Thermal localization

Ricardo Bencatel  
Department of Electrical and Computer Engineering  
University of Porto - School of Engineering  
Porto - Portugal  
ricardo.bencatel@fe.up.pt

Abstract—This paper develops a particle filter locator for atmospheric thermal flows. The underlying thermal model is more detailed and general than the Gaussian models typically used. The implemented particle filter is a regularized and adaptive version. This filter handles thermals global localization and the underlying non-linear models, providing excellent results.

Index Terms—Thermal localization; Particle Filter; Unmanned Aerial Vehicle; Soaring

I. INTRODUCTION

An important issue for aircraft operation is the energy consumption. Aircraft can use updrafts to diminish fuel consumption, in particular small Unmanned Air Vehicles (UAVs), with slow flight airspeeds and good maneuverability. These updrafts can originate from terrain topography or from thermal flows. The first type, called orographic updrafts, are generated when the wind hits a terrain slope, originating strong updrafts above the terrain. The thermal flows are generated by hot spots on the ground. These create a thermal gradient, heating the surrounding air. The density of the heating air decreases, forcing the upward movement. Thermal updrafts don’t depend as much on topography or on wind as the orographic ones.

A. General Problem

Updraft effects can be detected by aircraft. The energy balance can be computed through the aircraft sensing data. This allows us to quantify how much air flow energy is being conveyed to the aircraft, particularly on the vertical axis. As such, the aircraft can be regarded as a sensor, and the thermals as the objects whose parameters and state have to be estimated. This is a global localization problem. Further, the nonlinear nature of the problem is emphasized by the necessity to estimate the thermal shape and strength. Unlike hydrothermal plume sources [1], air thermals are fairly large features, with a diameter between 100m and 1000m [2], and their energy manifestations are not larger than themselves.

B. Literature Review

Lawrance and Sukkarieh [3] presented an energy based method for soaring path planing. This method is suitable for static and dynamic soaring, i.e., both in thermals and shear flow fields. They present a toroidal model for the thermal’s flow structure. Allen [4] presents an updraft model developed at NASA Dryden Flight Research Center with field measurements. This work makes use of this thermal model. Regarding the localization of thermals Allen [5] presented a centroid-based method for dynamic localization of the thermal center. The system was demonstrated in flight tests over Edwards Air Force Base. The algorithm had some problems with filtering delays, resulting in considerable thermal center localization errors [6]. Edwards [6] extended Allen’s approach, mixing it with Wharington’s [7] neural-network based locator. The flight results were quite good, demonstrating highly effective thermal energy harvesting. Both methods assumed the estimation of a single thermal at each time.

C. Paper Structure

We present the thermal model in section II. The model is divided in regional and individual characteristics. Next, we describe the UAV sensing method and the measurement computation (section III). Section IV describes the inference framework, the standard particle filter, the adaptive and regularized versions. In section V, we present the propagation model and the observation model used by the particle filter. Sections VI, VII, VIII, and IX present the simulation process, the results obtained, the conclusions and future work.

II. THERMAL DYNAMICS

A. Regional Characteristics

The two regional parameters for the thermals are the mixing-layer thickness and the convective velocity scale [4],[8]. The mixing-layer thickness ($z_i$ - fig. 1) is the maximum altitude for the thermal activity. The convective velocity scale ($w^*$) is a reference which indicates the predicted velocity magnitudes in and around a thermal. Both parameters vary slowly with the time and location. As such, we assume that one can disregard the dependency on the time of day and merge both sources of variation into a single source of uncertainty. The only extra factor affecting $w^*$ is the wind speed, which seems to disrupt any thermal if blowing above 12.87m/s (on the Mojave Desert [4]). In [4] the author describes the yearly statistics for $w^*$ and $z_i$. The data indicates both parameters can

Fig. 1. Updraft representation with mixing-layer thickness ($z_i$) (illustration from [4])
be modeled by Gamma distributions with averages 2.56m/s and 1401m, respectively.

**B. Individual Thermal Characteristics**

1) **Outer radius**: Lenschow and Stephens [8] state that the thermal radius average at a certain height is a direct function of $z_i$ by (fig. 2):

$$ r_2 = \max \left[ 10, 0.102 \left( \frac{z}{z_i} \right)^{\frac{1}{2}} \left( 1 - 0.25 \frac{z}{z_i} \right) \cdot z_i \right] $$

2) **Vertical velocity**: Lenschow and Stephens [8] state that the magnitude of vertical velocity variation within a thermal may be larger than the magnitude of the mean updraft velocity term. The reference updraft velocity has a bell shape (fig. 3) modeled by [4]:

$$ w = w_{peak} \left( \frac{1}{1 + k_1 \frac{r}{r_2} + k_2 \frac{r}{r_2} + k_3 \frac{r}{r_2} + w_D} \right) \times \ldots$$

$$ \ldots \times \left( 1 - \frac{w_c}{w_{peak}} \right) + w_c. $$

The constants $k_1, k_2, k_3,$ and $k_4$ are defined on table II-B2. The variables $w_{peak}, w_D, w_c,$ and $r_1$ are defined next [4]. The average updraft speed function (fig. 4) is:

$$ \overline{w} = w^* \left( \frac{z}{z_i} \right)^{\frac{1}{2}} \left( 1 - 1.1 \frac{z}{z_i} \right). $$

**TABLE I**

<table>
<thead>
<tr>
<th>$z_i$</th>
<th>$k_1$</th>
<th>$k_2$</th>
<th>$k_3$</th>
<th>$k_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.14</td>
<td>1.5352</td>
<td>2.5826</td>
<td>-0.0113</td>
<td>0.0008</td>
</tr>
<tr>
<td>0.25</td>
<td>1.5265</td>
<td>3.6054</td>
<td>-0.0176</td>
<td>0.0005</td>
</tr>
<tr>
<td>0.36</td>
<td>1.4866</td>
<td>4.8354</td>
<td>-0.0220</td>
<td>0.0001</td>
</tr>
<tr>
<td>0.47</td>
<td>1.2042</td>
<td>7.9004</td>
<td>0.0848</td>
<td>0.0001</td>
</tr>
<tr>
<td>0.58</td>
<td>0.8816</td>
<td>13.972</td>
<td>0.3404</td>
<td>0.0001</td>
</tr>
<tr>
<td>0.69</td>
<td>0.7067</td>
<td>23.994</td>
<td>0.5689</td>
<td>0.0002</td>
</tr>
<tr>
<td>0.80</td>
<td>0.6189</td>
<td>42.797</td>
<td>0.7157</td>
<td>0.0001</td>
</tr>
</tbody>
</table>

Now, the radius for which the updraft speed is almost constant ($r_1$) comes from

$$ \frac{r_1}{r_2} = \begin{cases} 0.0011 r_2 + 0.14 & \text{if } r_2 < 600 \\ 0.8 & \text{otherwise} \end{cases}. $$

The maximum updraft speed ($w_{peak}$) is defined as

$$ w_{peak} = \frac{3r_2^2 (r_2 - r_1)}{r_2^3 - r_1^3}. $$

The skirt downdraft speed ($w_D$) is computed by

$$ w_D = \begin{cases} \frac{3\pi}{12} \left( \frac{z}{z_i} - 0.5 \right) \sin \left( \frac{\pi r_2}{r_2} \right) & \text{if } r \in (r_1, 2r_2) \\ 0 & \text{otherwise} \end{cases}. $$

To maintain a null regional net vertical velocity, we have to define the natural sink speed ($w_c$) as:

$$ w_c = -\frac{N \pi r_2^2}{A_{reg} - N \pi r_2^2} \ldots$$

$$ \ldots \begin{cases} 1 - 2.5 \left( \frac{z}{z_i} - 0.5 \right) & \frac{z}{z_i} \in (0.5, 0.9) \\ 1 & \text{otherwise} \end{cases}, $$

where $A_{reg}$ is the affected region.

3) **Drift speed**: The drift speed is not always the same as the prevailing wind. Because the wind fully disruptive speed is around 13m/s, we assume the thermals drift speed is constrained to a maximum of 6m/s.
III. UAV THERMAL SENSING

The UAV is affected by the thermal local vertical flow. As such, it may estimate the thermal state through its energy measurements. The variation of the aircraft energy depends on the engine model, the drag at the current speed and attitude, the aircraft acceleration, and the altitude variation. All these variables incur estimation errors, in particular the altitude change.

It is assumed that the UAV will use its autopilot pose estimate to localize the energy measurements. The pose estimate \( \mathbf{p} = [\dot{x}, \dot{y}, \dot{z}, \dot{\phi}, \dot{\theta}, \dot{\psi}, \dot{p}, \dot{q}, \dot{r}]^T \) is a vector with the position and velocity in \( x, y, z \), and the Euler angles and angular rates. This estimation is obtained by fusing GPS, IMU, and pressure sensor data, which tends to yield low relative error in the short term. Because of that, and because we are not trying to enhance the UAV position estimate, we will consider that estimate as ground truth for the measurement. In addition to the pose, several UAV measurements or parameters are used to compute the air flow energy rate.

We’ll now deduce the measurement equation in terms of UAV on-board sensor measurements.

A. Frames and Angles

Two reference frames are important for the UAV dynamics definition. The fixed reference frame is centered at the origin, with the \( x \) axis pointing to North, the \( y \) axis pointing to East, and the \( z \) axis pointing to the Earth center. The aircraft or body reference frame is fixed to the aircraft center of mass, with the \( x \) axis pointing to the nose, the \( y \) axis pointing to the right wing tip, and the \( z \) axis pointing down. The three important air relative angles are the air-climb, the angle-of-attack, and the sideslip angle. The air-climb angle equation is:

\[
\gamma_a = \arctan \frac{\dot{z}_V - W_z}{\sqrt{\left(\dot{x}_V - W_x\right)^2 + \left(\dot{y}_V - W_y\right)^2}},
\]

which is important to define the velocity equations:

\[
\begin{bmatrix}
\dot{x}_V \\
\dot{y}_V \\
\dot{z}_V
\end{bmatrix} = \begin{bmatrix}
\dot{x}_a \\
\dot{y}_a \\
\dot{z}_a
\end{bmatrix} + \mathbf{w} = \ldots
\]

\[
\ldots = V_a \begin{bmatrix}
\cos \psi \cos \gamma_a \\
\sin \psi \cos \gamma_a \\
\sin \gamma_a
\end{bmatrix} + \begin{bmatrix}
W_x \\
W_y \\
W_z
\end{bmatrix},
\]

where \([\dot{x}_V, \dot{y}_V, \dot{z}_V]^T\) is the velocity vector on the fixed reference frame, \([\dot{x}_a, \dot{y}_a, \dot{z}_a]^T\) is the velocity vector relative to the air (flow field), \([W_x, W_y, W_z]^T\) is the wind velocity vector on the fixed reference frame, and \(V_a\) the air relative total speed.

The angle-of-attack is defined by \(\alpha = \theta - \gamma_a\), and is important to set the aerodynamic coefficients and the transformation of the vector \(\vec{V}\) to the body-fixed axis. The sideslip angle (\(\beta\)) will be regarded as null, as that is one of the autopilot assignments.

B. UAV Dynamics

Let us now take the simpler case where the UAV is constrained to the x-z plane, i.e., \(\psi = 0\). The equations of motion governing the UAV are:

\[
m\ddot{x}_V = -L \sin \gamma_a + (T - D) \cos \gamma_a,
\]

\[
m\ddot{z}_V = -L \cos \gamma_a - (T - D) \sin \gamma_a + mg,
\]

where \(L\) and \(D\) are the aerodynamic lift and drag, \(T\) is the propulsion thrust, \(m\) is the aircraft mass, and \(g\) is the gravity acceleration. If we differentiate \(\dot{x}_V\) and \(\dot{z}_V\) from (9), we obtain,

\[
\begin{bmatrix}
\dot{x}_V \\
\dot{z}_V
\end{bmatrix} = \frac{dV_a}{dt} \begin{bmatrix}
\cos \gamma_a \\
-\sin \gamma_a
\end{bmatrix} + \ldots
\]

\[
\ldots - V_a \begin{bmatrix}
\sin \gamma_a \\
\cos \gamma_a
\end{bmatrix} \frac{d\gamma_a}{dt} + J_W \begin{bmatrix}
\dot{x}_V \\
\dot{z}_V
\end{bmatrix},
\]

where \(J_W\) is the Jacobian of the wind speeds:

\[
J_W = \begin{bmatrix}
\frac{d\dot{x}_V}{dw} & \frac{d\dot{x}_V}{dx}
\end{bmatrix}.
\]

Combining (12) with the equations of motion (10) and (11):

\[
\begin{bmatrix}
\frac{dV_a}{dt} - \frac{T - D}{m}
\end{bmatrix} \begin{bmatrix}
\cos \gamma_a \\
-\sin \gamma_a
\end{bmatrix} + J_W \begin{bmatrix}
\dot{x}_V \\
\dot{z}_V
\end{bmatrix} = \ldots
\]

\[
\ldots = \left(V_a - \frac{L}{m}\right) \begin{bmatrix}
\sin \gamma_a \\
\cos \gamma_a
\end{bmatrix} \frac{d\gamma_a}{dt} + \begin{bmatrix}
0 \\
g
\end{bmatrix}.
\]

Solving for \(\frac{dV_a}{dt}\) yields,

\[
\frac{dV_a}{dt} = \frac{T - D}{m} - g \sin \gamma_a - \begin{bmatrix}
\cos \gamma_a \\
-\sin \gamma_a
\end{bmatrix}^T J_W \begin{bmatrix}
\dot{x}_V \\
\dot{z}_V
\end{bmatrix}.
\]

C. UAV Energy Dynamics

The total energy is

\[
E = -mgz_V + \frac{1}{2} mV_a^2
\]

The derivative yields:

\[
\dot{E} = -mg \dot{z}_V + mV_a \frac{dV_a}{dt} = \ldots
\]

\[
\ldots = mg \left(V_a \sin \gamma_a - W_z\right) + mV_a \frac{dV_a}{dt}
\]

Substituting \(\frac{dV_a}{dt}\) from (15) in (17), the altitude-airspeed energy transfer terms cancel each other, resulting in:

\[
\dot{E} = -mgW_z + V_a (T - D) - \ldots
\]

\[
\ldots - mV_a \begin{bmatrix}
\cos \gamma_a \\
-\sin \gamma_a
\end{bmatrix}^T J_W \begin{bmatrix}
\dot{x}_V \\
\dot{z}_V
\end{bmatrix}
\]

The predicted nominal energy rate, which depends on the propulsion setting and the UAV attitude, is \(\dot{E}_n = V_a (T - D)\). The explicit UAV data output is defined in terms of engine power (\(P\)), air-relative speed (\(V_a\)), and the drag (\(D\)), yielding:

\[
\dot{E}_n = P - V_a \cdot D.
\]
D. UAV Energy Measurement

Now, we define \( \dot{E}_a \) as the UAV energy rate variation caused by the air flow, i.e., the difference between the total energy rate (17) and the predicted nominal energy rate (19):

\[
\dot{E}_a = -mg\dot{z}_V + mV_a \frac{dV_a}{dt} - P + V_a \cdot D. \tag{20}
\]

We can approximate the speed time derivative, by \( \frac{dV_a}{dt} \approx \ddot{x}_B \cos \alpha \), if the wind speed time derivative is small, and because the sideslip-angle (\( \beta \)) is assumed null. \( \ddot{x}_B \) is the acceleration in the UAV \( x \) axis. Now, the measurement of \( E_a \) may be computed directly from

\[
\dot{E}_{a,UA} = m \left( V_a \ddot{x}_B \cos \alpha - g\dot{z} \right) - P + V_a \cdot D. \tag{21}
\]

IV. INFERENCE FRAMEWORK

A. Particle Filter (PF)

In this work, the estimation is implemented through a Particle Filter [9]. The belief distribution at each estimation step is represented by particles with 6 state variables. The thermal center state vector \( x = [x_T, y_T, u_T, v_T]^T \), where \( x_T \) and \( y_T \) are the center position coordinates and \( u_T \) and \( v_T \) are the respective velocity components. And the mixing-layer state vector \( y = [z_i, w^*_i] \). Each particle is an hypothesis of the current state. At each step, particles are propagated, evaluated and resampled, to create a new estimate. The particles are propagated through the propagation model described in section V.

The observation model, also presented in section V, provides a measurement of likelihood of the hypothesis represented by each particle. The resampling prunes the unlikely particles (hypothesis).

In this implementation, the standard particle filter suffers from degeneracy problem, i.e., the particles converge to an hypothesis that is not the correct one. This happens due to the small number of particles used to represent the initial belief distribution, when compared to the state size.

B. Adaptive Particle Filter (APF)

Thrun et al. [9] present an adaptive version of the Particle Filter in section 8.3.5. This version is able to cope with the global localization problem. It uses a combination of resampled particles and random particles at each update step. In this work, the method was used to increase the sample diversity when the quality of the estimate was decreasing. It is assumed that the particles’ average likelihood evaluates the estimation quality at each step. This is used as a driver to set the relation between resampled and random particles.

C. Regularized Particle Filter (RPF)

Ristic et al. [10] describes the Regularized Particle Filter on section 3.5.3. This method was developed to reduce the degeneracy problem. The RPF does not resample directly from the propagated particle set, which is a discrete approximation of the prior. RPF resamples from a continuous approximation of the posterior, by actively jittering the resampled particle values:

\[
x^i_t = \hat{x}^i_t + h_{opt} \cdot D_i \epsilon_i \tag{22}
\]

where \( \epsilon_i \) is a sample from a Gaussian distribution, \( D_i \) is such that \( D_i D_i^T = S_i \), \( S_i \) is the empirical covariance matrix, and \( h_{opt} = \frac{1}{2} \left( \frac{N}{\pi n_x + 1} \right)^{\frac{1}{2}} \) is the optimal Gaussian kernel bandwidth, function of the state dimension (\( n_x \)) and the number of particles \( N \). On the current implementation the kernel bandwidth was set to \( h = G \cdot \sqrt{\frac{1}{2 N h_{opt}}} \). The tuning gain \( G \) was needed, because the initial distribution of particles was leading to a divergence of the filter.

D. Regularized Adaptive Particle Filter (RAPF)

The current implementation mixes both Regularized and Adaptive methods. The regularization increases sample diversity, allowing a smaller generation of random particles by the adaptive part. This leads to a stronger convergence, yet enabling the filter to avoid wrong convergences.

E. Particle Generation

As this is a global localization problem, no single initial state can be assumed. As such, a particle generation function was developed. This is used when the UAV first detects a probable thermal and when random particles are injected. Three rules define the particle generation. The new particles’ position \( (x_T, y_T) \) distribution is uniform on a bounded radius around the aircraft. The drift velocity \( (u_T, v_T) \) distribution is also bounded by the drift speed constraint. The mixing-layer altitude \( z_i \) and the convective velocity scale \( w^*_i \) are sampled from Gamma distributions with average 1401m and 2.56m/s, respectively. Figure 5 illustrates the typical particle generation distribution.

V. INFERENCE MODELS

A. Propagation Model

1) Thermal center: The thermal center motion dynamics are described by:

\[
x_t = A_t \cdot x_{t-1} + \epsilon_T, \tag{23}
\]

with \( A_t = \begin{bmatrix} 1 & 0 & \frac{1}{2} \lambda t & 0 \\ 0 & 1 & 0 & \frac{1}{4} \lambda t^2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \), \( \epsilon_T = \begin{bmatrix} 0 \\ \epsilon_u T \\ \epsilon_v T \\ \epsilon_{\delta_{Urft}} \end{bmatrix} \) (the thermal center disturbance vector), \( \sqrt{\epsilon_{\delta_{Urft}}^2 + \epsilon_u T^2} \sim N \left( 0, \delta_{Urft} \right) \), and \( \delta_{Urft} \) is the drift velocity change rate.
2) Mixing-layer change: The mixing-layer dynamics are described by:

\[ y_t = y_{t-1} + \varepsilon_M \]  \hspace{1cm} (24)

with \( \varepsilon_M = [\varepsilon_{z_i}, \varepsilon_{w_i}]^T \), \( \varepsilon_{z_i} \sim \mathcal{N}(0, \delta_{z_i}^2) \), and \( \varepsilon_{w_i} \sim \mathcal{N}(0, \delta_{w_i}^2) \) (the mixing-layer disturbance vector), and \( \delta_{z_i}, \delta_{w_i} \) are the Mixing-layer altitude and convective velocity scale change rates, respectively.

B. Observation Model

The updraft energy rate observation (\( \hat{E}_a \)) may be described by:

\[ \hat{E}_a = \tilde{E} - \hat{E}_a \approx h(x, y, p) + \varepsilon_{E}. \]  \hspace{1cm} (25)

This can be derived from (18) and (19), and expanded to the general 6DOF case is done by expanding \( x \) to the \( XY \) plane, yielding:

\[ h(x, y, p) = -mgW_z - mV_a \left[ \begin{array}{c} \cos \psi \cos \gamma_a \\ \sin \psi \cos \gamma_a \\ -\sin \gamma_a \end{array} \right]^T J_W \left[ \begin{array}{c} \dot{x}_V \\ \dot{y}_V \\ \dot{z}_V \end{array} \right], \]  \hspace{1cm} (26)

where the wind speed Jacobian is now:

\[ J_W = \left[ \begin{array}{ccc} \frac{\partial W_x}{\partial W} & \frac{\partial W_y}{\partial W} & \frac{\partial W_z}{\partial W} \\ \frac{\partial W_x}{\partial x} & \frac{\partial W_y}{\partial y} & \frac{\partial W_z}{\partial y} \\ \frac{\partial W_x}{\partial z} & \frac{\partial W_y}{\partial z} & \frac{\partial W_z}{\partial z} \end{array} \right]. \]  \hspace{1cm} (27)

The vertical wind velocity is the thermal updraft speed (\( W_z = w \)) calculated from equations (3), (4), (5), (6), and (7), with \( r = \sqrt{(x_T - x_V)^2 + (y_T - y_V)^2} \) and the altitude in the thermal \( z = z_V \). In this case we are assuming constant horizontal wind speed, yielding:

\[ h(x, y, p) = mV_a \sin \gamma_a \left[ \begin{array}{c} \frac{\partial W_x}{\partial \theta} \\ \frac{\partial W_y}{\partial \theta} \\ \frac{\partial W_z}{\partial \theta} \end{array} \right]^T \left[ \begin{array}{c} \dot{x} \\ \dot{y} \\ \dot{z} \end{array} \right] - mgW_z. \]  \hspace{1cm} (28)

The observation uncertainty (\( \varepsilon_{E} \)) has two components. The one due to thermal updraft variability has a similar magnitude to that of \( h \). The uncertainty caused by the UAV sensing noise and the wind disturbances is constant. As such we can model \( \varepsilon_{E} \) as:

\[ \varepsilon_{E} = \varepsilon_{E, Thermal} + \varepsilon_{E, Sens+W ind} \Leftrightarrow \varepsilon_{E} \sim \mathcal{N}(0, K_1 \cdot h(x, y, z) + K_2). \]  \hspace{1cm} (29)

For the simulation we used \( K_1 = 0.5 \) and \( K_2 = 2m. \)

VI. SIMULATION

A simulator was built to test the estimator. The UAV is simulated by a unicycle model with altitude change. The UAV is not commanded to circle inside the thermal, but rather to execute random passes inside the thermal from time to time. This shows that the estimator doesn’t require the UAV to be always inside the thermal nor any constrained trajectory. The waypoint controller randomly selects waypoints in a grid moving with the center of the simulated thermal. This grid was a square with 100 by 100 meters. The thermal simulation includes drift direction and speed noise. The mixing-layer parameters (\( Z_i \) and \( w^* \)) noise was simulated by a Random Walk. Additionally, the updraft speed was also affected by a gain factor, which was set by a Scalar Gauss-Markov Process. No other measurement noise was added.

VII. RESULTS

Table II compares the various filters’ performance. As indicated the standard Particle Filter (PF) is not a good solution for this problem since it would need a huge number of particles to present a reliable convergence. The Adaptive Particle Filter (APF) doesn’t seem a good alternative, although its estimates fall around the true state. The average estimation error is close to the average thermal radius, providing poor information. Further, the estimate is very unstable, jumping all around the true state. The Regularized Particle Filter (RPF) is a better solution. With a low quantity of particles (600) it is not very reliable, often suffering from the same problem as the standard PF. With more particles the confidence level on a correct lock-on is also higher, showing good reliability with 2000 particles. The Regularized Adaptive Particle Filter (RAPF) seems the best choice. It converges on a good and stable solution as illustrated in figure 6. The regularization enables a good convergence and the adaptive part allows it to identify and exit erroneous convergence processes. The reliability is a lot higher than RPF for the same number of particles.

Figure 7 shows some performance evaluation plots over a simulated flight. The top plot illustrates an initial convergence of the thermal center position estimate with the real one. This is followed by strong divergence after which the estimator converges stably to the correct solution. The thermal radius and mixing-layer altitude (\( Z_i \)) estimates (3\textsuperscript{rd} and 4\textsuperscript{th} plots) also show the same behavior. The convection velocity scale \( (w^*) \), last plot) shows a stable convergence throughout the simulation. The energy rate prediction (2\textsuperscript{nd} plot) is quite close to the measured one, providing a good input for a future trajectory controller.

Good estimation results were obtained with a 5Hz update rate, although lower rates may be used with low estimation degradation. Table II also shows that the RPF and RAPF results were obtained with a computation time quite suitable for real time estimation (\( \approx \frac{1}{3} \) of the update time, in MatLab\textsuperscript{TM}, on a Intel\textsuperscript{R} Core \textsuperscript{TM} 2 Duo CPU, T9300 @ 2.5GHz).
TABLE II
ESTIMATORS PERFORMANCE

<table>
<thead>
<tr>
<th>Filter</th>
<th>Localization error ($\mu_{err} \pm \sigma_{err}$)</th>
<th>Strength error ($\mu_{err} \pm \sigma_{err}$)</th>
<th>Update time ($\mu_T \pm \sigma_T$)</th>
<th>Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td>PF</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>Converges to a wrong solution</td>
</tr>
<tr>
<td>APF (600part)</td>
<td>127 ± 97 m</td>
<td>26 ± 11 W</td>
<td>0.044 ± 0.0028 s</td>
<td>Sometimes converges to a wrong solution</td>
</tr>
<tr>
<td>RPF (600part)</td>
<td>27 ± 17 m</td>
<td>13 ± 7.5 W</td>
<td>0.184 ± 0.0096 s</td>
<td>Converges to a good solution</td>
</tr>
<tr>
<td>RAPF (600part)</td>
<td>14.6 ± 9.7 m</td>
<td>9.7 ± 3.5 W</td>
<td>0.042 ± 0.0013 s</td>
<td>Converges to a good solution</td>
</tr>
</tbody>
</table>

As this was developed to be integrated in UAVs, the method will only be fully proved when applied to real flight data, and interfaced with a soaring control system. An online trust measure on the estimate has also to be created, for user information and automatic control mode switching. The ultimate extension would be the tracking of more than one thermal, allowing the UAV to optimize a soaring flight plan.

ACKNOWLEDGMENT

We would like to thank Ryan Eustice for his teachings on the Introduction to Probabilistic Mobile Robotics class, which led to the development of the current project. The author would like to further thank Prof. A. Girard, who read the manuscript carefully and pointed out areas that needed work. We gratefully acknowledge the support of the AsasF group and the researchers from the Underwater Systems and Technology Laboratory, specially João Sousa, Gil Gonçalves, and Eduardo Oliveira, and the Portuguese Air Force Academy. The research leading to this work was funded by Financiamento plurianual of the FEUP ISR Porto R&D unit and by the FCT (Foundation for Science and Technology) under PhD grant SFRH/BD/40764/2007.

REFERENCES