QUT Digital Repository:
http://eprints.qut.edu.au/

Eng, Pillar C. and Mejias, Luis and Walker, Rodney A. and Fitzgerald, Daniel L. (2010) ***Guided chaos : path planning and control for a UAV-forced landing***. IEEE Robotics and Automation Magazine, 17(2). pp. 90-98.

Copyright 2010 IEEE

Guided Chaos

Path Planning and Control for a UAV-Forced Landing

© DIGITAL VISION

BY PILLAR ENG, LUIS MEJIAS, RODNEY WALKER, AND DANIEL FITZGERALD

In recent years, unmanned aerial vehicles (UAVs) have been widely used in combat, and their potential applications in civil and commercial roles are also receiving considerable attention by industry and the research community. There are numerous published reports of UAVs used in Earth science missions [1], fire-fighting [2], and border security [3] trials, with other speculative deployments, including applications in agriculture, communications, and traffic monitoring. However, none of these UAVs can demonstrate an equivalent level of safety to manned aircraft, particularly in the case of an engine failure, which would require an emergency or forced landing. This may be arguably the main factor that has prevented these UAV trials from becoming full-scale commercial operations, as well as restricted operations of civilian UAVs to only within segregated airspace.

To date, parachutes or parafoils are most commonly used to allay the severity of a UAV-forced landing, while still providing some degree of controllability for the aircraft [4]. The problem with this approach is that it is highly susceptible to changing winds that may adversely affect the final impact point. Having a parachute or parafoil onboard also increases the aircraft weight and complexity. Another solution that has seen recent operation at the 2008 UAV Outback Challenge held in Australia requires the UAV to deflect its control surfaces such that the aircraft will spiral into the ground,

Digital Object Identifier 10.1109/MRA.2010.936949

1070-9932/10/$26.00©2010 IEEE

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

following an engine or communications failure [5]. Although this solution has been proven to successfully terminate the flight and also contain the crash site within a limited area, it is hardly desirable for operations over populated areas. Other safety systems currently available allow the UAV to fly toward a predefined safe ditching area selected from a database of such known locations [6]. To date, the only reported successful UAV forced landing involves the U.S. Air Force Global Hawk, which performed a gliding descent under remotely-piloted control to an emergency airstrip in 2006 [7].

Procedures for guiding manned aircraft during an emergency landing exist and are discussed in [8]. As the ability to execute a successful forced landing remains the primary indicator for safety in the manned aviation industry, automating this capability for UAVs will help to overcome a key impediment to their routine operations over civilian airspace. In developing such a system [9]–[11], we have divided the problem into the following three research areas:

1) automated visual identification and classification of UAV-forced landing sites
2) automated path planning and control for descent and landing
3) automated multicriteria decision making for high-level reasoning during the descent.

This article will focus on the work to date in the second category, which has mainly involved simulating several different path planning and control strategies using the MATLAB computing program. We have used a six-degrees of freedom flight dynamics model (FDM) of a Boomerang radio-controlled aircraft, which represents the UAV platform to be used in future flight tests. The FDM was constructed using the classic coefficient buildup method [12] as part of an undergraduate project at the Queensland University of Technology (QUT), and wind gusts and lulls were simulated using the von Karman turbulence model. Specifically, the challenge we face is that of how to guide an unpowered, fixed-wing aircraft to arrive at a specific point in space for landing (approach point), and at a certain airspeed and heading while accounting for any kinodynamic constraints, regardless of the ambient wind conditions. The desired airspeed and heading at the approach point are supplied by the higher level decision maker, based on information obtained by the site-selection algorithms. In this research, we have restricted our discussion to the case of engine failure only and assumed that the avionics and flight control surfaces are still intact; the aircraft is aligned with the longest portion of the selected landing site and facing into the wind at the approach point; there are no obstacles in the flight path; and wind velocities can be estimated by onboard instruments (albeit with certain errors).

Our initial approaches for path planning and control consisted of two sets of differing algorithms tested via Monte-Carlo simulations [9], [10], with an average miss distance of 200 m at the approach point. The key contributions of this article exhibit much better performance when compared with our previous algorithms and include the development of a three-dimensional (3-D) path planning algorithm for a UAV-forced landing, based on the work of Dubins [13] and Ambrosino [14], as well as the enhancement of the path following algorithm developed by Park et al. [15] to suit an unpowered, gliding descent. Simulation results have demonstrated that the path planning and control approaches are matured enough to be implemented in a prototype system for flight testing.

Path Planning

Numerous robotics path planning techniques are presented in the literature, and a comprehensive summary of existing methods can be found in [16]. However, we have found that the motion of a gliding aircraft can best be described by trajectories derived from Dubins' curves [13]. In [17], a variation of Dubins' algorithm was used to construct the shortest-time path for a UAV through known two-dimensional (2-D) waypoints that varied their position to account for wind disturbances. In [14], Dubins' algorithms were used to construct an optimal 2-D path. This was then used to initialize a 3-D trajectory that accounted for path angle constraints. Finally, in [18], a time-optimal 3-D trajectory is formed for aircraft path planning and control problems. The authors show that this 3-D path can be constructed of a series of arcs corresponding to the aircraft minimum turn radius, straight line segments, and pieces of planar elastic selected from a library, and that time-optimality can be obtained by using the Pontryagin Maximum Principle [19]. For the UAV-forced landing problem, time-optimality is not as critical as enabling the aircraft to complete the descent and land safely. Hence, we have based our path planning approach on the work presented in [14], albeit with certain modifications as described next.

Initially, a 2-D Dubins path is constructed having the form:

$$\{L_\alpha R_\beta L_\gamma, R_\alpha L_\beta R_\gamma, L_\alpha S_d L_\gamma, L_\alpha S_d R_\gamma, R_\alpha S_d L_\gamma, R_\alpha S_d R_\gamma\}, \quad (1)$$

in which L and R correspond to left and right turns at a bank angle that does not exceed the maximum bank angle of the aircraft, S corresponds to flying in a straight line, and $\alpha,\gamma \in [0,2\pi)$, $\beta \in (\pi,2\pi)$, and $d \geq 0$. The radii of the arcs were calculated using the equation:

$$R_{0,\mathrm{f}} = \frac{V_{\mathrm{TAS}}^2}{g \tan(\phi_{0,\mathrm{f}})}, \quad (2)$$

where $R_{0,\mathrm{f}}$ are the initial and final radii of the arcs of circumference, V_{TAS} is the true airspeed of the aircraft, g is the gravitational acceleration constant (9.80665 m/s^2), and $\phi_{0,\mathrm{f}}$ are the initial and final bank angles, respectively, which can be different.

To simplify the path planning process, we have taken V_{TAS} to be the best glide speed V_{bg} of the aircraft, which gives the greatest straight-line flight distance in still air from the potential energy of height. As shown in Figure 1, we can estimate V_{bg} by first fitting a curve (black) through the descent rates at various airspeeds (red diamonds). The best glide speed is then obtained by drawing the blue line from the origin tangent to the curve, giving V_{bg} as 18.63 m/s and a nominal lift-to-drag ratio (L/D) of 24.5. Although this value may seem too optimistic, remember that it was obtained using simulated sensor data under ideal conditions, with no wind and no sensor errors.

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

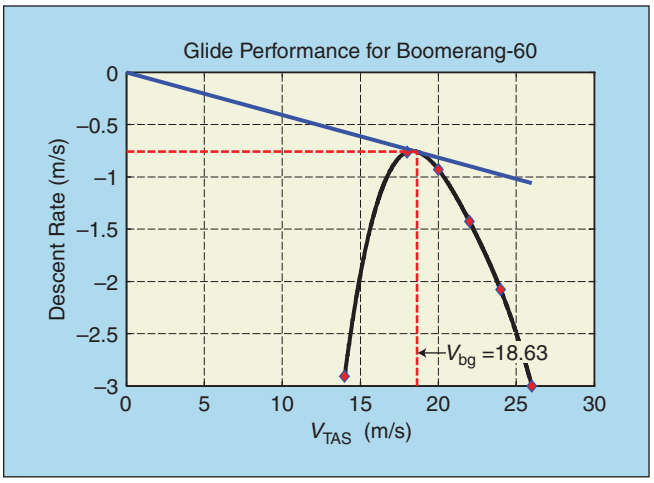

Figure 1. *Speed polar diagram for a Boomerang 60 size UAV, showing how the best glide speed (V_{bg}) is obtained.*

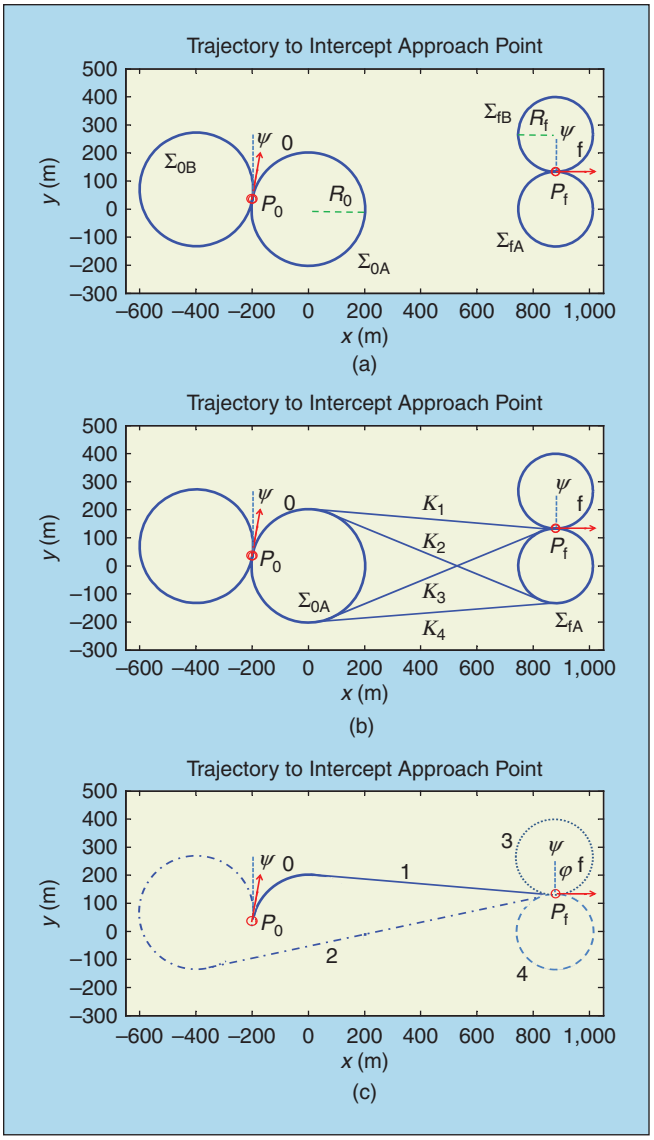

Figure 2. *(a) Step 1 of generating the 2-D path. (b) Step 2 of generating the 2-D path. (c) Step 3 of generating the 2-D path. Four plausible paths are obtained; the optimal path is path number 1.*

Once the radii are determined, the optimal 2-D path is obtained with a geometrical construction adapted from [14]. First, two circles with radii R_0 are drawn containing the starting point P_0 and a vector pointing along the aircraft's initial heading ψ_0 [Figure 2(a)]. The circumferences of the circles are denoted by $\sum_{0A}$ and $\sum_{0B}$. Next, the same process is repeated at the goal point P_f with the final aircraft heading ψ_f, and circumferences $\sum_{fA}$ and $\sum_{fB}$. Following this, tangent lines are constructed that join the circumferences of these circles, such as depicted in Figure 2(b) for $\sum_{0A}$ and $\sum_{fA}$. Considering Figure 2(b), we readily observe that there are four paths connecting P_0 to P_f, where a path is formed by the union of an arc on the circumference $\sum_{0A}$, a segment K on one of the four tangent lines, and finally an arc on the circumference $\sum_{fA}$. However, only one of these paths, Γ_{AA} is compatible with the initial and final headings of the UAV [Figure 2(b) and (c)]. In a similar way, three other paths Γ_{AB}, Γ_{BA} and Γ_{BB} can be obtained—the optimal path is the shortest between Γ_{AA}, Γ_{AB}, Γ_{BA}, and Γ_{BB} and depicted as a thick, solid line in Figure 2(c).

Following the construction of Γ_{xy}, and given the distance d_{tgt} from the initial point of failure to the approach point, we can then obtain the path angle:

$$\gamma_{xy} = \tan^{-1}\left(\frac{z_{arc_f} - z_{arc_o}}{d_{tgt}}\right), \tag{3}$$

which allows the UAV to descend from an altitude of z_{arc_0} to z_{arc_f}. To ensure stability: $\gamma_{min} \leq \gamma \leq \gamma_{max}$. However, if the difference in altitude between the start and end positions should result in the maximum allowable path angle being exceeded, one of the other suboptimal paths can be selected to lose the approximate amount of altitude required. Other options include enlarging R_0 and/or R_f, as well as commanding the aircraft along a helical trajectory (similar to a spring) to lose excess altitude, before joining the path at the start of the first arc.

To form the 3-D path, we note that a gliding aircraft that is rolled into a steady, coordinated turn at a constant bank angle ϕ and flies at a constant descent angle will trace a helical path γ on an imaginary circular cylinder with radius R. Thus, the 3-D path can be formed by a straight line at a constant path angle that joins two arc sections. To simplify the design, we have not included the $\{L_\alpha R_\beta L_\gamma\}$ or $\{R_\alpha L_\beta R_\gamma\}$ type paths, these will be addressed in future work. The relationship between ϕ and γ is given by:

$$\cot \gamma_{\phi_{0,f}} = \frac{V_{TAS}}{V_S} \cos \phi_{0,f}, \tag{4}$$

where V_S is the descent rate of the gliding aircraft, and V_{TAS}/V_S approximates L/D. The path angle γ has also been limited between 0 and $-10°$ to comply with the aircraft dynamic constraints.

Now, the altitude lost while traversing the two arc sections can be calculated as:

$$S_{0,f} = \frac{1}{2\pi} \frac{\left\| \sum_{0,f} \right\|}{R_{0,f}} S_{\phi_{0,f}}. \tag{5}$$

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

Giving the altitude where the arc sections join, the 3-D line as:

$$z_{\text{arc}_0} = z_0 - S_0,$$
$$z_{\text{arc}_f} = z_f + S_f, \tag{6}$$

where, z_0 is the altitude at the start of a forced landing, and z_f is the desired altitude to achieved at the final approach point. Given the terminal points on the arcs $P_{\text{arc}_0} = [x_{\text{arc}_0}\ y_{\text{arc}_0}\ z_{\text{arc}_0}]$ and $P_{\text{arc}_f} = [x_{\text{arc}_f}\ y_{\text{arc}_f}\ z_{\text{arc}_f}]$, it is then a straightforward process to obtain Γ_{line}. The relationship between the different elements of the 3-D path $\Gamma = \Gamma_{\text{arc}_0} \bigcup \Gamma_{\text{line}} \bigcup \Gamma_{\text{arc}_f}$ is illustrated in Figure 3.

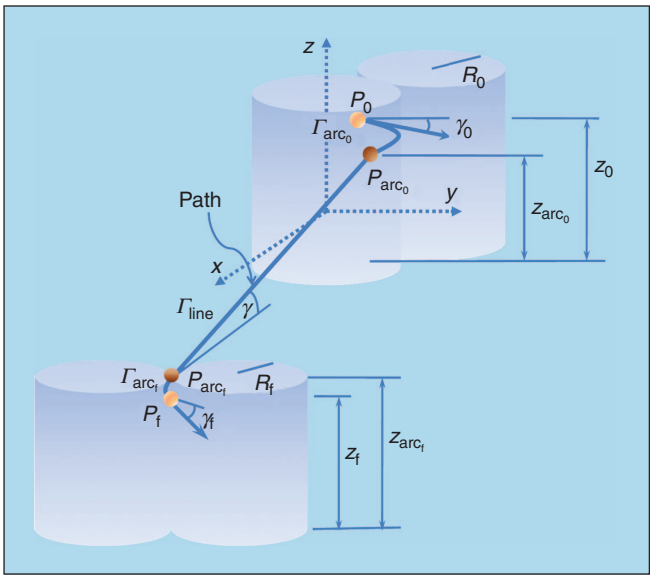

Figure 3. *Relationship between elements of the generated 3-D flight path. The generated path is* $\Gamma = \Gamma_{arc_0} \bigcup \Gamma_{line} \bigcup \Gamma_{arc_f}$.

Guidance and Control

In designing our guidance and control algorithms, we have used a path following, rather than a trajectory tracking approach, where the objective is to be on the path rather than at a certain point at a particular time. This removes the time dependency of the problem and allows the design of simpler controllers.

Our lateral guidance approach is based on the work presented in [15]. However, we have enhanced this algorithm to include wind information in the guidance logic, rather than merely treating wind as an adaptive element for the control system. This addition has demonstrated robust, linear path following in strong winds. Second, by making a simple assumption in formulating the circular path following equation, we have simplified the guidance logic without sacrificing performance. In addition, we have implemented a longitudinal guidance and control element that caters for the dynamics of powerless flight. Finally, following the well-established aircraft control design procedures [12], we have separated our design into two modes: an inner control loop that provides aircraft dynamic stability and an outer guidance loop that generates the required acceleration and position commands to follow a path. The design of the inner loop controls will not be presented in this article.

For lateral guidance, a reference point P_{ref} is selected on the desired trajectory and used to generate an acceleration command. As shown in Figure 4, P_{ref} is located a distance L_1 ahead of the vehicle and, at each point in time, a circular path (dotted line) can be defined by the position of L_1, the vehicle position, and tangential to V, the aircraft velocity vector.

The acceleration required to follow the instantaneous circular segment, for any radius R, is then given by:

$$a_{s_{\text{cmd}}} = \frac{V^2}{R} = 2\frac{V^2}{L_1}\sin\eta. \tag{7}$$

Thus, the guidance logic will tend to rotate the aircraft so that its velocity direction will always approach the desired path at an angle that is proportional to the relative distance between vehicle and path. For following a straight line, we can model the vehicle kinematics as shown in Figure 5.

Now, consider the UAV in a straight glide at an arbitrary position relative to the path between waypoints P_1 and P_3, and at a heading ψ [Figure 5(a)]. Given the aircraft velocity and position in the {North, East} reference frame and the angular measurements defined in Figure 5(b), we can obtain the position and velocity components in the $\{x_{\text{track}}, y_{\text{track}}\}$ reference frame by:

$$\vec{V}_{\text{track}} = \text{T}_\psi \vec{V},$$
$$\vec{W}_{\text{track}} = \text{T}_\psi \vec{W}, \tag{8}$$

where the rotation matrix is

$$\text{T}\psi = \begin{bmatrix} \cos(\psi_{12} - \pi/2) & -\sin(\psi_{12} - \pi/2) \\ \sin(\psi_{12} - \pi/2) & \cos(\psi_{12} - \pi/2) \end{bmatrix}. \tag{9}$$

The cross-track velocity can then be written as:

$$\dot{y}_{\text{track}} = V_{y_{\text{track}}} + W_{y_{\text{track}}}$$
$$= -V\ \sin(\psi(t) - \psi_{12}) - W\sin(\psi_w - \psi_{12}), \tag{10}$$

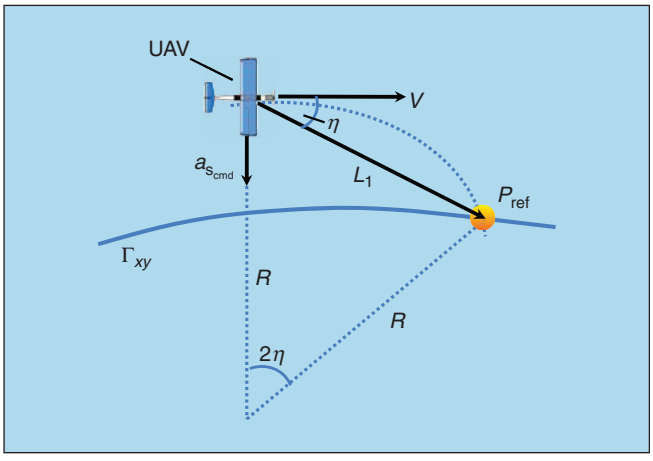

Figure 4. *Diagram showing the lateral guidance law.*

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

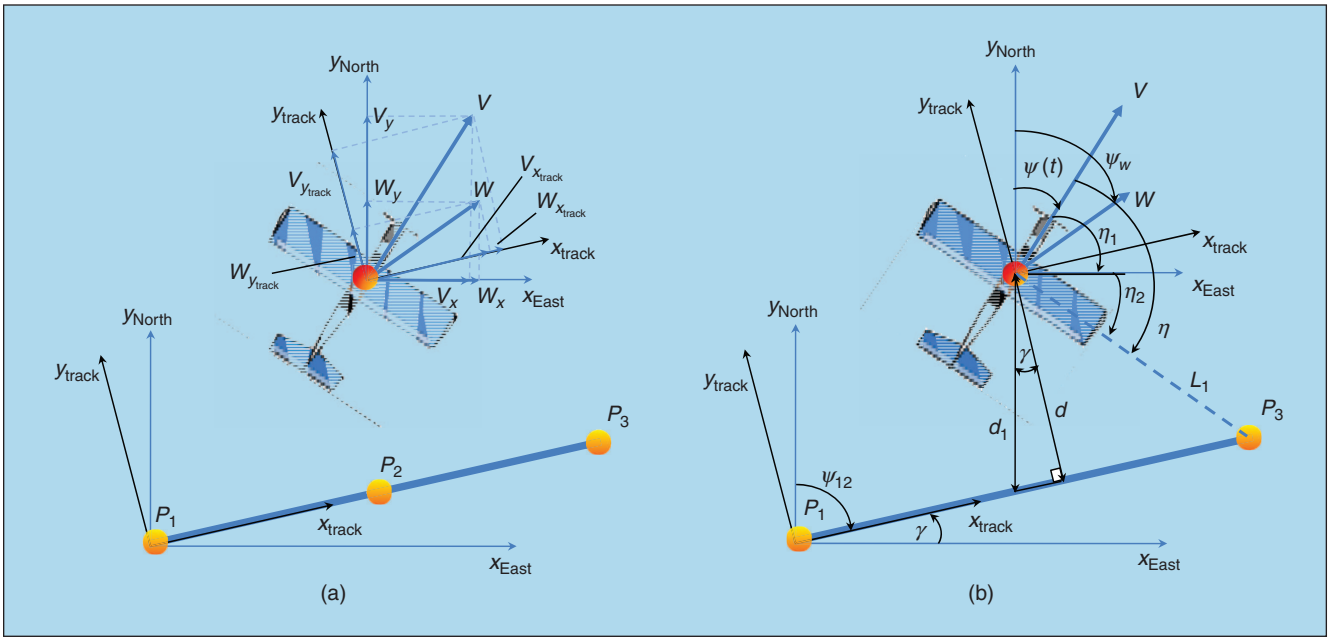

Figure 5. *Vehicle kinematics for straight line following, showing (a) the relationship between the aircraft velocity V and the expected/average wind velocity W, and (b) the relationship between the aircraft bearing $\psi(t)$, the path bearing ψ_{12}, the wind bearing ψ_w, and the angle η. In addition, the relationship between the cross-track error d, the hypotenuse d_1 and the look-ahead distance L_1 is also shown.*

and assuming η is small, we get

$$\sin\eta \approx \eta_1 + \eta_2, \tag{11}$$

and

$$\eta_1 \approx \frac{d}{L_1}, \eta_2 \approx \frac{\dot{d}}{V}, \tag{12}$$

where $\dot{y}_{\text{track}}$ has been relabeled as $\dot{d}$, and d is the cross-track error. We can obtain d by letting

$$d_1 = n_{ac} - \tan(\delta)e_{ac} \tag{13}$$

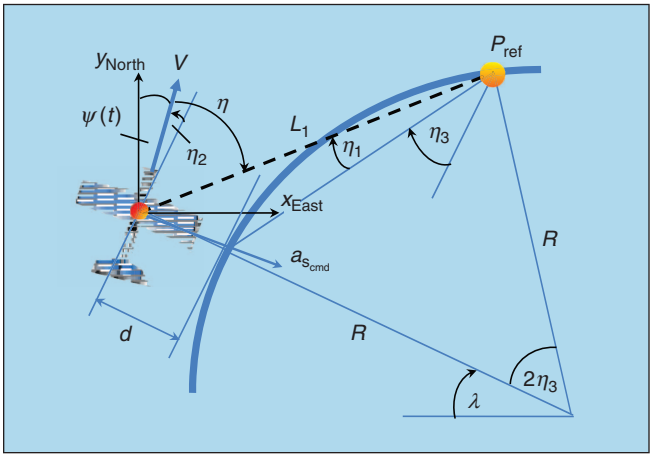

Figure 6. *Vehicle kinematics for following an arc.*

and

$$d = d_1\cos\delta. \tag{14}$$

Combining (7) with (12), we obtain

$$a_{S_{\text{cmd}}} = 2\frac{V}{L_1}\left(\dot{d}\frac{V}{L_1}d\right). \tag{15}$$

For following an arc of circumference, we can model the vehicle kinematics as shown in Figure 6.

Here, the angles η_1 and η_2 are assumed to be small, but η_3 is not necessarily small

$$\eta_1 \approx 0,\ \eta_2 \approx 0,\ |\eta_3| \gg 0. \tag{16}$$

As shown in [15], using (7) and (16), we can estimate

$$\sin\eta_3 \approx \frac{L_1}{2R}, \tag{17}$$

and define

$$c = \cos\eta_3 \approx \sqrt{1 - \left(\frac{L_1}{2R}\right)^2}. \tag{18}$$

Then, using small angle assumptions for η_1 and η_2, we can show

$$a_{S_{\text{cmd}}} \approx \frac{2V^2}{L_1}\{\eta_1\cos\eta_3 + \eta_2\cos\eta_3 + \sin\eta_3\}, \tag{19}$$

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

with

$$\eta_1 \approx \frac{d}{L_1}\cos\eta_3, \quad \dot{d} = V\ \sin\eta_2 \approx V\eta_2, \tag{20}$$

and applying (17) and (18), (19) becomes:

$$\frac{2V^2}{L_1}\sin\eta = \frac{2V^2c^2}{{L_1}^2}d + \frac{2Vc}{L_1}\dot{d}\frac{V^2}{R}. \tag{21}$$

Now, if we assume that a good yaw damper can be designed to damp the aircraft Dutch roll motion and reduce the sideslip to zero, then we can neglect the second term on the RHS of (21) to obtain:

$$a_{S_{cmd}} = \frac{2V^2}{L_1}\sin\eta = \frac{2V^2c^2}{{L_1}^2}d + \frac{V^2}{R}. \tag{22}$$

To convert the acceleration to a desired roll command and simplify calculations, we assume that the aircraft maintains sufficient lift to balance weight, even though banked at an angle ϕ. This gives

$$L\cos\phi = W = \mathrm{mg}, \qquad L\sin\phi = \mathrm{ma_s}, \tag{23}$$

and

$$\phi_d = \tan^{-1}\left(\frac{a_s}{g}\right). \tag{24}$$

In reality, a gliding aircraft will never overcome gravity but will descend with a vertical velocity $\dot{h}$; this is an inherent limitation in the forced landing problem. We have not included additional terms in the equation for wind effects, as we have done with the case of following a straight line. The reason is that, in this case, the vehicle groundspeed (as a surrogate for inertial velocity) is used for V in (7) at each instant in generating the acceleration command. Since the groundspeed is a function of the airspeed and windspeed, the guidance logic accounts for the inertial velocity changes due to wind and adapts to the situation accordingly.

For testing purposes, we have specified that the required path following performance be a horizontal (lateral) and vertical (longitudinal) cross-track error at the approach point of no greater than 2 m (approximately 6.56 ft) and a maximum vertical and horizontal deviation of no greater than 30 m (approximately 100 ft) on average. These upper and lower bounds are commonly accepted as the performance standard for general aviation aircraft [8].

Results and Discussion

A total of 128 simulations have been performed in which the aircraft initial altitude and wind conditions were allowed to vary, while the initial and final aircraft headings and positions were kept constant. A Monte-Carlo simulation using a larger input set will be performed at a later date. For these early experiments, we have also assumed perfect knowledge of the wind conditions and no errors in the sensor readings. A sample of the test data is included in Table 1. Table 1(a) shows the initial and desired final positions used to plan a path from 1,640 to 500 ft in calm conditions, whereas Table 1(b) shows the positions for a descent from 850 to 500 ft in winds, with a maximum windspeed of 9 m/s. As we are concerned with gliding flight, the required airspeed at the approach point is absorbed by the flight path angle requirement at this point.

Table 1. Sample test data for the path planning and path following algorithms.

	Initial Condition	Final Condition
(a) No wind		
x (m)	−199	885
y (m)	37	133
z (ft)	1,640	500
ψ (°)	10	90
γ (°)	0	−6
(b) With wind		
x (m)	−199	885
y (m)	37	133
z (ft)	850	500
ψ (°)	10	90
γ (°)	0	−6

Figure 7 shows that in planning a path from a high-initial altitude [Table 1(a)], the algorithm generates the required number of helix spirals to bleed off the excess altitude, before joining the spirals with the arc-line-arc Dubins path (solid black line). This satisfies flight path angle constraints and prevents excessive stress to the UAV structure. The horizontal and vertical track errors at the approach point are 0.3 and 1.3 m, which are well within our stated tolerance and comparable to the results obtained for other flight path angles. The difference in altitudes between aircraft and path at the start is due to

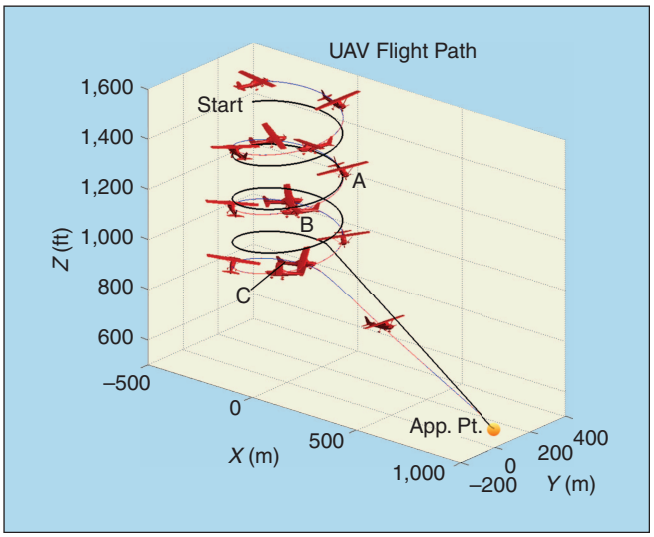

Figure 7. *Path planning and following for a forced landing from a high-initial altitude in no wind conditions.*

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

the planning algorithm rounding the required number of helix spirals to the nearest complete (360°) spiral turn, to preserve the desired initial heading. Notwithstanding this, the aircraft is still able to converge onto the path at Point A. However, at Point B, the aircraft descends below the path and reaches a maximum vertical deviation of 600 ft at Point C (the horizontal error here was ≈9 m) but recovers to intercept the approach point with the errors stated earlier.

The poor vertical path following in the first half of the descent may be caused by using a fixed airspeed in constructing the path, as it is not possible to predict beforehand the actual airspeed due to the control actions. In a descent, the varying airspeed results in a nonuniform loss in altitude. In addition, an aircraft rolled into a continuous banking motion will also experience some amount of yawing motion called sideslip, no matter how good the yaw damper may be; this in turn increases the altitude lost. Thus, the amount of loss in altitude factored into the path planning equations is ideal at best and does not fully take into account the associated loss in altitude due to varying airspeeds and other atmospheric effects. Hence, the current solution relies on the path following algorithm being robust to these uncertainties in guiding the aircraft to the desired approach point. A possible alternative is to increase the path angle of the initial helices to more closely match that of the straight segment and/or increase the radii of the helices such that the number of spirals is reduced. These will help reduce the amount of altitude loss due to sideslips and a prolonged banking action.

Next, we show the performances of our path following algorithm in winds [Table 1(b)] and compare the results with those obtained using the original path following algorithm in [15], hereby referred to as the unmodified nonlinear guidance (UNG) algorithm. Two different wind scenarios are chosen for illustration. Figure 8(a) shows the aircraft able to follow the desired path (solid black line) in a 6 m/s South–South–Westerly wind (green arrows), whereas Figure 8(b) shows the aircraft following the prescribed path in changing winds from the North–North–East and South–South–West. For the second case [Figure 8(b)], we have tried to introduce a degree of realism into our simulations, by including errors in the airspeed and global positioning system (GPS) velocity measurements, which in turn produce errors in the estimated wind velocity. Taking the worst case scenario, V_{TAS} is assumed to have a maximum error of ±3 m/s, and the GPS velocity a maximum error of ±1 m/s in magnitude (groundspeed), and a maximum error of ±1° in bearing. The magnetic compass measurement

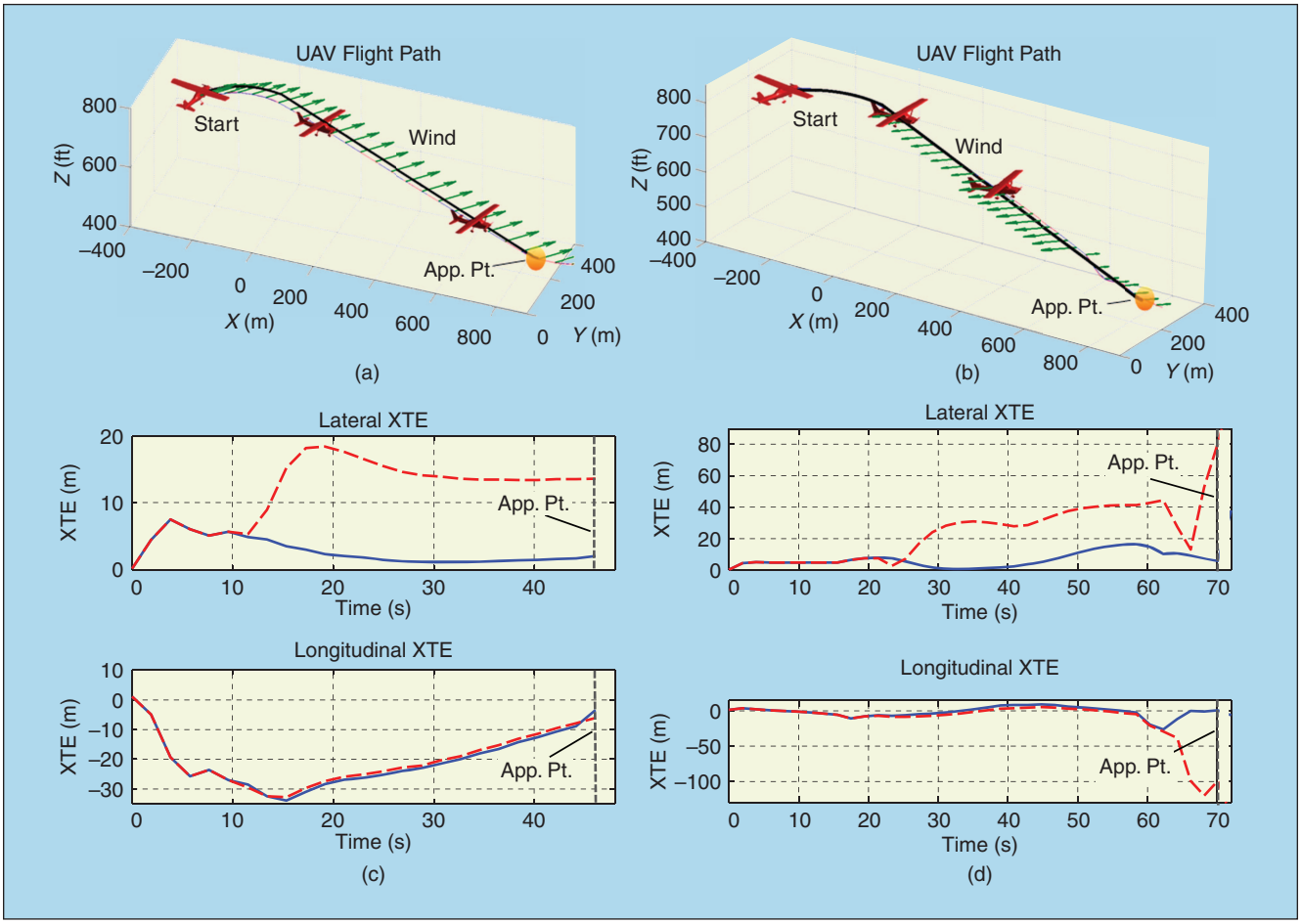

Figure 8. *Path planning and following in ambient winds, showing (a) aircraft response in 6 m/s wind from SSW; (b) aircraft response in 0–9 m/s changing wind from NNE and SSW; (c) horizontal and vertical track errors for case (a); and (d) horizontal and vertical track errors for case (b).*

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

was also assumed to have a maximum error of ±3° in bearing. These error values are consistent with instrument errors observed on small aircraft and give rise to a maximum error of ±1 m/s in the estimated north and east wind component values, which can be modeled as a Gaussian distribution. With these stated errors, a wind bearing error of up to 42° has been observed in simulations.

In the scenario depicted in Figure 8(b), the wind velocities have been programmed to vary in magnitude and direction every 20 s, with a maximum windspeed of 9 m/s and with the vertical wind velocity kept constant at zero. From Figure 9, we see that, in the first 20 s, there is calm, and then the wind blows from 225° at ≈6 m/s for the next 20 s. In the next epoch, it increases to ≈9 m/s, before assuming a 180° phase change and decreasing to ≈3 m/s for the last 20 s. Although this constant change in wind may not represent the actual conditions experienced during a forced landing, it nonetheless presents a very challenging environment to test the robustness of our guidance algorithms. Note that the green arrows in Figure 8 show only the general direction of the incident wind vectors—the actual turbulence fields in the longitudinal, lateral, and vertical directions (including wind gusts and lulls) as traversed by the aircraft are depicted in Figure 10.

As shown in the top halves of Figure 8(c) and (d), our path following algorithm produced lateral errors at the approach point of 1.8 and 5.5 m, respectively (blue lines), whereas that of UNG was 14.1 and 80 m for the two different wind conditions (red dashed lines). Although UNG did not include a longitudinal path following component, we have nonetheless plotted the vertical track error to show what might have transpired had that lateral guidance algorithm, coupled with our longitudinal guidance algorithm, been used to follow the path. As shown in the lower halve of Figure 8(c) and (d), the vertical track errors for our guidance algorithm is approximately 1.2 and 1.5 m for the two wind conditions, respectively, whereas that for the case of UNG coupled with our longitudinal guidance algorithm is 1.5 and 100 m. Thus, it can be clearly seen that our path following algorithm is robust to wind perturbations and uncertainties in wind measurements in both the semireal and ideal conditions. Our guidance algorithm also outperforms the UNG algorithm, especially in conditions of varying winds, where it would seem that the UNG algorithm cannot deal appropriately with significant changes (≥180°) in wind directions. Further, the average lateral and longitudinal path errors incurred while using our guidance algorithms (for the two different wind scenarios) are well within 100 ft, which as mentioned earlier is commonly accepted as the maximum allowable path deviation for general aviation aircraft.

From the simulations, we have also observed that the path following algorithm is able to contain the errors at the approach point, within the stated tolerances, for windspeeds not exceeding 7 m/s. In stronger winds, these errors can degrade to >20 m horizontally and up to 5 m vertically, or the aircraft may lose control and crash. A possible explanation is that the small size and weight (5.55 kg) of our model aircraft, as well as the limited thrust available means that it cannot achieve the necessary control authority to overcome strong winds and gusts. We have also

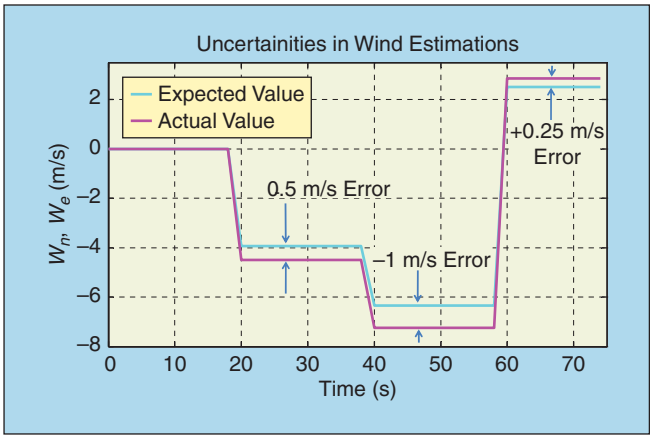

Figure 9. *Input winds for the scenario in Figure 8(b), showing changes in windspeed over time, and the errors between the expected (cyan) and actual (magenta) north and east wind component values.*

noticed from simulations that the vertical track error at the approach point is >7 m in sinking air of 1 m/s. Once again, this relatively poor performance may be due to the structural and aerodynamic factors stated earlier. However, when compared with our previous work as discussed in [9] and [10], we find that the performances of our current algorithms are far superior. We believe that this improved performance may be largely because we have allowed the airspeed and flight path angle to vary to counter different wind scenarios.

Conclusions and Future Work

In this article, we have presented, in what we believe to be the first of its kind, the design and simulated testing of automated path planning and control strategies for a fixed-wing UAV executing an unpowered descent for landing during an emergency. Simulated test results demonstrate the ability of

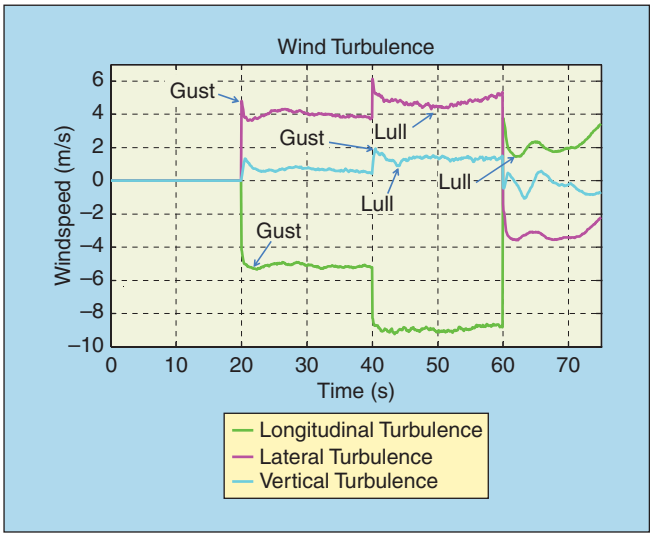

Figure 10. *Wind turbulence for the scenario in Figure 8(b), showing examples of gusts and lulls in the longitudinal, lateral, and vertical directions.*

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.

the gliding aircraft to follow the prescribed path in changing winds, with average path deviation errors that are comparable to or even better than that of manned, powered aircraft. To further verify the performances of our algorithms, we are currently preparing for flight tests with a Boomerang UAV.

In the future, we will experiment with different techniques to reduce the vertical track error when the aircraft is following a helix spiral, and several options have already been proposed in the preceding section. Second, we desire to extend the path planning component to include a replanning capability for cases where the efficacy of the original plan is reduced or nullified. This could be due to winds that are simply too strong for the aircraft to overcome or when a better landing site is identified as the aircraft nears the ground. These modular enhancements will also be progressively flight tested.

Acknowledgments

The work presented in this article is sponsored by the Australian Postgraduate Award, the QUT Vice-Chancellor's Top-Up Award, and the Commonwealth Scientific and Industrial Research Organization Information and Communication Technologies (CSIRO ICT) Scholarship Top-Up Award.

Keywords

UAV, forced landing, path planning, path following, control, MATLAB, Simulink.

References

[1] B. Cobleigh, "Ikhana: A NASA UAS supporting long duration earth science missions," Tech. Rep. TR 20070026136_2007022414, 2006.

[2] I. Velinov. (2008, July 23). NASA's Ikhana helps in California's firefighting effort, *The Epoch Times* [Online]. Available: http://en.epochtimes.com/

[3] J. Wise. (2008, Nov. 26). Civilian UAVs: No pilot, no problem, *Popular Mechanics* [Online]. Available: http://www.popularmechanics.com/

[4] C. Redelinghuys, "A flight simulation algorithm for a parafoil suspending an air vehicle," *J. Guidance, Control, Dyn.*, vol. 30, pp. 791–803, May–June 2007.

[5] Australian Research Centre for Aerospace Automation. UAV Challenge Technical Committee. (2010, Apr. 28). Search and rescue challenge: Mission, rules, and judging criteria [Online]. Available: http://www.uavoutbackchallenge.com.au/uavoutbackchallenge/register/UAVChallengeRulesSearchAndRescur2010v1_1.pdf.

[6] D. R. Nelson, D. B. Barber, T. W. Mclain, and R. W. Beard, "Vector Field Path Following for Miniature Air Vehicles," *IEEE Trans. Robot.*, vol. 23, no. 3, pp. 519–529, 2007.

[7] C. P. Froeschner. (2008). UAS, *Flying Safety4* [Online]. Available: http://findarticles.com/

[8] CASA, *Visual Flight Rules Guide*, 2nd ed. Canberra, Civil Aviation Safety Authority Australia, 2007.

[9] P. Eng, L. Mejias, D. Fitzgerald, and R. Walker. (2007). Simulation of a fixed-wing UAV forced landing with dynamic path planning. *Proc. Australasian Conf. Robotics and Automation*, Brisbane, Australia [Online]. Available: www.araa.asn.au/acra/acra2007/contents.html

[10] D. Fitzgerald, L. Mejias, P. Eng, and X. Liu. (2007). Towards flight trials for an autonomous UAV emergency landing using machine vision. *Proc. Australasian Conf. Robotics and Automation*, Brisbane, Australia [Online]. Available: www.araa.asn.au/acra/acra2007/contents.html

[11] D. L. Fitzgerald, "Candidate landing site selection for UAV forced landings using machine vision," Ph.D. dissertation, Sch. Eng. Sys., BEE, QUT, Brisbane, Australia, 2006.

[12] R. C. Nelson, *Flight Stability and Automatic Control*, 2nd ed. New York: McGraw-Hill, 1998.

[13] L. E. Dubins, "On curves of minimal length with a constraint on average curvature with prescribed initial and terminal positions and tangents," *Amer. J. Math.*, vol. 79, no. 3, pp. 497–516, 1957.

[14] G. Ambrosino, M. Ariola, U. Ciniglio, F. Corraro, A. Pironti, and M. Virgilio, "Algorithms for 3-D UAV path generation and tracking," in *Proc. 45th IEEE Conf. on Decision & Control*, San Diego, CA, 2006, pp. 5275–5280.

[15] S. Park, J. Deyst, and J. P. How, "Performance and Lyapunov stability of a nonlinear path-following guidance method," *J. Guidance, Control, Dyn.*, vol. 30, pp. 1718–1728, Nov.-Dec. 2007.

[16] S. M. Lavalle, *Planning Algorithms*. New York: Cambridge Univ. Press, 2006.

[17] T. G. McGee and J. K. Hedrick, "Pathplanning and control for multiple point surveillance by an unmanned aircraft in wind," in *Proc. American Control Conf.*, Minneapolis, MN, 2006, pp. 4261–4266.

[18] H. Chitsaz and S. LaValle, "Time-optimal paths for a Dubins airplane," in *Proc. 46th IEEE Conf. Decision and Control*, New Orleans, LA, Dec. 12–14, 2007, pp. 2379–2384.

[19] S. Iyanaga and Y. Kawada, "Pontryagin's [sic] maximum principle," in *Encyclopedic Dictionary of Mathematics*, Mathematical Society of Japan and K. Ito, Eds. Cambridge: MIT, 1980, pp. 295–296.

Pillar Eng received a B.Eng. degree in avionics from QUT, Australia, in 2004. He is currently pursuing his Ph.D. degree at the Australian Research Centre for Aerospace Automation (ARCAA) and is a Student Member of the IEEE.

Luis Mejias received a B.Eng. degree in electronics from Universidad Nacional Experimental Politécnica (UNEXPO), Venezuela, as well as an M.Sc. degree in telecommunication systems and a Ph.D. degree in robotics and automation from Universidad de Politécnica de Madrid (UPM), Spain. His research interests include UAV navigation, guidance, and control for both fixed and rotary-wing aircraft. He is currently employed as a research scientist with ARCAA and is also a Member of the IEEE.

Rodney Walker holds a B.Eng. degree in electronic systems, an App.Sc. degree in computing and a Ph.D.degree in satellite navigation from QUT. From 1998 to 2005, he was responsible for the GPS payload on Australia's Federation Satellite, working closely with National Aeronautics and Space Administration Jet Propulsion Laboratory (NASA JPL) during this time. Since 2000, he has directed his interests to ICT in aviation and created the ARCAA, which now has more than 30 full-time staff. He is a professor at QUT and has been a Member of the IEEE since 2001.

Daniel Fitzgerald received a B.Eng. degree in avionics in 2002 and a Ph.D. in computer vision in 2007 from QUT. He is a research scientist at the CSIRO ICT Centre's Aerial Robotics group. His research interests include classification techniques using neural networks and feature tracking for UAV inspection tasks and forced landing.

Address for Correspondence: Pillar Eng, Australian Research Centre for Aerospace Automation, School of Engineering Systems, Queensland University of Technology, Brisbane QLD 4032, Australia. E-mail: p.eng@qut.edu.au.

Authorized licensed use limited to: QUEENSLAND UNIVERSITY OF TECHNOLOGY. Downloaded on June 21,2010 at 02:17:16 UTC from IEEE Xplore. Restrictions apply.