond the gliding distance from the initial take-off point performing waypoint navigation. Amongst existing approaches to automatic cross-country soaring, the work of Edwards et al. [5] is the only one which includes flight testing. His work lead to the participation in a cross-country soaring challenge for remotely piloted gliders and the performance of a fully autonomous soaring flight over a distance of 50 km. With no a priori information about thermal locations in the far environment, the flight path was defined as the direct line between two consecutive waypoints. The aircraft flight control mode was set to thermal centering mode, when encountering strong enough thermals on the path - detected by Inertial Measurement Unit (IMU) and GPS measurements. With this suboptimal flight path, the UAV could only benefit from a subset of possible updrafts - more precisely those that were directly located on the line of sight to a given waypoint. Evidently this approach is limited to conditions where a strong density of thermals is provided along the direct path and by consequence carries a significant risk of mission failure.

The author of [6] considered autonomous cross-country soaring from a top down approach and proposed path planning algorithms assuming that a perfect map with pinpoint thermal locations is at hand which raises doubts about its applicability beyond the synthetic case of computer simulations.

Human glider pilot mostly rely on their vision to locate thermal updrafts indicated by cumulus clouds. Doing so they can fly distances of up to 3000 km. Inspired by these performances, the paper on hand describes the development of an algorithm for remotely sensing thermal updrafts by locating cumulus clouds. An increased uncertainty of a thermal position estimate can significantly augment the time the UAV will spend on hitting the thermal and thus impacts the cross-country soaring performance. Therefore, the filter was designed to not only provide fast convergence but also a confident estimation of the thermal position uncertainty. Finally, a perspective is presented on how to take into account the uncertainty of estimated thermal positions in the cost of a cross-country path planning algorithm.

2 Cumulus clouds and thermal updrafts

Consider a UAV flying in a sky that is partly covered by cumulus clouds. Depending on their stage, these clouds are the most important visual indicator for thermals that glider pilots rely on during thermal soaring.

2.1 Thermals and their visible features

Vision-based object recognition algorithms detect objects in the real world from an image of the real world based on models. Since algorithmic description of this task still remains difficult, especially when dealing with objects such as clouds, varying in shape, color and texture, most simple and informative features are to be used in order to augment the recognition performance.

Clouds that are based on thermals, in general undergo a certain decay and rebirth process consisting of two different stages. As long as a thermal source on the ground feeds the cloud, it will continue growing and remains in the first building stage (fig. 1a). In case the thermal source vanishes, the cloud will start dying out (fig. 1b). The stages of a cumulus cloud are indicated by a variety of visible signs. For a grow-

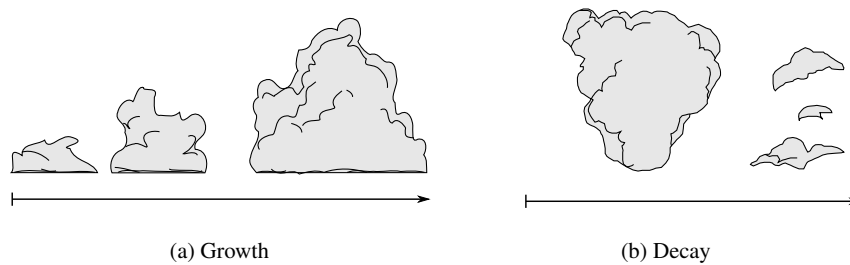(a) Growth   (b) Decay

Fig. 1: Growth and decay of a cumulus cloud

ing cumulus, these features include sharp outlines as well as a dark and flat baseline. In contrast, the shape of a dying cloud is poorly defined and its baseline is rather frayed. In addition to these contrast and shape indicators, the color of a cumulus cloud varies as well during its cycle. While a growing cumulus cloud will tend to be gray or white, a dying cloud appears to be off color since its moisture particles evaporate which results in a change of its reflectivity.

Regarding these facts, the most simple-to-detect feature of a far away growing or mature thermal is the baseline of its related cumulus. In this paper, it is supposed that an image processing algorithm capable of extracting the baseline center of a cumulus cloud, as illustrated in fig. 2 is available. A simple but yet efficient algorithm for edge detection could therefore be used as presented by the author of [7].

Fig. 2: Baseline detection of a cumulus cloud

2.2 Dynamics of cumulus clouds

Solar radiation causes heating of the earth's surface. Variations in the heating of the ground result in rising parcels of air. On a day with typical soaring weather conditions, these parcels will first rise under dry adiabatic conditions i.e. without condensation. Once reaching the Cumulus Condensation Level (CCL) which is the point where the relative humidity of the parcel attains 100%, water vapor starts condensing and cumulus cloud droplets are formed. As shown in [8] and illustrated in fig. 3, the cumulus cloud base altitude z_c is the line intersection of the Dry Adiabatic Lapse Rate (DALR) and the Dew Point (DP).

$$z_c = \frac{T_0 - T_{d,0}}{\frac{\partial T}{\partial z} - \frac{\partial T_d}{\partial z}} \approx 125(T_0 - T_{d,0})[\frac{m}{K}] \tag{1}$$

with T as the air temperature and T_d the dew point temperature. Note that the factor 125 corresponds to a temperature of $20°C$ and barely varies with the temperature. The index 0 stands for the temperature on ground. The index e (as used in fig. 3) is used for a local North East Down (NED) frame.

When flying over a region with constant geological and orographical structure, only tiny variations of the temperature difference between air and DP can be found [9]. Consequently, the cloud base is nearly uniform as depicted in fig. 3 which linearly constraints the altitude of each individual cloud base. The following section describes a cloud mapping algorithm that incorporates this constraint.
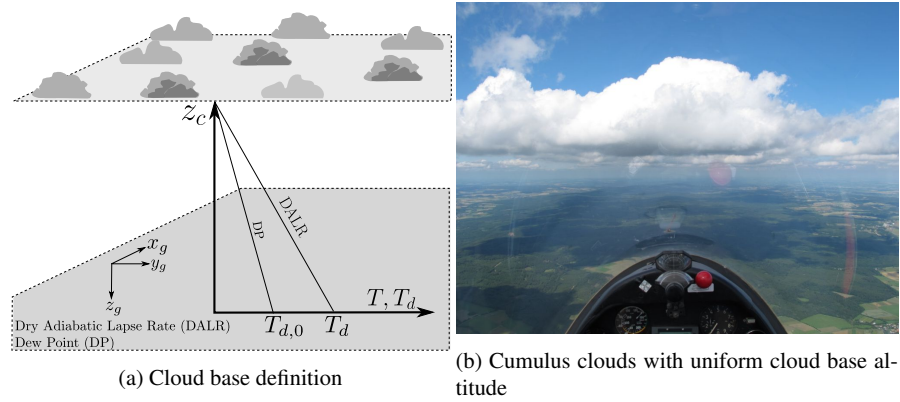
(a) Cloud base definition

(b) Cumulus clouds with uniform cloud base altitude

Fig. 3: Dynamics of cumulus clouds

3 Cloud mapping algorithm

Combining the UAV's state estimates with the output of the image processing algorithm, it becomes possible to estimate the 3D position of clouds in the inertial reference frame (index g). This problem is referred to as bearings-only target localization.

3.1 State definition and process model

With the cloud map containing the individual positions of all n clouds that are encountered during a flight, the $3 \times n$ dimensional state vector $\mathbf{x}$ is defined as

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_1^T & \mathbf{x}_2^T & \ldots & \mathbf{x_n}^T \end{bmatrix}^T \tag{2}$$

where $\mathbf{x}_i$ represents the cloud position of a single cloud in a local *NED* frame. The index g is not further carried for the sake of better readability. In general, the wind velocity has an effect on the drift of cumulus clouds, even if due to the inertia of the thermal air (note that the mass of a cumulus cloud can measure into the thousands of tons), cumulus clouds drift much slower than the surrounding air. Therefore, we consider a scenario with no cloud drift corresponding to weather conditions with only little or no horizontal wind. This assumption is legitimate, since algorithms will be tested on a small UAV glider whose flight envelope restricts operation to low wind conditions. In that case, the state transition equation for a single cloud $\mathbf{x}_i$ can be modeled as

$$\mathbf{x}_{i,k} = f(\mathbf{x}_{i,k-1}) + \mathbf{w}_{i,k-1} = \mathbf{x}_{i,k-1} + \mathbf{w}_{k-1} \tag{3}$$

where $\mathbf{w}$ is the white and Gaussian process noise with covariance $\mathbf{Q}$ i.e.

$$\mathbf{w} \sim \mathbf{w}(0, \mathbf{Q}) \tag{4}$$

3.2 Measurement model

3.2.1 Pixel coordinates of cloud baseline's center

A forward looking camera is mounted on the UAV with a fixed offset from the vehicles center of gravity (CG) as well as a known angular offset from the body axis. At each time instance, the image processing algorithm outputs the center positions of the m cumulus cloud baselines in the image resulting in a $2 \times m$ dimensional measurement vector

$$\mathbf{y}_{ip} = \begin{bmatrix} \mathbf{y}_{ip,1}^T & \mathbf{y}_{ip,2}^T & \cdots & \mathbf{y}_{ip,m}^T \end{bmatrix}^T \tag{5}$$

Note that the index ip is used for vectors in the image frame in pixels and the index im denotes vectors in the image frame in meters as described in fig. 4. To project vectors from the camera frame (index c) onto the image plane, a pinhole camera model as shown in fig. 4 is used.
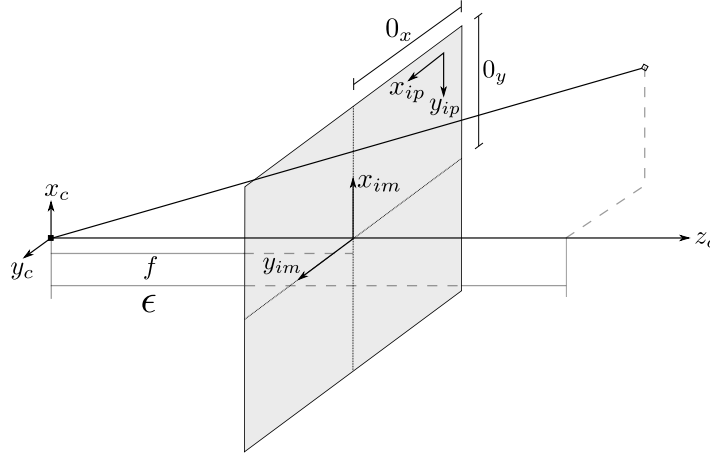

Fig. 4: Pinhole camera projection model

A transformation from the inertial to the image frame in pixels is given by

$$\mathbf{y}_{ip} = \frac{1}{\varepsilon} \mathbf{T}_{ipg} \mathbf{x} \tag{6}$$

where ε is the image depth. The transformation matrix $\mathbf{T}_{ipg}$ includes the translation from the target to the vehicle $\mathbf{T}_{vg}$, the rotation from the vehicle to the body frame $\mathbf{T}_{bv}$, the combined translation and rotation from the body to the camera frame $\mathbf{T}_{cb}$ as well as the combined projection and unit conversion (m to px) $\mathbf{C}$ from the camera frame onto the image plane (see fig. 4)

$$\mathbf{T}_{ipg} = \mathbf{C}\mathbf{T}_{cb}\mathbf{T}_{bv}\mathbf{T}_{vg} \tag{7}$$

with the camera calibration matrix $\mathbf{C}$ being defined as

$$\mathbf{C} = \begin{bmatrix} 0 & f_x & 0_x & 0 \\ -f_y & 0 & 0_y & 0 \end{bmatrix} \tag{8}$$

The two quantities f_x and f_y are function of the focal length f and the unit conversion factors S_x and S_y

$$f_x = \frac{f}{S_x} \text{ and } f_y = \frac{f}{S_y}$$

where the unit conversion is given by

$$S_x = \frac{y_{im}}{x_{ip} - 0_x} \text{ and } S_y = \frac{x_{im}}{-y_{ip} - 0_y}$$

Note that the parameters 0_x and 0_y are the offsets to the center of the image from the upper left hand corner.

Adding Gaussian white measurement noise $\mathbf{v}$, with zero mean and covariance $\mathbf{R}$, the discrete measurement equation is stated as

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k = \frac{1}{\varepsilon}\mathbf{T}_{ipg,k}\mathbf{x}_k + \mathbf{v}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \tag{9}$$

Note that from here on the index of the measurement vector (ip) is not further carried for the sake of better readability.

3.2.2 Pseudo measurement for the altitude constraint

Significant filter performance augmentations can be reached when including the dynamic relation between cloud base altitudes as a state constraint in the estimation process. Applied to the path planning, a faster convergence of position estimates and covariances will as well invoke a faster reduction of the uncertainty ellipses. Consequently, the cross- country speed of the UAV glider increases, since less time will be spent on encountering thermals.

However, this benefit comes with a price. The assumption of a uniform cloudbase is an approximation of the reality, and thus only a soft constraint where it is hard to detect constraints violations during estimation.

To comprise the uniform cloud base state constraint, the $2 \times m$ dimensional measurement vector eq. (9) is augmented with the pseudo measurement $\mathbf{d}$

$$\mathbf{y}_a = \begin{bmatrix} \mathbf{y} \\ \mathbf{d} \end{bmatrix} = \begin{bmatrix} \mathbf{H} \\ \mathbf{D} \end{bmatrix} \mathbf{x} + \begin{bmatrix} \mathbf{v} \\ \mathbf{v}_1 \end{bmatrix} \tag{10}$$

where $\mathbf{d}$ is a null vector of dimension n and $\mathbf{D}$ is the $n \times 3n$-dimensional constraint matrix with diagonal elements $\mathbf{D}_1 = \begin{bmatrix} 0 & 0 & (1 - \frac{1}{n}) \end{bmatrix}$ and off-diagonal elements $\mathbf{D}_0 = \begin{bmatrix} 0 & 0 & \frac{-1}{n} \end{bmatrix}$ carrying the geometrical state restriction that the individual cloud bases z_i equals to the mean cloud base $\bar{z}$. $\mathbf{v}_1$ is the white Gaussian noise of the state constraint with covariance $\mathbf{R}_1$.

3.3 Estimation Algorithm

The bearings-only target localization is a highly nonlinear estimation problem. A variety of nonlinear filters has been proposed to solve this problem. What is common to nearly all of these methods, is the idea of providing a least squares estimate of the process's state. The standard approach for nonlinear estimation is the Extended Kalman Filter (EKF) that however comes with two significant drawbacks. Not only that the computation of the Jacobians is usually cumbersome, but if the linearization is poor, the estimated state covariance will tend to be inconsistent and in the worst case overconfident as discussed in [10]. Projected to the problem of autonomous cross-country soaring, this will erroneously tighten the error ellipsoid associated to the estimated position of a cloud and potentially results in a thermal search within an area of sinking air.

A common way to cope with this known weakness of the EKF is to artificially magnify the state covariance after each update or simply to drop certain observations. This is however an unfortunate and iterative procedure, since it discards information that is potentially useful.

A main challenge of the bearings-only target localization is caused by its lack of depth-observability. With the trajectory having a significant impact on the observability, there have been attempts [11] to design trajectories that optimize the target observability. However, in the case of autonomous cross-country soaring where the clouds are far away from the observing vehicle and the UAV aims to minimize energy consumption, favoring target observability in the trajectory design is inefficient. More recently, a group of algorithms [12, 13, 14, 15] has been published to address the issues of the EKF by using deterministic sampling approaches circumventing both laborious linearization and suboptimal performance due to poor linearizations. These algorithms referred to as sigma-point Kalman filters (SPKF) as well follow the prediction-correction procedure of the Kalman filter. But rather than linearizing the nonlinear system equations, they use the intuition that it is easier to approximate a probability distribution than it is to approximate an arbitrary nonlinear function or transformation. This is done by first propagating a weighted set of samples called sigma points $\mathcal{X}$ through a nonlinear function. Then, the statistic properties of the propagated state are recaptured. The principle behind this probability distribution approximation is called Unscented Transform (UT) and was first presented in [12].

SPKF show a certain resemblance to Particle Filters (PF) in the way that the probability distribution is approximated by a set of points. However, they operate with a much lower number of points reducing computational effort which renders them more appropriate for real time implementation. This has led to the SPKF of [14] where the authors present a filter, capable of operating with a minimum set of sigma points that contains the most important information of the state's probability distribution.

A state-constrained version of this filter has been developed to estimate the position of clouds. The data flow during the estimation process is depicted in fig. 5. In the
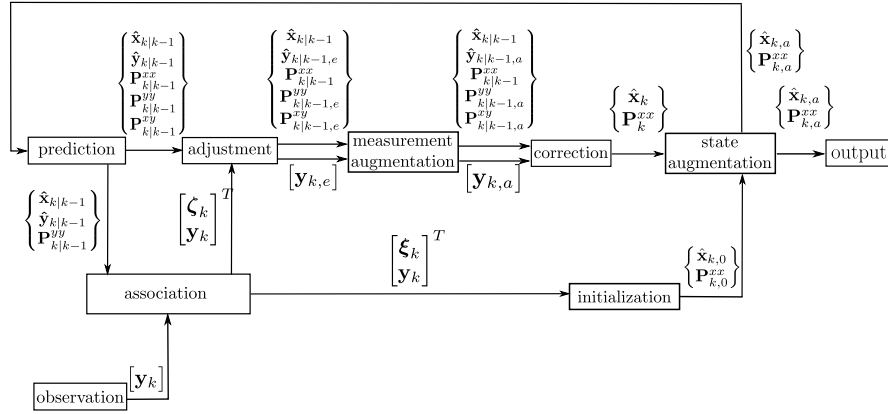

Fig. 5: Data flow during estimation

following, the individual tasks during estimation are discussed.

3.3.1 Prediction

Before conducting the actual prediction using the UT, a set of weight vectors $\boldsymbol{\eta}$ is computed. These vectors depend on both the scaling parameters α, β, κ and the state dimension $L = 3 \times n$.

$$\lambda = \alpha^2 + (L + \kappa) - L$$
$$\eta_0^m = \lambda / (L + \lambda)$$
$$\eta_0^c = \lambda / (L + \lambda) + 1 - \alpha^2 - \beta$$
$$\eta_i^m = \eta_i^c = 1/[2(L+\lambda)], \, i = 1,...,2L \tag{11}$$

According to this notation, the indices m and c in eq. (11) stand for measurement and covariance respectively. With the state estimate prior to k, the sigma points follow as

$$\boldsymbol{\mathcal{X}}_{k-1} = \left[\hat{\mathbf{x}}_{k-1} \; \hat{\mathbf{x}}_{k-1} + \sqrt{L+\lambda}\sqrt{\mathbf{P}^{xx}_{k-1}} \; \hat{\mathbf{x}}_{k-1} - \sqrt{L+\lambda}\sqrt{\mathbf{P}^{xx}_{k-1}}\right] \qquad (12)$$

Note that there are different approaches to compute the square root of a matrix. As suggested in [16], the lower Cholesky decomposition method is applied i.e. $\sqrt{\mathbf{P}} = chol(\mathbf{P})$. Each of the sigma points $\boldsymbol{\mathcal{X}}^{(i)}$ is then propagated through the state transition function eq. (3) yielding the propagated state

$$\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} = \mathbf{f}(\boldsymbol{\mathcal{X}}^{(i)}_{k-1}) \text{ for } i = 1, ..., 2L+1 \qquad (13)$$

In this notation, the index $k|k-1$ stands for the state at time k incorporating knowledge prior to k and the parenthesized superscript stands for the index of the sigma-point. Also note that (eq. (13)) is only mentioned for the purpose of completeness, since the propagation does not impact the state as can be seen in eq. (3).
With the weight vector $\boldsymbol{\eta}^m_i$ the mean of the propagated state is

$$\hat{\mathbf{x}}_{k|k-1} = \sum_{i=1}^{2L+1} \eta^m_i \boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} \qquad (14)$$

Given the process noise covariance $\mathbf{Q} = E[\mathbf{w}\mathbf{w}^T]$, the propagated state covariance matrix yields

$$\mathbf{P}^{xx}_{k|k-1} = \mathbf{Q} + \sum_{i=1}^{2L+1} \eta^c_i (\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} - \hat{\mathbf{x}}_{k|k-1})(\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} - \hat{\mathbf{x}}_{k|k-1})^T \qquad (15)$$

Each of the sigma points is then processed through the nonlinear measurement equation, leading to a set of $2L+1$ predicted observations

$$\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} = \mathbf{h}(\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1}) \text{ for } i = 1, ..., 2L+1 \qquad (16)$$

This yields the mean of the predicted measurement

$$\hat{\mathbf{y}}_{k|k-1} = \sum_{i=1}^{2L+1} \eta^m_i \boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} \qquad (17)$$

Summing the measurement covariance $\mathbf{R}$ and the covariance of the transformed state, the predicted measurement covariance is

$$\mathbf{P}^{yy}_{k|k-1} = \mathbf{R} + \sum_{i=1}^{2L+1} \eta^c_i (\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\mathbf{y}}_{k|k-1})(\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\mathbf{y}}_{k|k-1})^T \qquad (18)$$

The prediction step is accomplished with the computation of the cross covariance matrix

$$\mathbf{P}^{xy}_{k|k-1} = \sum_{i=1}^{2L+1} \eta^c_i (\boldsymbol{\mathcal{X}}^{(i)}_{k|k-1} - \hat{\mathbf{x}}_{k|k-1})(\boldsymbol{\mathcal{Y}}^{(i)}_{k|k-1} - \hat{\mathbf{y}}_{k|k-1})^T \qquad (19)$$

3.3.2 Data association, measurement augmentation and adjustment

Depending on meteorological conditions, the density of the thermals in an area can significantly vary [4]. Assuming that each thermal is visible through convection i.e. brings out a cumulus cloud, multiple clouds will simultaneously lie in the camera's field of vision. Therefore, precise matching between incoming measurements and already registered estimates is required to avoid filter divergence. Also, measurements from newly detected clouds have to be distinguished from those belonging to already initialized ones.

Data association

In this work, we apply a gated nearest neighbor approach based on the Mahalaboni distance. Where the underlying idea is to compute the probability that a predicted measurement corresponds to an incoming measurement. This technique has proven to work reliably [17, 18], provided that the uncertainty of the predicted measurements $\mathbf{P}^{yy}_{k|k-1}$ is sufficiently small.

At each time instance with incoming measurements for m detected clouds, the measurement vector is defined by eq. (9). A score r is defined and computed for the $m \times n$ combinations between predicted measurements and incoming measurements

$$r_k^{(ij)} = (\mathbf{y}_k^{(j)} - \boldsymbol{\mathcal{Y}}_{k|k-1}^{(i)})\mathbf{P}^{yy,i}_{k|k-1}(\mathbf{y}_k^{(j)} - \boldsymbol{\mathcal{Y}}_{k|k-1}^{(i)})^T \tag{20}$$

An estimate with index i is updated with a measurement with index j if their common score r_{ij} is the minimum score of all the scores belonging to the measurement j and is smaller than some fixed threshold known as gate g. This procedure leads to a $2 \times l$ dimensional vector $\boldsymbol{\zeta}_k = \begin{bmatrix} \boldsymbol{\zeta}_{k,\mathbf{y}} & \boldsymbol{\zeta}_{k,\hat{\mathbf{y}}} \end{bmatrix}$ containing the indices of the l associated pairs of estimates i and measurements j.

If not all measurements have been related to an initialized estimate, those measurements that have erroneously not been related and the ones that arise from a newly detected cloud have to be distinguished. Therefore, for all of the measurements that have not been associated, it is checked if they attain the minimum score to any of the n estimates. In case this statement is false, measurement j is considered as a newly detected cloud and used to initializue a new cloud state. Otherwise, it is rejected. The indices of measurements that are used to initialize new clouds are stored in a vector $\boldsymbol{\xi}_k$.

Adjustment

As illustrated in fig. 5, the predicted quantities $\hat{\mathbf{y}}_k, \mathbf{P}^{yy}_{k|k-1}, \mathbf{P}^{xy}_{k|k-1}$ and the measurement vector $\mathbf{y}_k$ are adjusted by selecting the relevant elements (index vector $\boldsymbol{\zeta}$) which have been related to a measurement. Where the index e stands for effect (see fig. 5).

$$\mathbf{y}_{k,e} = \hat{\mathbf{y}}_k(\boldsymbol{\zeta}_{k,\mathbf{y}}) \text{ with } \boldsymbol{\zeta}_{k,\mathbf{y}} = \begin{bmatrix} j_1, \ldots, j_l \end{bmatrix}^T \text{ and } 0 \leq l \leq m$$

$$\hat{\mathbf{y}}_{k|k-1,e} = \hat{\mathbf{y}}_{k|k-1}(\boldsymbol{\zeta}_{k,\hat{\mathbf{y}}}) \text{ with } \boldsymbol{\zeta}_{k,\hat{\mathbf{y}}} = \begin{bmatrix} i_1, \ldots, i_l \end{bmatrix}^T$$

$$\mathbf{P}^{yy}_{k|k-1,e} = \mathbf{P}^{yy}_{k|k-1}(\boldsymbol{\zeta}_{k,\hat{\mathbf{y}}}, \boldsymbol{\zeta}_{k,\hat{\mathbf{y}}})$$

$$\mathcal{Y}_{k|k-1,e} = \mathcal{Y}_{k|k-1}(\boldsymbol{\zeta}_{k,y}, \boldsymbol{\alpha}) \text{ with } \boldsymbol{\alpha} = \begin{bmatrix} 1, \ldots n \end{bmatrix}$$

$$\mathbf{P}^{xy}_{k|k-1,e} = \mathbf{P}^{xy}_{k|k-1}(\boldsymbol{\alpha}, \boldsymbol{\zeta}_{k,\hat{\mathbf{y}}}) \tag{21}$$

Measurement augmentation

According to eq. (10), both the adjusted measurement $\mathbf{y}_{k,e}$ as well as the adjusted and predicted quantities $\hat{\mathbf{y}}_{k,e}, \mathcal{Y}_{k|k-1,e}$ vector are augmented using the state constraint on the uniform cloud base yielding

$$\mathbf{y}_{k,a} = \begin{bmatrix} \mathbf{y}_{k,e} \\ \mathbf{d}_k \end{bmatrix} \text{ and } \hat{\mathbf{y}}_{k,a} = \begin{bmatrix} \hat{\mathbf{y}}_{k|k-1,e} \\ \hat{\mathbf{d}}_k \end{bmatrix} \tag{22}$$

where $\hat{\mathbf{d}}$ containts the n variations from the individual cloud base z_i to the mean cloud base $\bar{z}$

$$d_i = z_i - z_c$$

Also, the measurement noise is augmented such that $\mathbf{R}_a = diag(\mathbf{R}, \mathbf{R}_1)$.
The prediction steps eqs. (17) to (19) are then recomputed for the augmented quantities yielding $\hat{\mathbf{y}}_{k,a}, \mathbf{P}^{yy}_{k|k-1,a}$ and $\mathbf{P}^{xy}_{k|k-1,a}$.

3.3.3 Correction

Using the adjusted predicted quantities as well as the adjusted measurement vector, the classical Kalman correction step is accomplished following

$$\mathbf{K}_k = \mathbf{P}^{xx}_{k|k-1}(\mathbf{P}^{yy}_{k|k-1,a})^{-1}$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k(\mathbf{y}_{k,a} - \hat{\mathbf{y}}_{k|k-1,a})$$

$$\mathbf{P}^{xx}_k = \mathbf{P}^{xx}_{k|k-1} - \mathbf{K}_k \mathbf{P}^{yy}_{k|k-1,a} \mathbf{K}^T_k \tag{23}$$

3.3.4 Cloud initialization and state augmentation

Each time a new cloud is detected, both its initial estimate $\mathbf{x}$ its error covariance $\mathbf{P}^{xx}$ have to be computed from only one measurement. The state initialization causes potential difficulties, because the data association is prone to errors in the covariance of the predicted estimate.
Clouds are assumed to have approximately the same base altitude. Therefore, it is straightforward to compute the initial state estimate by calculating the plane-line

intersection between the cloud base plane and the line-of-sight from the current vehicle position $\mathbf{p}$ along the bearing $\mathbf{b}$ to the new cloud, if some knowledge of the cloud base z_0 is given. For the very first cloud, an a-priori estimate of the cloud base z_0 is used. Subsequent clouds are initialized based on the actual estimated altitude. The initial cloud position ($\begin{bmatrix} x & y & z_0 \end{bmatrix}^T$) can be obtained by the function $\mathbf{s}$ with an input vector $\mathbf{m} = \begin{bmatrix} \mathbf{p} & \mathbf{q} & z_0 \end{bmatrix}^T$ as shown in eq. (24)

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{s}(\mathbf{m}) = \begin{bmatrix} p_x \\ p_y \end{bmatrix} + \mu \begin{bmatrix} b_x \\ b_y \end{bmatrix} \tag{24}$$

Where the bearing $\mathbf{b}$ and the magnitude μ are defined as:

$$\mathbf{b} = \mathbf{T}_{ce}^{-1} \begin{bmatrix} y_x \\ y_y \\ 1 \\ 1 \end{bmatrix} \qquad \text{and} \qquad \mu = \frac{z_0 - p_z}{b_z} \tag{25}$$

Unscented transformation proves again to be a convenient method to convert the measurement uncertainty $\mathbf{P}^m$ into an initial state covariance

$$\mathbf{P}^m = \begin{bmatrix} \sigma_{y_x}^2 & 0 & 0 \\ 0 & \sigma_{y_y}^2 & 0 \\ 0 & 0 & \sigma_{z_0}^2 \end{bmatrix} = \begin{bmatrix} \mathbf{R} & 0 \\ 0 & \sigma_{z_0}^2 \end{bmatrix} \tag{26}$$

where σ_{z_0} is the standard deviation of the a-priori knowledge on z_0. Defining the incoming measurement vector $\mathcal{M}_0$, the related $2L_i + 1$ sigma points result as

$$\mathcal{M} = \begin{bmatrix} \mathcal{M}_0 & \mathcal{M}_0 + \sqrt{L+\lambda}\sqrt{\mathbf{P}^m} & \mathcal{M}_0 - \sqrt{L+\lambda}\sqrt{\mathbf{P}^m} \end{bmatrix} \tag{27}$$

Where the state dimension is $L_i = 3$ when dealing with a single cloud. Each of the $2L + 1$ sigma points is instantiated through the initialization function $\mathbf{s}(\mathbf{m})$ which yields the matrix $\mathcal{O}$ containing seven 3D positions of the cloud

$$\mathcal{O}_j = s(\mathcal{M}_j) \tag{28}$$

Both the initial state estimate $\mathbf{x}_0$ and the state covariance $\mathbf{P}_0^{xx}$ are then obtained

$$\mathbf{x} = \bar{\mathbf{o}} = \sum_{j=0}^{2L} \eta_i^c \mathbf{o}_j \qquad \mathbf{P}^{xx} = \sum_{j=0}^{2L} \eta_j^c (\mathcal{O}_j - \bar{\mathbf{o}})(\mathcal{O}_j - \bar{\mathbf{o}})^T \tag{29}$$

As illustrated in fig. 5, the corrected state and covariance estimate are augmented with the state and covariance of the initialized clouds.

4 Simulation Results

3DOF simulations were conducted for the following two purposes: First, to demonstrate that the SPKF is able to provide a convergent and confident estimation of cumulus cloud positions that can reliably be used for path planning algorithms. Second, that the filter formulation with a soft state constraint based on the assumption of a uniform cloud base leads to faster and still confident convergence of both state and covariance estimation - even in case of strongly varying cumulus cloud bases.

4.1 Simulation scenario and settings

A forward looking camera was moved along a circular and climbing trajectory (as shown in fig. 6) for an observation duration of 300 s to simulate the estimation process during a standard thermaling flight where the clouds repeatedly appear and disappear on the image sensor due to the circular trajectory.
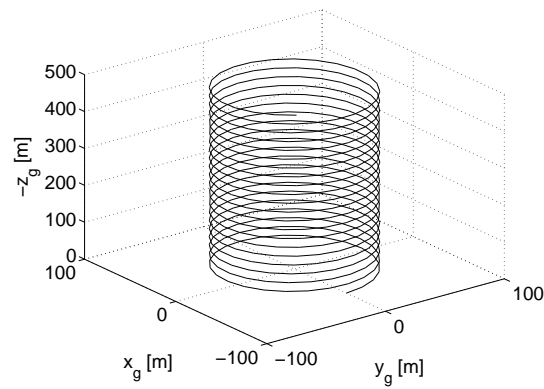
Fig. 6: Camera trajectory

Cumulus clouds were located around the center of the trajectory for two scenarios as depicted in fig. 7.Note that the camera's trajectory is the circle around the origin. It appears small due to the small scale of the map.

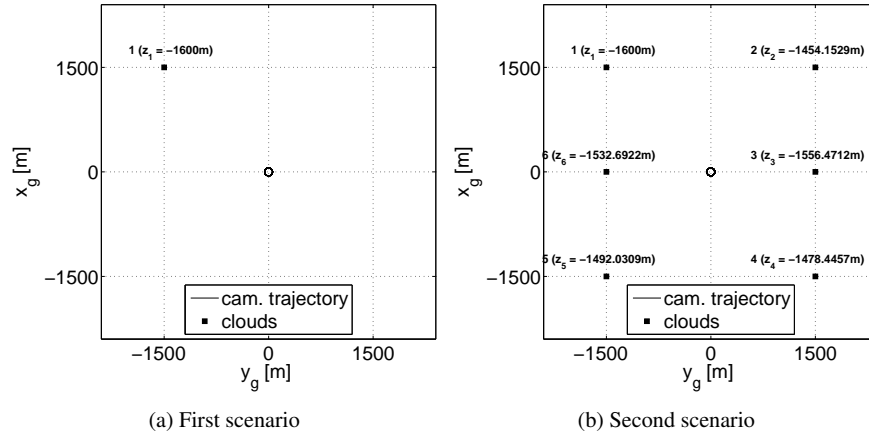(a) First scenario   (b) Second scenario

Fig. 7: Cloud position estimation scenarios

The first scenario (fig. 7a) was chosen to demonstrate the performance of an unconstrained filter formulation by observing only one cloud which suppresses the last n rows of eq. (10). In the second scenario (fig. 7b), six clouds were positioned around the center of the trajectory. In order to obtain realistic estimation results, the six cloud bases were normally distributed around a mean $\bar{z}_c = -1500\,m$ with a standard deviation of $\sigma_{z_c} = 25\,m$. Subsequently, the cloud base altitude of cloud number one was biased by $100\,m$ to simulate strong variations. Note that cloud number one has precisely the same horizontal position as the cloud in the first scenario.

The parameters of the camera model eq. (8) $\mathbf{C}$, i.e. the focal length, field of view and image size were selected to represent the performance of a small low-cost camera. A constant and unbiased measurement noise $\mathbf{v}$ with a standard deviation of $5\,\%$ of the image size was used to simulate uncertainties evoked by the image detection algorithm. Where the primary scaling parameter α which determines the spread of the sigma points was set to 1, the secondary scaling parameter β was set to 2 which according to [16] is the optimum value for Gaussian distributions. The tertiary parameter κ was set to a commonly-used value of 0. The process noise covariance $\mathbf{Q}$ was chosen to be $0.01\,(m/s)^2$. The initial cloud base z_0 can for example be obtained as a cloud ceiling provided in METAR/TAF information. In this simulation it was set to a value of $z_0 = -1200\,m$ with a significant uncertainty $\sigma_{z_0} = 600\,m$. The gate g in the data association was set to 9.

As discussed in section 2, the soft state constraint enables faster convergence of cloud position estimates. This is because, even if a cloud is out of the field of view, its position estimate can be corrected based on the state constraint measurement (last n rows in eq. (10)). However, the position estimation will be biased and/or the error interval will be predicted too tight, if the assumption of a uniform cloud base does not hold for a particular cloud, i.e. the state constraint is biased. A too small covariance $\mathbf{R}_1$ will reduce the slackness of the constraint and cause a fast overconfidence during estimation. Therfore, the parameter $\mathbf{R}_1$ has to be selected carefully depend-

ing on cloud base variations that can be encountered in the real world. That being said, $\mathbf{R_1}$ was tuned with the second scenario such that the filter ensures estimation confidence for clouds with an altitude variation of up to 100 m. This value roughly corresponds to the maximum the main author has observed during various cross-country soaring flights and has been confirmed by a meteorologist. The procedure lead to a parameter value of $\mathbf{R}_1 = 750000 \, m^2$.

4.2 Estimation performance

The position estimate of the first cloud is considered to compare the filter's performance in terms of confidence and convergence for the unconstrained (noted *unc.* in fig. 8) and *c.* for the constrained formulation as depicted in fig. 8. Recall that this corresponds to the estimation performances of the two defined scenarios. While the dark envelope and the dashed line denote the 3σ envelope and the position error for the unconstrained case, the light envelope and the solid line correspond to the 3σ envelope and the estimation error when considering the state constraint.
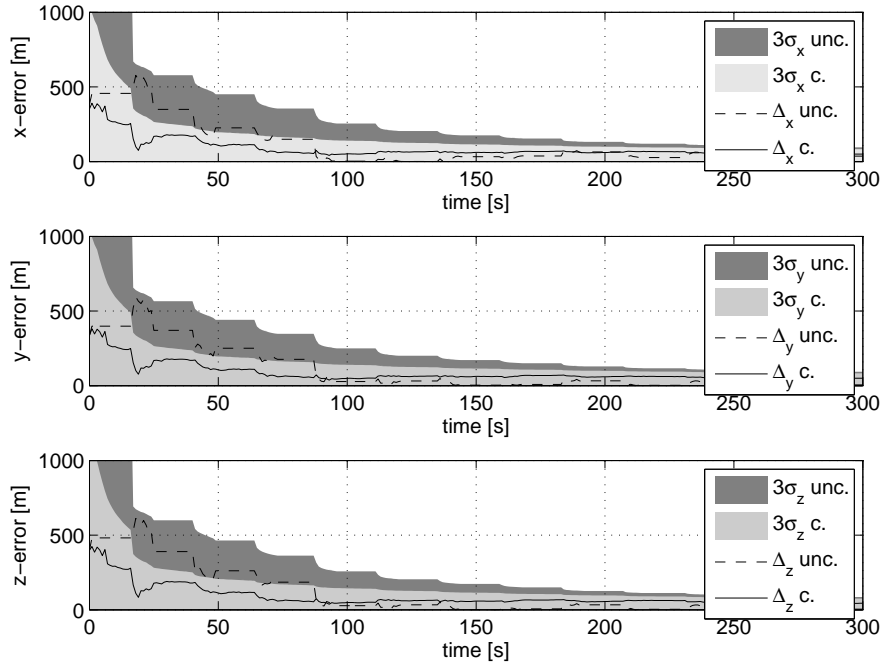

Fig. 8: Estimation performance

As expected, both regarding rate of convergence and the error, the SCSPKF out-performs the standard SPKF. In all three cases, the upper bound of the position error is reliably predicted. The huge uncertainty of the initial cloud base impacts the filter's transient behavior which can be seen in terms of estimation overshoots in the beginning of the estimation process. Also, the settling time for the unconstrained filter process is extended since each cloud is visible only for approximately 35 % of the estimation duration. Periods with no measurements can be seen at the long horizontal segments within the graph. In contrast, for the constrained formulation a significant reduction of the state's settling time is obtained due to the measurement augmentation.

Estimation degradation is expected whenever the vertical position of a cloud strongly varies from the mean and its position update is only performed using augmented measurements. This is particularly the case for clouds that lie behind the camera's field of view when flying towards the next waypoint in cross-country soaring. However, this degradation is not predominant, since clouds that lie behind the vehicle have no impact on the future path.

5 Perspectives

This paper presented a method providing functionality to the remote sensing of thermal updrafts. The information of both state and covariance estimation should be taken into account in path planning algorithms to enable more efficient autonomous cross-country soaring.

In general, the problem of autonomous cross-country soaring can be stated as a waypoint navigation (from A to B) as illustrated in fig. 9.
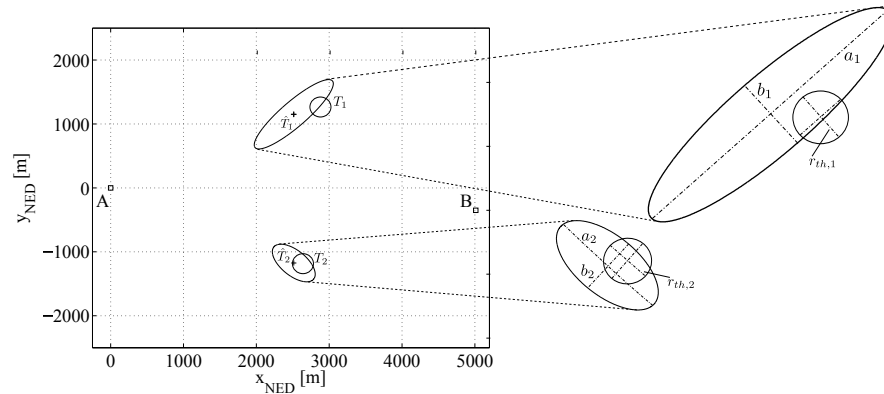

Fig. 9: Path planning problem

In this example, an unpowered UAV glider has to fly from waypoint A to waypoint B given position estimates for the two thermal updrafts T_1 and T_2, where the true thermal centers are supposed to lie somewhere within the 2D error ellipses of the estimates. The mission starts at waypoint A where the UAV is scanning the sky for thermals while climbing in a thermal before planning the path to the next intermediate or global target B. The ability of the glider to perform this mission in minimum time depends on three factors. Firstly, the vehicle's performance in terms of its glide ratio i.e. its capacity to transform potential energy into travelled distance. Secondly, the flight control's performance to center around a given thermal. Thirdly and most importantly meteorological conditions and the pilot's capacity to read them i.e. to locate far away thermals in order to plan the most efficient path.
If the direct path from A to B is not feasible due to the vehicle's limited glide ratio, it has to fly a detour via one of the two thermals to regain altitude. The total flight time for the two path options is given by

$$
\begin{aligned}
(A \to T_1 \to B): \quad & t_{AB} = t_{AT_1} + \underbrace{t_{en,T_1} + t_{th,T_1}}_{\text{time spent at } T_1} + t_{T_1B} \\
(A \to T_2 \to B): \quad & t_{AB} = t_{AT_2} + \underbrace{t_{en,T_2} + t_{th,T_2}}_{\text{time spent at } T_2} + t_{T_2B}
\end{aligned}
\tag{30}
$$

where the encounter time t_{en} is the time to hit the thermal while searching within the error ellipse, and the thermal time t_{th} is the time spent in the thermal updraft during climb. The latter depends on the initial altitude at which the vehicle enters the thermal and the strength of the updraft as well as the cloud base. The vehicle is supposed to leave the thermal once the cloud base is reached. Assuming equal thermal strength, the two routes seem to be on par regarding the flight time. However, the larger position uncertainty of T_1 might require more time to encounter the thermal using some search pattern whose size is determined by the error ellipse. That being said, the uncertainty of the thermal position impacts the flight time. According to Allen's research on modeling thermal updrafts for autonomous UAV soaring [4], the thermal radius r_{th} as illustrated in fig. 9 can be predicted when knowing z_c and the altitude z at which the vehicle reaches the thermal

$$
r_{th} = 0.5 \left[0.203(\frac{z}{z_c})^{\frac{1}{3}}(1 - 0.25\frac{z}{z_c})z_c \right]
\tag{31}
$$

Whenever the thermal radius r_{th} is larger than the half of the semi minor b belonging to the 2D error ellipse (see $\hat{T}_1$ in fig. 9), the maximum time to encounter the thermal can be predicted by the speed V (which is considered to be constant during operation) and the semi major a

$$
t_{en} = \frac{a}{V}
\tag{32}
$$

Otherwise, a systematic search pattern has to be flown within the error ellipse. Regardless of the pattern's shape, the maximum time to encounter the thermal is

$$t_{en} = \frac{l_p}{V} \text{ where } l_p = l_p(\mathbf{P}_T, p) \tag{33}$$

where the pattern length l_p depends on the uncertainty $\mathbf{P}_T$ as well as on the shape of the pattern p.

These upper bounds on t_e render it possible to incorporate the uncertainty of thermal position estimates into the cost function thus reducing the total flight time.

Future work will concentrate on two fields. First, the design of path planning algorithms for autonomous cross-country soaring including crucial meteorological aspects as thermal updrafts and wind. Second, the design of image processing algorithms capable of deducing information about thermals given images of clouds. This includes for instance thermal strength prediction based on the color and contrast of the related cumulus as well as cloud size and shape. If those visible features can be detected, and thermal strength can fairly be predicted, even more efficient path planning becomes possible by taking into account this additional information.

6 Conclusion

In this paper, a SCSPKF was developed for remotely sensing thermal updrafts indicated by cumulus clouds in autonomous soaring. Two design efforts were focused on:

- Including the state constraint of a uniform cumulus cloud base for faster convergence
- Maintaining estimation consistency and providing a confident estimate of the uncertainty

Simulation results clearly demonstrate the benefits of the constrained filter formulation in terms of convergence rate. The filter still provides consistent estimation for strong model deviations with biases in the cumulus cloud base of up to 100 m.

Finally, a perspective for a new path planning approach for autonomous cross-country soaring was presented considering the uncertainty of thermal position estimates to augment the efficiency of future UAV operations.

References

[1] M. J. Allen and V. Lin, "Guidance and Control of an Autonomous Soaring UAV with Flight TTest Results," in *45th AIAA Aerospace Sciences and Meeting and Exhibit*, American Institute for Aeronautics and Astronautics (AIAA), January 2007.

[2] K. Andersson, I. Kaminer, V. Dobrokhodov, and V. Cichella, "Thermal Centering Control for Autonomous Soaring; Stability Analysis and Flight Test Re-

sults," *Journal of Guidance Navigation and Control*, vol. 35, pp. 963–975, 2012.

[3] Naseem Akhtar and James F Whidborne and Alastair K Cooke, "Real-time trajectory generation technique for dynamic soaring UAVs," in *Proceedings of the UKACC International Conference on Control*, 2008.

[4] M. J. Allen, "Updraft model for development of autonomous soaring uninhabited air vehicles," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, American Institute for Aeronautics and Astronautics (AIAA), January 2006.

[5] D. J. Edwards and L. M. Silberberg, "Autonomous Soaring: The Montague Cross-Country Challenge," *Journal of Aircraft*, vol. 47, pp. 1763–1769, 2010.

[6] N. Kahveci, *Robust Adaptive Control For Unmanned Aerial Vehicles*. PhD thesis, University of Southern California - Faculty of the Graduate School, USA, 2007.

[7] John Canny, "A Computational Approach to Edge Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 679–698, November 1986.

[8] E. Kleinschmidt, *Handbuch der Meteorologischen Instrumente und ihrer Auswertung*. Verlag von Julius Springer, 1935.

[9] Dennis Pagen, *Understanding the sky*. Dennis Pagen, February 1992.

[10] T. K. Yaakov Bar-Shalom, X. Rong Li, *Estimation with Applications To Tracking and Navigation*. John Wiley and Sons, Inc., 2001.

[11] S. S. Ponda, "Trajectory Optimization for Target Localization Using Small Unmanned Aerial Vehicles," Master's thesis, Massachusetts Institute of Technology, September 2008.

[12] S. Julier, "A skewed approach to filtering," in *In SPIE Conference on Signal and Data Processing of Small Targets*, vol. 3373, pp. 271–282, SPIE, April 1998.

[13] S. J. Julier and J. K. Uhlman, "A New Extension of the Kalman Filter to Nonlinear systems," in *Proc. SPIE 3068, Signal Processing, Sensor Fusion, and Target Recognition VI*, April 1997.

[14] R. V. der Merwe, "The square-root unscented Kalman filter for state and parameter estimation," in *Acoustics, Speech, and Signal Processing, 2001. Proceedings. (ICASSP '01). 2001 IEEE*, vol. 6, pp. 3461–3464, May 2001.

[15] R. V. der Merwe and E. Wan, "The unscented Kalman filter for nonlinear estimation," in *Adaptive Systems for Signal Processing, Communications, and Control Symposium 2000. AS-SPCC. The IEEE 2000*, pp. 153–185, IEEE, October 2000.

[16] M. Rhudy and Y. Gu, "Understanding Nonlinear Kalman Filters, Fart II: An Implementation Guide," *Interactive Robotics Letters*, 2013.

[17] J. W. Langelaan, *State estimation for autonomous flight in cluttered environments*. PhD thesis, Stanford University, March 2006.

[18] M. Montemerlo and S. Thrun, *FastSLAM: A Scalable Method For The Simulations Localization and Mapping Problem in Robotics*. Springer, 2007.