Feasibility of Varying Geo-Fence around an Unmanned Aircraft Operation based on Vehicle Performance and Wind

Sarah D'Souza
Systems Analysis Office
NASA Ames Research Center
Moffett Field, CA 94035

Abe Ishihara
Stinger Ghaffarian Technologies, Inc.
Moffett Field, CA 94035

Ben Nikaido
and Hashmatullah Hasseeb
Science & Technology Corp.
Moffett Field, CA 94035

***Abstract*—Managing trajectory separation of unmanned aircraft is critical to ensuring accessibility, efficiency, and safety in low altitude airspace. The concept of a geo-fence has emerged as a way to manage trajectory separation. A geo-fence consists of distance buffers that enclose individual trajectories to identify a 'keep-in' region and/or enclose areas that identify 'keep-out' regions. The 'keep-in' geo-fence size can be defined as a static number or calculated as a function of vehicle performance characteristics, state of the airspace, weather, and other unforeseen events such as emergency or disaster response. Given that the fleet of Unmanned Aircraft Systems (UAS) operating in low altitude airspace will be numerous and non-homogeneous, calculating a 'keep-in' geo-fence will need to balance operational safety and efficiency. A recently tested UAS Traffic Management (UTM) prototype used a geo-fence size of 30 meters, horizontally and vertically, for every operation submitted. The goal of this work is to determine the feasibility of a generalized, simple algorithm that calculates geo-fence sizes as a function of vehicle performance and potential wind disturbances. The resulting geo-fence size could be smaller or larger because the vehicle performance in the presence of wind is considered, thus leading to trajectory separation that is safe and efficient.**

In this paper, two simplified methods were developed to determine the feasibility of calculating a geo-fence as a function of vehicle parameters and wind information. The first method calculates the geo-fence using basic vehicle parameters and wind sensor data in a set of algebraic-geometric equations. The second method models a generic PID control system that uses a simplified set of equations of motion for the plant and uses gain scheduling to account for wind disturbances. It was found that the Algebraic-Geometric Geo-fence Algorithm provides geo-fence sizes of approximately 15 meters horizontally and 5 meters vertically, which is much smaller than the UTM static value of 30 meters. In the PID Controller Geo-fence Algorithm it was found that the geo-fence size is further reduced to less than 5 meters, horizontally and vertically. These results reveal that implementing geo-fence calculations provide UTM with the ability to schedule and separate operations based on geofences that are dynamic to vehicle capability and environment, which is more efficient than using a single static geo-fence.

I. Introduction

Unmanned Aircraft Systems (UAS) are becoming increasingly popular for commercial and governmental applications, e.g. fighting wildfires, agricultural monitoring, surveillance, and delivery [1]. NASA initiated UAS Traffic Management (UTM) research could develop airspace integration requirements for these UAS in low altitude airspace [2]. Managing trajectory separation of unmanned aircraft is critical to ensuring accessibility, efficiency, and safety in low altitude airspace. The concept of a geo-fence has emerged as a way to manage trajectory separation. A geo-fence consists of distance buffers that enclose individual trajectories to identify a 'keep-in' region and/or enclose areas that identify 'keep-out' regions. The 'keep-in' geo-fence size can be defined as a static number or calculated as a function of vehicle performance characteristics, state of the airspace, weather, and other unforeseen events such as emergency or disaster response. Given that the fleet of UAS operating in low altitude airspace will be numerous and non-homogeneous, calculating a 'keep-in' geo-fence will need to balance operational safety and efficiency. A recently tested UTM prototype used a geo-fence size of 30 meters, horizontally and vertically, for every operation submitted. The goal of this work is to determine the feasibility of a generalized, simple algorithm that calculates geo-fence sizes as a function of multirotor vehicle performance and potential wind disturbances so that the geo-fence changes with the operation. The resulting geo-fence size could be smaller or larger because the vehicle performance in the presence of wind is considered, thus leading to trajectory separation that is safe and efficient.

II. Background

In this paper, two simplified methods were developed to determine the feasibility of calculating a geo-fence as a function of vehicle parameters and wind information. The first method calculates the geo-fence using basic vehicle parameters and wind sensor data in a set of algebraic-geometric equations. This method was developed based on the prevalence of simplified equations and kinematic modeling found in traffic management and path planning literature. These methods are practical in operational software due to reliability of a solution. These models can be found in NASA's Air Traffic Management trajectory prediction algorithms, with various fidelity [3], [4], [5]. Path planners for autonomous UAS use a decomposition approach, where assumptions are made to derive simplified kinematic and algebraic equations [6]. This allows for numerous, fast-time solutions to be generated and evaluated. McGee and Hedrick [7], [8] demonstrated that a mapping of the kinematic equations can be used on a fixed wing vehicles to account for the effects of wind. In this paper, the first method uses straightforward geometric relationships and point mass kinematic equations to calculate geo-fence sizes for multiple different multirotor UAS, as a practical implementation for operational software.

The second method develops a generalized control system model for multi-rotor UAS that accounts for winds along the path. This method was selected to improve the granularity of simulating trajectory deviations due to wind, while guaranteeing a solution. In Pounds et al. [9] the flight dynamics of a quadrotor are modeled using point mass rigid body dynamics with a PID controller, that accounts for the effects of flapping, roll, and pitch damping, but no winds. In Hoffman et al. [10] a point mass model is implemented with a PID controller for path tracking and attitude control. The attitude control is augmented with angular acceleration feedback. In Raza and Etele [11] and Waslander and Wang [12] the flight dynamics of a quadrotor, in the presence of wind gusts, is modeled using a similar methodology to [10]. Additionally, a comprehensive survey of rotorcraft UAV control methods [13] identified PID controllers as one of the most successful and widely used controllers, where its capability can be expanded via gain scheduling. While many other control algorithms are described in the survey paper, this work focused initially on PIDs and low fidelity approaches before developing higher fidelity approaches. Thus, the second method developed in this paper models the trajectory path using a simplified rigid body dynamics model [14] and PID controllers. The inner-loop controller uses constant gains, while the outer loop controller uses gain scheduling to adapt to varying wind conditions. The gain schedule table is determined using the Artificial Bee Colony (ABC) genetic optimization method [15], [16]. This method was selected because it is a gradient-free search, immune to problems related to local optima, and the code was available in open source [17].

Section III of this paper discusses the three different wind models used in this work, Section IV details the Algebraic-Geometric Geo-fence Algorithm (AGGA), Section V discusses the PID Controller Geo-fence Algorithm (PCGA), and Section VI discusses the results in the context of geo-fence sizing.

III. Wind Modeling

Key to the successful operation of airspace management is an understanding of the nominal and potentially hazardous weather patterns that exist within the airspace. This is especially true at low altitude, where thermal effects and winds can be unpredictable and potentially catastrophic to vehicle operations. In this study, three different wind models were integrated in the geo-fence algorithms. The best available wind information at the time of this study was acquired from the NOAA High Resolution Rapid Refresh (HRRR) model, the California State University-Mobile Atmospheric Profiling System (CSU-MAPS), and OpenFOAM simulation. Note that HRRR is a forecasting product, CSU-MAPS is real-time sensor data of wind at a single location, and OpenFOAM is a CFD modeling that is time-invariant for a particular flow condition. It should be noted that, although the geo-fence algorithms (Method 1 and Method 2) utilize different wind models, the magnitude of the wind components are comparative in magnitude.

A. NOAA Wind Data Integration

The National Oceanic and Atmospheric Administration (NOAA) National Center for Environmental Prediction (NCEP) provides several forecast products available for download pertaining to different aspects of weather information. Each product differs based on the following: variables predicted, forecast horizon, spatial and temporal granularity, forecast grid, underlying prediction model, and model run frequency. HRRR offers the highest spatio-temporal resolution data compared to the other products. The HRRR model runs once every hour providing a 15 hour forecast with a temporal resolution of 15 minutes. The grid over the US has a spatial resolution of 3 km. In addition, the HRRR forecast provides a 0-hr ahead forecast based on inputs from various sensor measurements during every model run which is treated as the ground truth during model validation. The variables used include wind velocity components at 10m and 80m heights above ground (AGL).

Depending on the location of the UAV operation, wind information is extracted from HRRR and assumed to be constant over each 3km by 3km spatial grid elements. Note that interpolation is not done over the spatial dimensions; future research will include this. An illustration of the HRRR grid is shown in Fig. 1

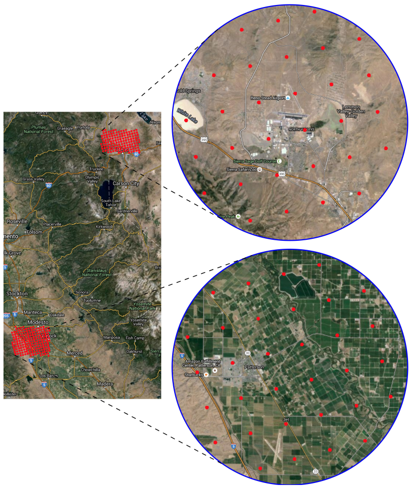

Fig. 1. Example grid plot

The truth data from June 29, 2016, at UTM test site Reno-Stead Airport (lat=39.6781,lon-119.876), is plotted in Fig. 2. While HRRR provides 15 minute resolution over a 15 hour forecast window, the truth data is extracted in hourly increments. Wind velocity components, $[u, v]$, are plotted at 10 and 80 meters AGL from 6AM to 4PM PDT. The vertical component of wind is not provided in the HRRR data set. The HRRR data is used in the AGGA.

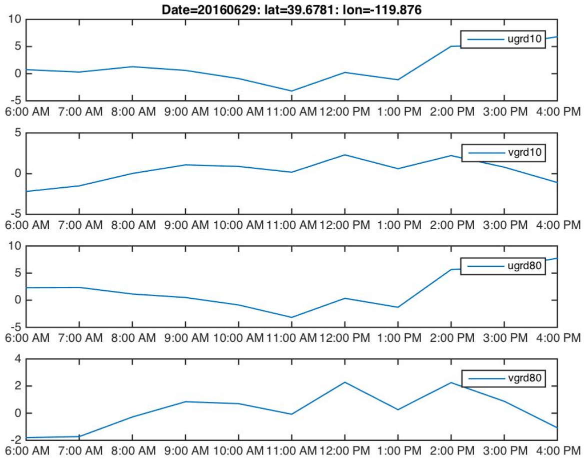

Fig. 2. Horizontal wind component profiles versus time from HRRR data at Reno-Stead on June 29th, 2016.

B. California State University-Mobile Atmospheric Profiling System Data

The CSU-MAPS is a joint facility developed by San Jose and San Francisco State Universities and funded by the National Science Foundation [18]. The CSU-MAPS is the most advanced mobile atmospheric boundary-layer profiling system in the United States. The system consists of two remote sensing tropospheric profilers: a scanning Doppler Lidar (Halo Photonics, Ltd, Streamline 75), and a microwave temperature and humidity profiler (Radiometrics, Inc., MP3000). Additionally, a mobile meteorological tower, mounted on a dual-axle trailer, collects wind measurements up to 32 m (106 ft) at 5 altitude levels.

The AGGA uses the CSU-MAPS three-dimensional (3D) wind data at altitudes of 12 and 32 meters. In Figure 3 the trend in the wind components over the course of the day is plotted.

C. OpenFOAM Wind Field

The wind models discussed above are primarily in non-urban areas, and it was important to develop an understanding of wind flows within urban areas, e.g. around buildings. Thus, a computation fluid dynamics (CFD) simulation was run on a model of a single simplified building, as shown in Figure 4.

This model was based on a wind tunnel experiment and included windows on the upstream and downstream faces that allowed cross flow through the building [19]. The open source CFD solver OpenFOAM version 2.2.2 [20] was utilized along with its Atmospheric Boundary Layer (ABL) inflow boundary condition to generate a wind field velocity solution of the single building model (Figure 5).

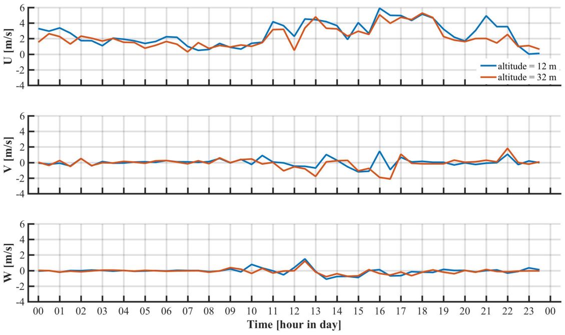

Fig. 3. Wind component profile versus time from CSU-MAPS sensor data on August 31st, 2015

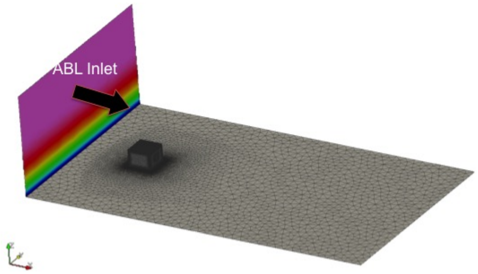

Fig. 4. Building geometry and volume.

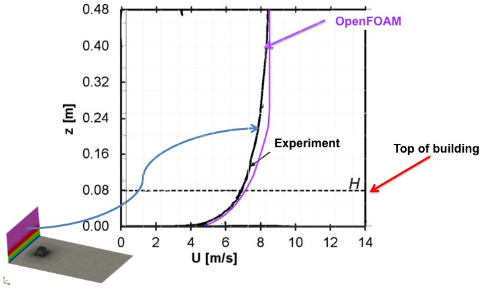

Fig. 5. Atmospheric altitude versus the inflow velocity at the ABL inlet.

The ABL boundary condition models the incoming velocity and turbulence profiles that result from the wind flowing over upstream distances and obstacles. Required boundary condition inputs include friction velocity, flow direction, ground normal direction, and surface roughness height [20]. Turbulence modeling in OpenFOAM's ABL boundary condition is based on the k-epsilon turbulence model [21]. OpenFOAM was chosen because it has functionality to model the Atmospheric Boundary Layer, which plays a large role in characterizing the wind field at low altitude.

OpenFOAM is a cell-centered unstructured CFD solver. The commercial grid generation software Pointwise was used to generate a mixed-element prismatic and tetrahedral volume mesh around the building geometry. Scalar solution values including x, y, and z components of velocity are saved to locations at the center of every cell of the volume mesh. OpenFOAM solves the 3D incompressible Navier-Stokes equations, which provides velocity vectors at each of the discretized points in the OpenFOAM volume solution, both in and around the building.

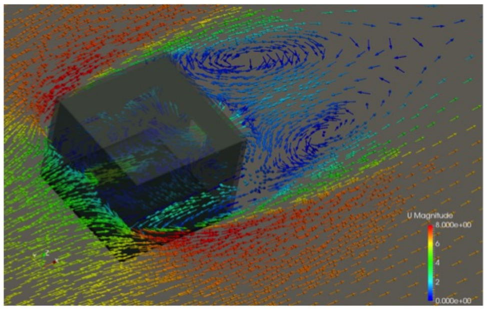

Fig. 6. Cross sectional view of the wind field solution from OpenFOAM

Figure 6 shows the wind vectors corresponding to a cross section of the flow volume at a specific altitude, where the spectrum of red to blue wind vectors indicate high speed and low speed respectively. The 3D wind field generated here iss used in the PCGA.

IV. Method 1: Algebraic-Geometric Geo-fence Algorithm

A. *Methodology*

The first method is a simplistic approach that guarantees a geo-fence solution can be found every time and for any multirotor vehicle operation. The vehicle parameters of interest for the AGGA are maximum vehicle airspeed (V_{max}) and the estimated time from detection of disturbance to vehicle recovery ($t_{control}$). Since this variable was not available from the UTM vehicle partners, $t_{control}$ was set to a constant value of 1 second based on the settling/response times for multirotors documented in references [11], [22], [23], [24], [25], [26], [27]. The maximum vehicle speed for NASA UTM vehicle partners operating multirotor vehicles were used in this study. All the values in Table I were reported to the NASA UTM project by the vehicle partners.

TABLE I
Vehicles and corresponding maximum velocity.

Vehicle	$V_{max} [m/s]$
UAV_1	8.9
UAV_2	20.0
UAV_3	18.0
UAV_4	24.6
UAV_5	17.9
UAV_6	18.0
UAV_7	10.3
UAV_8	15.6
UAV_9	13.8
UAV_{10}	17.9

The geo-fence size is defined by a horizontal (d_{horz}) and vertical (d_{vert}) distance (Fig. 7), where d_{vert} is the maximum change in vehicle altitude (+/-). In order to determine these distances, the vehicle is considered a point mass with a velocity and heading. The heading is determined by the directional path from one waypoint to another (Fig. 7). The waypoints are defined by longitude (θ), latitude (ϕ), and altitude (h). The heading (ψ) changes with each waypoint pair, e.g. 1 to 2, 2 to 3, etc.

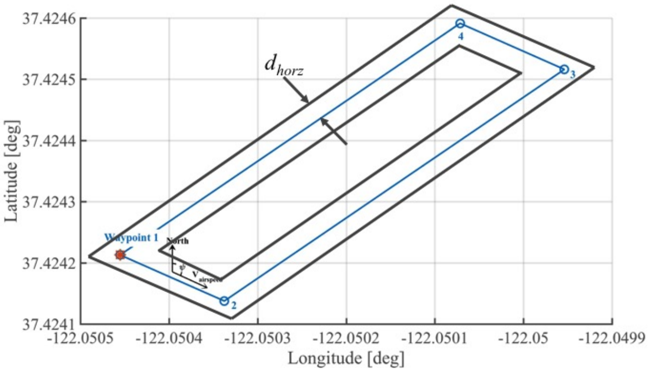

Fig. 7. Flight plan submitted by an operator requesting airspace with assigned geo-fence.

The effect of wind on the vehicle position is a function of the wind velocity (2) to the current vehicle velocity (1).

$$\begin{aligned} \vec{V}_{UAS} = V_{max} \cos\gamma \cos\psi \, \hat{i} + \\ V_{max} \cos\gamma \sin\psi \, \hat{j} + \\ V_{max} \sin\gamma \, \hat{k} \end{aligned} \tag{1}$$

$$\vec{V}_{wind} = U \, \hat{i} + V \, \hat{j} + W \, \hat{k} \tag{2}$$

$$\vec{V}_{total,AGGA} = \vec{V}_{UAS} + \vec{V}_{wind} \tag{3}$$

The UTM project has two test sites (at Crows Landing in Northern California and at Reno-Stead Airport in Nevada), for which HRRR wind information was extracted. The worst case (maximum) horizontal wind components were determined from the HRRR data. Specifically, the maximum winds at nine points on the HRRR grid, at 10 meters and 80 meters altitude, were used for each test site. The vertical wind component required special treatment, since HRRR data does not include this wind component. During the UTM flight tests in 2015, the CSU-MAPS facility was deployed at Crows Landing, where 3D wind data was collected at altitudes of 12 and 32 meters. This data was used to calculate the average ratio of the vertical wind component to the total wind magnitude (4). Using this factor, an analytical relationship was derived to determine the vertical wind component (5).

$$n_{wind} = \left[\frac{W}{\sqrt{U^2 + V^2 + W^2}} \right]_{CSU-MAPS} \tag{4}$$

$$W = n_{wind} \sqrt{\frac{[U^2 + V^2]_{HRRR}}{1 - n_{wind}^2}} \tag{5}$$

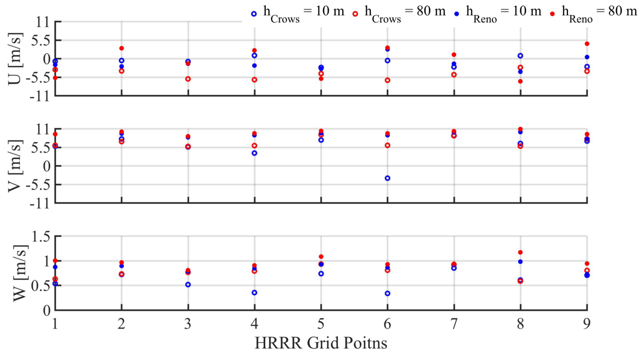

Fig. 8. Wind components derived from HRRR and CSU-MAPS data according to the maximum wind condition.

The final wind component values estimated, at each grid point and test site are plotted in Figure 8.

It is assumed that the vehicle's flight path angle (γ) is zero and that any changes in the vehicle's altitude is due to the vertical wind component. This leads to a geometric equation that determines an induced flight path angle 6 as a function of the vertical wind, W. This value is then input into (7).

$$\gamma_{wind} = \arctan \frac{W}{V_{max} \cos \psi + U} \tag{6}$$

The new position of the vehicle can now be found by integrating the kinematic equations of motions (7) - (9) over one time step that is equivalent to $t_{control}$.

$$\dot{h} = V_{total} \sin \gamma \tag{7}$$

$$\dot{\theta} = \frac{V_{total} \cos \gamma \sin \psi}{R \cos \phi} \tag{8}$$

$$\dot{\phi} = \frac{V_{total} \cos \gamma \cos \psi}{R} \tag{9}$$

The new position of the vehicle is as follows:

$$d_{vert} = t_{control} (V_{total} \sin \gamma_{wind}) \tag{10}$$

$$\theta_{new} = t_{control} \frac{V_{total} \sin \psi}{R \cos \phi_{old}} + \theta_{old} \tag{11}$$

$$\phi_{new} = t_{control} \frac{V_{total} \cos \psi}{R} + \phi_{old} \tag{12}$$

Where V_{total} is the magnitude of $\vec{V}_{total,AGGA}$ and R is the radius of the Earth. The new values of longitude and latitude represent the updated location of the vehicle due to displacement from the horizontal wind. Given the intended flight plan, the distance of this new location from a point perpendicular to the flight plan is calculated using the haversine equation (13). This distance is the horizontal geofence, d_{horz}. Note that the subscript 'fp' indicates the point along the flight plan that is perpendicular to the new vehicle location.

The geo-fence, $[d_{horz}, d_{vert}]$, is calculated for each vehicle/flight plan by averaging the intermediate geo-fence distances at each HRRR grid point.

B. Results

As expected, the results show that the geo-fence size is proportional to the velocity and wind magnitude for a constant $t_{control}$. Vehicles with higher speeds will have larger geo-fences. Table II and III show that the horizontal geo-fence sizes for each multirotor, are within 1 - 2 meters of each other for wind information at 10m and 80m altitude. However, the vertical geo-fence for UAV_4 (see Table III), corresponding to the Reno-Stead Airport wind information at 10m, is almost 4 meters larger than the other vertical geo-fences. This difference decreases to approximately 2 meters for wind information at 80m altitude. These results indicate some important conclusions regarding the feasibility of varying geo-fences. First, a simple algorithm that accounts for vehicle and wind information reduced the geo-fence size by approximately half the horizontal and vertical geo-fence size used by UTM of 30 m. Second, there is an operational advantage to simple methods that always guarantee a solution. Third, the size of the geo-fence will change with altitude due to changing terrains that generate highly variable winds. Thus, geo-fence algorithms should incorporate 3D wind fields that changes with time and altitude.

TABLE II
GEO-FENCE SIZES CORRESPONDING TO HRRR WIND DATA FOR CROW'S LANDING ON JUNE 29TH, 2016.

		Altitude = 10m		Altitude = 80m	
Vehicle	$V_{max}[m/s]$	$d_{horz}[m]$	$d_{vert}[m]$	$d_{horz}[m]$	$d_{vert}[m]$
UAV_1	8.9	11.5857	4.5194	11.9428	4.0280
UAV_2	20.0	12.8057	6.1315	13.1876	5.4933
UAV_3	18.0	12.5846	5.7713	12.9639	5.1279
UAV_4	24.6	13.3174	6.9510	13.7037	5.9792
UAV_5	17.9	12.5702	5.7469	12.9493	5.1136
UAV_6	18.0	12.5835	5.7694	12.9628	5.1268
UAV_7	10.3	11.7358	4.7936	12.0982	4.2364
UAV_8	15.6	12.3222	5.3061	12.6978	4.9363
UAV_9	13.8	12.1238	4.9384	12.4959	4.7110
UAV_{10}	17.9	12.5724	5.7507	12.9516	5.1158

TABLE III
GEO-FENCE SIZES CORRESPONDING TO HRRR WIND DATA FOR RENO-STEAD AIRPORT ON JUNE 29TH, 2016.

		Altitude = 10m		Altitude = 80m	
Vehicle	$V_{max}[m/s]$	$d_{horz}[m]$	$d_{vert}[m]$	$d_{horz}[m]$	$d_{vert}[m]$
UAV_1	8.9	11.9555	5.2231	12.0098	5.7588
UAV_2	20.0	13.1603	7.5511	13.1818	6.9278
UAV_3	18.0	12.9414	6.9430	12.9666	6.3297
UAV_4	24.6	13.6680	9.1401	13.6825	7.3194
UAV_5	17.9	12.9271	6.9076	12.9526	6.2925
UAV_6	18.0	12.9403	6.9402	12.9655	6.3268
UAV_7	10.3	12.1032	5.4294	12.1513	5.6056
UAV_8	15.6	12.6818	6.4051	12.7124	5.7873
UAV_9	13.8	12.4858	6.0592	12.5213	5.4380
UAV_{10}	17.9	12.9293	6.9130	12.9548	6.2982

V. METHOD 2: PID CONTROLLER GEO-FENCE ALGORITHM

A. Methodology

In contrast to the AGGA, the PCGA attempts to estimate vehicle deviations from the nominal trajectory due to wind, by

$$d_{horz} = 2R\arctan\frac{\sqrt{\sin^2\frac{\phi_{new}-\phi_{fp}}{2} + \cos\phi_{fp}\cos\phi_{new}\sin^2\frac{\theta_{new}-\theta_{fp}}{2}}}{\sqrt{1-\sin^2\frac{\phi_{new}-\phi_{fp}}{2} + \cos\phi_{fp}\cos\phi_{new}\sin^2\frac{\theta_{new}-\theta_{fp}}{2}}} \tag{13}$$

simulating the vehicle's control dynamics. The size of these deviation, horizontally and vertically, are then used to define the size of the geo-fence. Given the huge variety of UAVs, modeling each control system would be prohibitive. Therefore, a generalized and simple trajectory prediction model was developed such that the model, 1) does not rely on detailed knowledge of the control system and 2) predicts the UAV flight dynamics in the presence of urban winds. Requirements 1) and 2) are met by implementing a simplified set of equations of motion with a gain scheduled PID controller. The gain table was determined using a gradient-free genetic optimizer that is immune to problems related to local optima. This guaranteed a gain solution for every run and enabled the incorporation of wind field compensation in the control system. The following subsections describe the methodology.

1) Quadrotor Dynamics: Since the quadrotor is considered to be under-actuated, due to its four rotors controlling six degrees-of-freedom, some level of the dynamics in the plant must be modeled in the control design. The plant model assumes that the quadrotor is a point mass and that the equations of motion are derived based on rigid body dynamics and kinematics. A summary of the derivation for the equations of motion (EOMs) that describe the dynamics of a symmetric quadrotor follows below. For a detailed write-up of this derivation, see [14].

The two major reference frames are the vehicle-fixed body frame (B) and the Earth-fixed North-East-Up (NEU) inertial frame. The body-frame is attached to the center of gravity of the quadrotor. The NEU frame is attached to the location of the ground station tracking the vehicle. Upon determination of the transformation matrix from the B axis frame to the NEU axis frame, via the Euler angles, the body velocities and angular rates are transformed into the NEU axis frame. Then, Newton's Laws for translational and rotational motion are used to derive the six degree of freedom EOMs, see [14] for full set of equations. For implementation into the control system design, it is assumed that the quadrotor exhibits spherical aerodynamic characteristics (i.e., constant drag coefficient in all directions) and that body lift and other nonlinear aerodynamic effects are negligible. The sum of the forces (14) on the vehicle include thrust, drag (16), and gravity. The torques on the vehicle are a function of the force on each motor and the distance of each motor from the center of mass, (17) - (19).

$$\begin{aligned}\vec{F} = (-mg\sin\theta_e + D_x)\hat{i}_B &+ \\ (mg\cos\theta_e\sin\phi_e + D_y)\hat{j}_B &+ \\ (-mg\cos\theta_e\cos\phi_e + F_t + D_z)\hat{k}_B&\end{aligned} \tag{14}$$

$$\vec{V}_{rel,sqr} = V_{rel,x}^2\hat{i}_B + V_{rel,y}^2\hat{j}_B + V_{rel,z}^2\hat{k}_B \tag{15}$$

$$\vec{D} = -\frac{1}{2}C_D A_{ref}\rho\vec{V}_{rel,sqr} \tag{16}$$

$$\tau_{\phi_e} = l(F_L - F_R) \tag{17}$$

$$\tau_{\theta_e} = l(F_F - F_B) \tag{18}$$

$$\tau_{\psi_e} = F_t K = K(-F_F + F_R - F_B + F_L) \tag{19}$$

Where $\vec{D}$ is the drag acceleration vector, C_D is the drag coefficient, A_{ref} is the reference area, ρ is the density of air at sea level, and K is a constant parameter for torque. The small angle approximation is applied to linearize the EOMs while the yaw angle is assumed to be zero. These assumptions yield a simplified set of governing equations (20) - (25).

$$\ddot{x} = -\cos\phi_e\sin\theta_e\frac{F_t}{m} + \frac{D_x}{m} \tag{20}$$

$$\ddot{y} = \sin\phi_e\frac{F_t}{m} + \frac{D_y}{m} \tag{21}$$

$$\ddot{z} = -g + \cos\phi_e\cos\theta_e\frac{F_t}{m} + \frac{D_z}{m} \tag{22}$$

$$\ddot{\phi}_e = \frac{(F_L - F_R)l}{J_x} \tag{23}$$

$$\ddot{\theta}_e = \frac{(F_F - F_B)l}{J_y} \tag{24}$$

$$\ddot{\psi}_e = \frac{(-F_F + F_R - F_B + F_L)K}{J_z} \tag{25}$$

where $[x, y, z]$ are position, ϕ_e is the roll angle, θ_e is the pitch angle, ψ_e is the yaw angle, F_t is total thrust from the rotors, $F_{L,R,F,B}$ are the resultant forces from each rotor (left, right, front, and back), $D_{x,y,z}$ are the drag forces in the direction of the relative velocity, g is gravity, m is mass of the vehicle, l is the distance from the rotor to the vehicle center of gravity, and $J_{x,y,z}$ are the moments of inertia. Two dots above a letter (diaeresis) are used to indicate accelerations.

The set of rotational equations, (23) - (25), are uncoupled and depends only on the forces generated by the vehicle, whereas the translational set, (20) - (22), are coupled to the rotational equations. The effect of the wind field on the trajectory is determined via the drag terms in the translational EOMs. The wind velocity vector (26) is added to the vehicle velocity vector and substituted into the drag vector equation. The wind velocity components are extracted from the OpenFOAM CFD solution at each point along the trajectory.

$$\vec{U}_{wind} = u_{wind}\hat{i}_{NEU} + v_{wind}\hat{j}_{NEU} + w_{wind}\hat{k}_{NEU} \tag{26}$$

Note that the vehicle velocity is in the body axis. Thus, the wind vector must be transformed from the NEU axis to the body axis (27), using the transformation matrix R_B^{NEU}. The velocity vector (27) is substituted into the drag equation.

$$\vec{V}_{rel} = \vec{V}_{vehicle} + R_B^{NEU}\vec{U}_{wind} \tag{27}$$

2) Control System Model: The control variables are total force, pitch angle, and roll angle. The intermediate control laws for commanded total force, roll angle, and pitch angle are derived by first treating the left-hand sides of (20) - (22) as commanded accelerations and then solving for the commanded total force (thrust) using (22). The newly derived force equation is substituted into (20) and (21) to determine the roll and pitch angle commands, (29) and (30) respectively.

$$F_t^{cmd} = \frac{m\left(a_z^{cmd} + g - \frac{D_z}{m}\right)}{\cos\phi^{cmd}\cos\theta^{cmd}} \tag{28}$$

$$\phi^{cmd} = \arctan\frac{\left(a_y^{cmd} - \frac{D_y}{m}\right)\cos\theta^{cmd}}{a_z^{cmd} + g - \frac{D_z}{m}} \tag{29}$$

$$\theta^{cmd} = \arctan\frac{-\left(a_x^{cmd} - \frac{D_x}{m}\right)}{a_z^{cmd} + g - \frac{D_z}{m}} \tag{30}$$

The variables $[a_x, a_y, a_z]^{cmd}$ are acceleration commands in the x, y, and z body axis. The final control system diagram is shown in Figure 9.

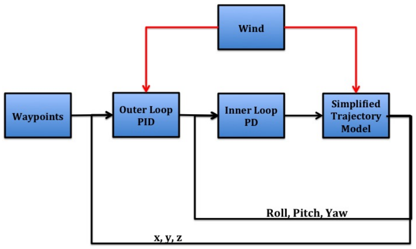

Fig. 9. Multirotor control system diagram

For this model, an outer-loop PID controller is used to reduce steady-state error (e_r) in reaching a waypoint, and an inner-loop PD controller is used to reflect the fast attitude dynamics of multi-rotor vehicles (31).

$$PID = K_p e_r + K_d\frac{de_r}{dt} + K_i\int_0^t e_r d\tau \tag{31}$$

The inputs to the control system are the desired waypoints that define the operation, $[x, y, z]$, and the outputs are controls that drive the equations of motion to generate the desired trajectory. The inner-loop of the control system in Figure 9 controls attitude; while the outer-loop controls vehicle position. The outer-loop system will be used to determine commanded accelerations that will in turn determine total thrust and commanded attitude angles using (28) - (30). The inner-loop system will take this information and output torques to achieve an intended orientation. The integrative term, in (31), will become more influential as error accumulates, driving the vehicle closer to its intended destination. Additionally, this term adds to the vehicle's robustness to wind disturbances provided there is no degradation in stability.

The control system uses constant gains for the inner-loop controllers and gain scheduling in the outer loop to adapt to varying wind conditions. The gain schedule table is determined using the Artificial Bee Colony (ABC) genetic optimization method [15], [16]. A diagram from [15] is modified to illustrate this paper's implementation of the ABC method in Figure 10.

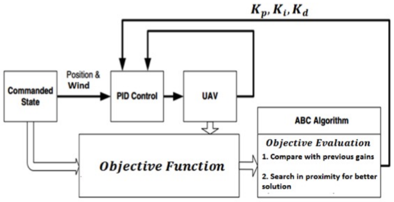

Fig. 10. Gain scheduling control diagram with ABC optimization

The design variables in the optimization are the PID gains for each controller for a given wind magnitude (not for each wind component). The minimum and maximum wind magnitudes were extracted from the OpenFOAM wind field model and gains were determined at each wind condition, in increments of $\Delta_{INCR} = \sqrt{3}$.

$$\left[\left|\vec{U}_{wind}\right|_{min} : \Delta_{INCR} : \left|\vec{U}_{wind}\right|_{max}\right] = \left[-5\sqrt{3} : \sqrt{3} : 5\sqrt{3}\right] \tag{32}$$

The cost function used to find the gains at each controller is a function of six parameters and weighting coefficients ($R_{j=1:6}$). These parameters are the position error at the final waypoint (e_{pos}), Integrated Time Absolute Error ($ITAE$), Percent Overshoot ($\%OS$), Rise Time (T_{rise}), Settling Time ($T_{settling}$) and the Number of oscillations in the trajectory (N_{ring}). The objective function, 33, is the sum of these costs.

$$\begin{aligned} J_{i=x,y,z} = (R_1(e_{pos})^2 + R_2(ITAE)^2 + \\ R_3(\%OS)^2 + R_4 T_{rise} + \\ R_5 T_{settling} + R_6 N_{ring})_{i=x,y,z} \\ J = J_x + J_y + J_z + RT_{final} \end{aligned} \tag{33}$$

3) Vehicle and Mission Parameters: The test vehicle for this simulation is the AscTec Pelican Quadrotor sUAS (AscTec) [28]. This vehicle is 0.651m long with a rotor diameter of approximately 25.4cm and length of 32.5cm. It has a maximum take-off weight (MTOW) of 1.65kg and a take off thrust of 36 Newtons. The specifications indicate that this vehicle can tolerate a wind speed up to 10m/s, but do not indicate whether this is a sustained wind or gust value. Based on the assumption of a spherical aerodynamic model, C_D is set to a constant 0.5, with a constant A_{ref} of 0.33 m^2. Additional vehicle parameters can be found in Table IV.

TABLE IV
AscTec Pelican vehicle parameters.

Parameter	Value
$MTOW$	1.65 kg
$F_{thrust,max}$	36 N
l	0.32550 m
J_x	0.012434 kg/m^2
J_y	0.012434 kg/m^2
J_z	0.021969 kg/m^2
C_D	0.5
A_{ref}	0.33285 m^2
$V_{wind,max}$	10 m/s

The integrated vehicle and wind field model was tested via two operational test profiles. The first test profile examined the vehicle's resilience to varying wind speeds when flying through the wake behind the building, Figure 11. The second test profile examined the vehicle's resilience to sustained, high wind speeds when flying near the side of the building, Figure 12.

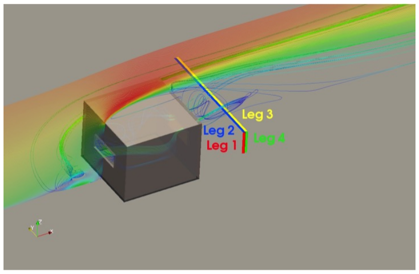

Fig. 11. Flight plan through the wake behind a building

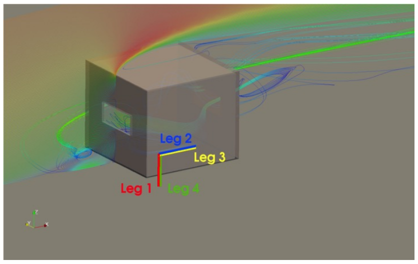

Fig. 12. Flight plan along corner and side of building

The test profile consists of four segments: 1) Ascent, 2) Uni-directional Forward Flight, 3) Uni-directional Backward Flight, and 4) Descent.

B. Results

The first test profile shows that the system is able to maintain an altitude of 3 meters and manages to arrive at the final destination at almost the same time as the no-wind trajectory (Figure 13). The control system also does exceptionally well at tracking the desired position with very little deviation from the no-wind path (Figure 14). Thus, even when the wind field is constantly varying due to the wake vortices behind the building (Figure 15), the flight path is robust to the wind variation. This result shows that the horizontal and vertical geo-fence can be very small ($< 5m$).

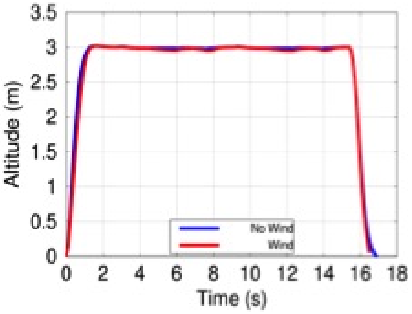

Fig. 13. Altitude versus time for test profile 1.

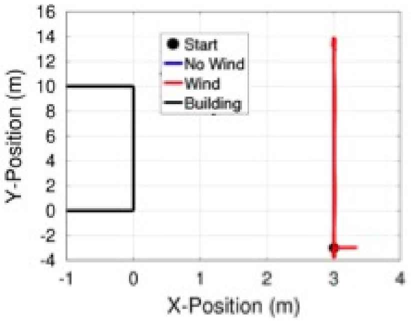

Fig. 14. Ground track for test profile 1.

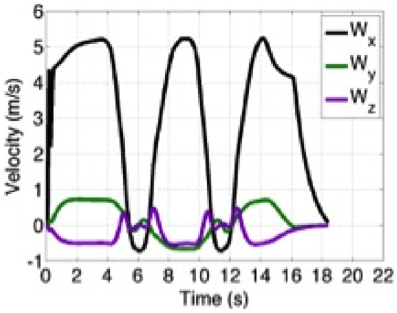

Fig. 15. wind components versus time for test profile 1.

The second test profile shows that the system is able to maintain an altitude of 3 meters, but arrives at its destination almost 120 seconds later than the no-wind trajectory (Figure 16). The control system, from a spatial perspective, does track to the horizontal position with very little deviation from the no-wind path (Figure 17), resulting in a small horizontal geo-fence. It should be noted that the constant force of the wind (Figure 18) on the vehicle exposed a region of poor performance in the PID, where the vehicle flight time was significantly larger than the nominal.

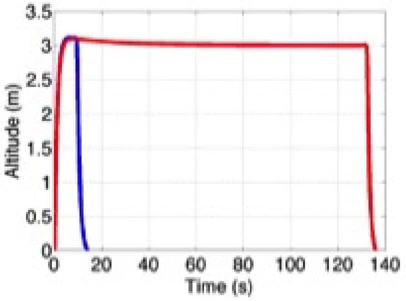

Fig. 16. Altitude versus time for test profile 2.

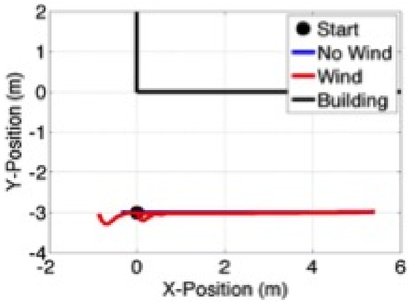

Fig. 17. Ground track for test profile 2.

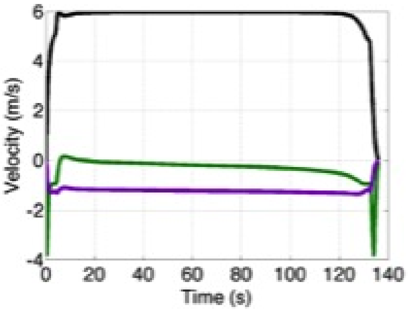

Fig. 18. wind components versus time for test profile 2.

At this point it is important to understand the reasons for the poor performance. This will allow for further identification of control system requirements that ensure robust performance for accurate geo-fencing and efficient operation. The poor performance of the controller may be due to the small angle approximation assumption applied to simplify the model. To test this, the small angle approximation was relaxed and the 2nd test profile was simulated again.

The resulting altitude profile (Figure 19) shows that the vehicle arrived at its final destination approximately two seconds later than in the no-wind condition and that the ground track deviation (Figure 20) was very small, leading to a geo-fence assignment that is very small across the operation.

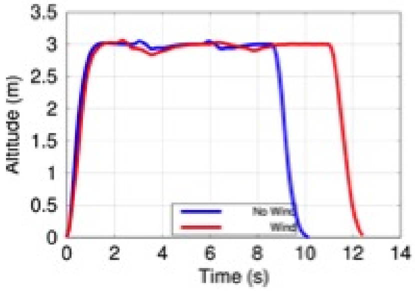

Fig. 19. Altitude versus time for test profile 2 with relaxed small angle approximation.

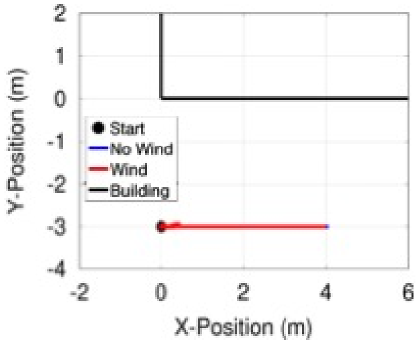

Fig. 20. Ground track for test profile 2 with relaxed small angle approximation.

Some important conclusions can be drawn from these results. The first is that the performance of even the most basic control system reveals geo-fence distances, in the horizontal and vertical, that are smaller than that of the AGGA or those values used in the UTM prototype. Second, the fidelity of the flight dynamics model needs to be increase to compensate for a high sustained wind to provide improved geo-fence sizes.

VI. Conclusions and Future Work

It was found that the AGGA calculates geo-fence sizes of approximately 15 meters horizontally and 5 meters vertically, which are smaller than the UTM static value of 30 meters. In the PCGA it was found that the geo-fence size is further reduced to less than 5 meters, horizontally and vertically. The PCGA results revealed that an increased level of fidelity must be added to accurately simulate vehicle deviations from wind disturbances. Overall, the results reveal that implementing geo-fence calculations will provide UTM with the ability to schedule and separate operations based on vehicle capability and environment, which is more efficient than using a single static geo-fence.

Future work should include development of a higher fidelity generalized model on four fronts: aerodynamics, flight mechanics modeling, control system, and urban wind field modeling. First, the aerodynamics of UAS must be modeled more accurately such that the effects of the wind disturbance correctly reflect the aerodynamics in each axis. UAS also experiences aerodynamic moments and lift depending on the vehicle's configuration, e.g. '+' or 'x' configuration. These forces impact how the control system maintains the vehicle's trajectory and stability. Second, UAS will not be limited to the linear region of operation as higher wind speeds and turbulent flows will require significant fast-time changes in pitch, roll, and yaw angles for maintaining flight trajectory. Therefore, the model must account for gyroscopic and other nonlinear effects that significantly influence the total dynamics of the system. Third, the task of dealing with the wide range of nonlinear aerodynamics and dynamic effects may require a model of a nonlinear control system. This implementation will require a trade-off in fidelity since this model will need to be broadly applicable to different types of UAS, e.g. fixed wing, multirotors, etc. Finally, fidelity of the urban wind environment should be increased by including multi-building configurations and varying the inflow condition. The inflow condition is the initial condition velocity profile that corresponds to the prevailing wind speed and terrain.

Acknowledgment

The authors would like to thank NASA Ames colleagues John Melton (Systems Analysis Office), Jaweoo Jung (UTM Vehicles and Surveillance lead), and William Chan (Modeling and Simulations Branch Chief) for their feedback and support of this work.

References

[1] *FAA Aerospace Forecast: Fiscal Years 2016-2036*, Federal Aviation Administration, 2016.

[2] P. Kopardekar, J. Rios, T. Prevot, M. Johnson, J. Jung, and J. Robinson, "Unmanned aircraft system traffic management (utm) concept of operations," in *16th AIAA Aviation Technology, Integration, and Operations Conference*, Washington, DC, Jun. 2016.

[3] K. Bilimoria and B. Sridhar, "Facet: Future atm concepts evaluation tool," in *3rd USA/Europe Air Traffic Management R&D Seminar*, Napoli, Italy, Jun. 2000.

[4] R. Slattery and Y. Zhao, "Trajectory synthesis for air traffic automation," *Journal of Guidance, Control, and Dynamics*, vol. 20, no. 2, pp. 232–238, Apr. 1997.

[5] D. Poles, "Base of aircraft data (bada) aircraft performance modeling report," EUROCONTROL, France, Tech. Rep. EEC Technical/Scientific Report No. 2009-009, Mar. 2009.

[6] C. Goerzen, Z. Kong, and B. Mettler, "A survey of motion planning algorithms from the perspective of autonomous uav guidance," *Journal of Intelligent and Robotic Systems*, vol. 57, pp. 65–100, Jan. 2010.

[7] T. McGee and J. K. Hedrick, "Path planning and control for multiple point surveillance by an unmanned aircraft in wind," in *Proceedings of the 2006 American Control Conference*, Minneapolis, MN, Jun. 2006.

[8] T. G. McGee and J. K. Hedrick, "Optimal path planning with a kinematic airplane model," *Journal of Guidance, Control, and Dynamics*, vol. 30, no. 2, pp. 629–633, Apr. 2007.

[9] P. Pounds, R. Mahony, and P. Corke, "Modelling and control of a large quadrotor robot," *Control Engineering Practice*, vol. 18, pp. 691–699, Feb. 2010.

[10] G. M. Hoffman, S. L. Waslander, and C. Tomlin, "Quadrotor helicopter trajectory tracking control," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, paper AIAA 2008-7410.

[11] S. A. Raza and J. Etele, "Simulation tool for testing and validating uav autopilots in wind gust environments," in *AIAA Atmospheric Flight Mechanics Conference*, paper AIAA 2012-4418.

[12] S. L. Waslander and C. Wang, "Wind disturbance estimation and rejection for quadrotor position control," in *AIAA Infotech@Aerospace Conference*, paper AIAA 2009-1983 address =.

[13] F. Kendoul, "Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems," *Journal of Field Robotics*, vol. 29, no. 2, pp. 315–378, Jan. 2012.

[14] R. Beard, "Quadrotor dynamics and control rev 0.1," Brigham Young University All Faculty Publications, Tech. Rep. Paper 1325, 2008.

[15] M. Abachizadeh, M. R. H. Yazdi, and A. Yousefi-Koma, "Optimal tuning of pid controllers using artificial bee colony algorithm," in *2010 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Montreal, Canada, 2010.

[16] P. Ghiglino, J. L. Forshaw, and V. J. Lappas, "Online evolutionary swarms algorithm for self-tuning unmanned flight control laws," *Journal of Guidance, Control, and Dynamics*, vol. 38, no. 4, pp. 772–782, Apr. 2015.

[17] D. Karaboga. (2009) Artifical bee colony algorithm. [Online]. Available: http://mf.erciyes.edu.tr/abc/index.htm

[18] C. B. Clements and A. J. Oliphant, "The california state university mobile atmospheric profiling system: A facility for research and education in boundary layer meteorology," American Meteorological Society, Nov. 2014.

[19] R. Ramponi and B. Blocken, "Cfd simulation of cross-ventilation flow for different isolated building configurations: validation with wind tunnel measurements and analysis of physical and numerical diffusion effects," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 104-106, pp. 408–418, May 2012.

[20] (2015) atmboundarylayer boundary condition, in openfoam programmer?s c++ documentation. [Online]. Available: http://foam.sourceforge.net/docs/cpp/a00072.html

[21] D. M. Hargreaves and N. G. Wright, "On the use of the k-epsilon model in commercial cfd software to model the neutral atmospheric boundary layer," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 95, pp. 355–369, 2007.

[22] A. Nemati, M. Sarim, M. Hashemi, E. Schnipke, S. Reidling, W. Jeffers, J. Meiring, P. Sampathkumar, and M. Kumar, "Autonomous navigation of uav through gps-denied indoor environment with obstacles," in *AIAA SciTech: Infotech @ Aerospace*, paper AIAA 2015-0715.

[23] R. A. Wise and R. T. Rysdyk, "Recovery of uav coordination robustness lost by slow information update rates," in *AIAA Guidance, Navigation, and Control Conference and Exhibit*, paper AIAA 2007-6412.

[24] J. Wang, T. Bierling, M. Achtelik, L. Hocht, F. Holzapfel, W. Zhao, and T. H. Go, "Attitude free position control of a quadrotor using dynamic inversion," in *AIAA Infotech @ Aerospace 2011*, paper AIAA 2011-1583.

[25] C. T. Ton, M. McCourt, and S. Mehta, "Robust tracking control of a quadrotor with time-varying gain in the presence of uncertainty and disturbances," in *AIAA SciTech: Guidance, Navigation, and Control Conference*, paper AIAA 2016-1389.

[26] B. Michini and J. P. How, "L1 adaptive control for indoor autonomous vehicles: Design process and flight testing," in *AIAA Guidance, Navigation, and Control Conference*, paper AIAA 2009-5754.

[27] M. Harada, R. Ichikawa, S. Watanabe, and K. Bollino, "L1 adaptive control for single coaxial rotor mav," in *AIAA SciTech: Guidance, Navigation, and Control Conference*, paper AIAA 2016-0894.

[28] Asctec pelican safety data sheet. [Online]. Available: http://www.asctec.de/downloads/datasheets/