AIAA SciTech
13-17 January 2014, National Harbor, Maryland
AIAA Guidance, Navigation, and Control Conference

AIAA 2014-0263

Closing the Loop in Dynamic Soaring

John J. Bird* and Jack W. Langelaan†
The Pennsylvania State University, University Park, PA 16802 USA
Corey Montella‡, John Spletzer§, and Joachim Grenestedt¶
Lehigh University, Bethlehem, PA 18015 USA

This paper examines closed-loop dynamic soaring by small autonomous aircraft. Wind field estimation, trajectory planning, and path-following control are integrated into a system to enable dynamic soaring. The control architecture is described, performance of components of the architecture is assessed in Monte Carlo simulation, and the trajectory constraints imposed by existing hardware are described. Hardware in the loop simulation using a Piccolo SL autopilot module are used to examine the feasibility of dynamic soaring in the shear layer behind a ridge, and the limitations of the system are described. Results show that even with imperfect path following dynamic soaring is possible with currently existing hardware. The effect of turbulence is assessed through the addition of Dryden turbulence in the simulation environment.

I. Introduction

For certain missions, such as ocean monitoring, dynamic soaring has the potential to greatly enhance range and endurance of small UAVs. Albatrosses and petrels are similar in both size and weight to small unmanned aircraft, and they routinely employ dynamic soaring to travel thousands of kilometers.[1,2,3] Deittert et al. show that the probability of winds that permit dynamic soaring by small UAVs exceeds 50% in the southern oceans, and this probability is roughly 90% for albatrosses.[4] The difference is because albatrosses can descend to very low altitude (dragging a wingtip in the water) and UAVs must maintain a significant safety margin.

Gordon describes results of flight tests of human-piloted dynamic soaring in the boundary layer, using a Blanik L-23 as test aircraft.[5] While the conclusion was that dynamic soaring did indeed result in an improvement in flight efficiency, safety constraints precluded a demonstration of net energy neutral flight (the minimum allowable altitude kept the aircraft well above the region of maximum wind gradient and the requirement to keep landing gear deployed reduced the glider's L/D).

Dynamic soaring has been expressed as a problem in trajectory optimization given a known wind field[6] where energy extracted is maximized or required wind speed to maintain flight is minimized. Robust trajectory optimization (with the goal of ensuring safety while enabling at least energy neutral trajectories) is described by Flanzer et al.[7]

However, *a priori* wind field knowledge may not be available, and the wind field has a significant effect on the trajectory that must be flown to enable energy neutral (or energy gaining) flight. In earlier work we described a method for mapping the mean wind field using polynomial basis functions;[8] here we close the loop and integrate the wind mapping algorithm (modified to use splines as basis functions, and combined with a model-based estimator[9]) with a trajectory planner and a commercially available flight control system (a CloudCap Technologies Piccolo SL). CloudCap's HiL simulation environment is used to evaluate the performance of the closed loop dynamic soaring system.

This paper describes the architecture of the dynamic soaring system (Section II). It then gives a discussion of the wind field mapping algorithm (Section III). Trajectory planning is described next, focusing on the

[*] Graduate Student, Department of Aerospace Engineering, Student Member AIAA.
[†] Associate Professor, Department of Aerospace Engineering, Associate Fellow AIAA.
[‡] Graduate Student, Computer Science and Engineering Department
[§] Associate Professor, Computer Science and Engineering Department
[¶] Professor, Department of Mechanical Engineering and Mechanics

effects of constraints imposed by the autopilot (Section IV). Trajectory following control and adaptations required to use the Piccolo are discussed (Section V). Results of simulations of dynamic soaring flights across a lee-ridge type shear layer are detailed in Section VI.

II. A Control Architecture for Dynamic Soaring

A general architecture for a system to enable autonomous dynamic soaring is shown in Figure 1.

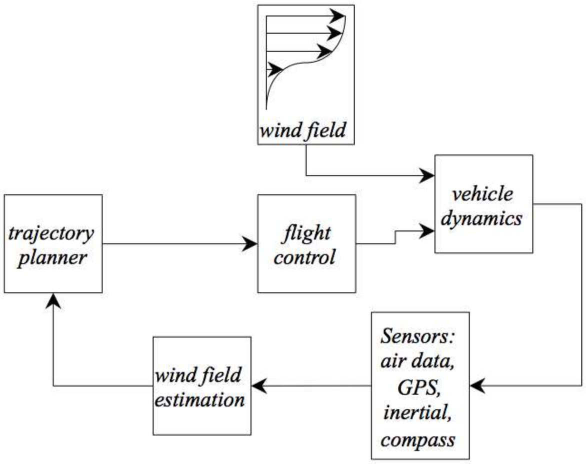


Figure 1. Top-level schematic of closed-loop dynamic soaring architecture.

A wind field estimation block uses data from aircraft sensors (GPS/INS, air data, compass) to compute the wind velocity at the current aircraft location. This wind velocity information is then used to compute an estimate of the wind field (i.e. the spatial variation in wind speed and direction– see Section III). The wind field estimate is then used by the trajectory planner (Section IV) to compute a flight path that maximizes the change in total energy over one cycle. If net energy change over one cycle is zero (or positive, meaning that energy is gained) then dynamic soaring is possible. Finally a flight controller follows the computed trajectory (Section V).

The estimate of wind field is continuously updated. This allows the system to account for temporal changes in the wind field and allows continuous refinement as data is gathered. The wind field estimate is essentially a time-averaged estimate of the wind field, with the averaging time long enough that the effects of gusts are smoothed but short enough that temporal variations in mean wind can be captured.

New trajectories are computed as the wind field estimates change. This system is thus robust to changes in wind conditions.

The flight control block attempts to follow the desired trajectory while rejecting disturbances such as gusts. It will become evident later in this paper that compensating for gusts increases energy consumption: for robustness to gusts a trajectory that is slightly energy positive can be planned.

Without prior knowledge of the wind field, its characteristics must be determined before dynamic soaring can be attempted. The wind field is estimated here using the process described by Langelaan et al.,[10] then modeled by basis splines and a parameter estimator for an explicit wind model. The wind estimation system requires aircraft orientation, inertial velocity, and air-relative velocity, while modeling only requires the current wind estimate and the aircraft position. The wind field estimation and modeling routines are implemented in a MATLAB application which communicates with the Piccolo to receive state information.

The trajectory planner has two parts: an off-line computation of trajectories for a number of candidate shear structures, and an on-line trajectory selection to match the observed shear. The on-line planner identifies the candidate shear model best matching the observed wind profile and returns a state trajectory which was planned to maximize energy gained for that wind profile. Path updates occur every dynamic soaring cycle.

The trajectory is controlled by setting roll and pitch angle targets in the Piccolo outer loops (the inner rate loops are inaccessible to the end user). The DS controller flies the selected trajectory by linearizing the trajectory around the nominal flight path and using a feedback law to fly out error in the path.

This system architecture allows the complex dynamic soaring trajectory to be flown while retaining the robust communications and dynamic inversion controller provided by the Piccolo. The Piccolo however, was not designed for accurately performing the aggressive maneuvers required for dynamic soaring. Inaccessibility of the inner control loops complicates fine trajectory control, limiting the shear environments which can be exploited. Despite these limitations, the system presented provides an initial step in closing a loop around the dynamic soaring problem, allowing an aircraft with no prior knowledge of a wind environment to explore the wind field and extract energy from the shear present.

III. Wind Field Mapping

Efficient computation of dynamic soaring trajectories requires knowledge of the wind field sufficiently detailed to capture features of the shear to be exploited but compact enough for rapid evaluation. In previous work, polynomial functions were shown to be useful in wind modeling.[8] However, choosing a polynomial of appropriate order to fully describe the wind field without over-fitting is difficult, and polynomial functions are unable to model functions with large and rapidly changing curvature such as the power law wind profile seen in the atmospheric boundary layer or logistic curves seen in the lee of ridges.[11] Polynomial basis splines are used in this paper as they can be evaluated easily, while allowing complex wind fields to be adequately described. In order to determine the fine structure of the shear for trajectory selection, the basis spline model is combined with a filter that estimates the parameters of an assumed shear structure.

III.A. Basis Splines

A spline is a piecewise polynomial function, and can be constructed by piecewise summation of polynomials. The nature of a spline can be visualized by dividing the domain of a function into a set of segments separated by "knots" – points where the k^{th} derivative of the function is allowed to be discontinuous, with k determining the order of the spline. Owing to its piecewise nature a spline can be written as a linear combination of simpler functions given a suitable basis. This form is known as the basis spline (or B-spline) form, and allows the spline function to be written as a linear mapping:[12]

$$\mathcal{S}(h) = \sum_{i=0}^{g+k-1} c_i N_i(h) \tag{1}$$

Where c are the coordinates specifying a particular function. After defining the spline order, k, and knot locations, $\lambda_0 ... \lambda_g$, the value of the spline basis, N, can be determined using a triangular scheme detailed by Diercx.[12]

III.B. Modeling an Arbitrary Wind Field

Since the spline $\mathcal{S}(z)$ represents a linear mapping, it can be implemented in a Kalman filter to build a model of the wind environment as measurements of the wind field are taken.[9] The Kalman filter states are formed by concatenating the vector of spline coordinates in each direction:

$$\mathbf{X} = \begin{bmatrix} \mathbf{c}_{north} \\ \mathbf{c}_{east} \\ \mathbf{c}_{down} \end{bmatrix} \tag{2}$$

Assuming that changes in the wind field are small and random so that the state transition is trivial, the Kalman filter can be constructed conventionally with the prediction step proceeding:

$$\begin{aligned} \hat{\mathbf{X}}_{t|t-1} &= \hat{\mathbf{X}}_{t-1|t-1} \\ \hat{\mathbf{P}}_{t|t} &= \hat{\mathbf{P}}_{t|t-1} + \mathbf{Q}_t \end{aligned} \tag{3}$$

Since the spline basis forms a partition of unity,[12] a uniform bias in the coefficients will result in an identical bias in the wind model for any point in the wind field. This property of the basis spline offers a convenient way to select a process noise with physical meaning – the process noise is chosen to be the expected uniform change in the wind field between iterations of the Kalman filter, allowing it to "walk" with time. The observation model is defined as a block diagonal of the spline basis at the current aircraft location:

$$\mathbf{H}_t = \begin{bmatrix} \mathbf{N}_{north}(h) & 0 & 0 \\ 0 & \mathbf{N}_{east}(h) & 0 \\ 0 & 0 & \mathbf{N}_{down}(h) \end{bmatrix} \tag{4}$$

Where $\mathbf{N}_{(\cdot)}(h)$ is a row-vector defining the spline basis at the current aircraft location. While the state and observation model structures described here permit different spline models to be used for each coordinate direction, identical models are used in this paper to reduce the number of times the spline basis must be computed.

The remainder of the Kalman filter proceeds conventionally:

$$\begin{aligned} \mathbf{K}_t &= \hat{\mathbf{P}}_{t|t-1}\mathbf{H}_t^T(\mathbf{H}_t\hat{\mathbf{P}}_{t|t-1}\mathbf{H}_t^T + \mathbf{R}_t)^{-1} \\ \hat{\mathbf{X}}_t &= \hat{\mathbf{X}}_{t|t-1} + \mathbf{K}_t(\mathbf{w} - \mathbf{H}_t\hat{\mathbf{X}}_{t|t-1}) \\ \hat{\mathbf{P}}_{t|t} &= (\mathbf{I} - \mathbf{K}_t\mathbf{H}_t)\hat{\mathbf{P}}_{t|t-1} \end{aligned} \tag{5}$$

The wind measurement covariance, $\mathbf{R}$ and measured wind velocity, $\mathbf{w}$, are determined using the method of direct wind field computation described by Langelaan et al.[10] Even with an arbitrarily detailed spline model (unrestricted in the number of coefficients required to define the model), the quantity $(\mathbf{H}_t\hat{\mathbf{P}}_{t|t-t}\mathbf{H}_t^T + \mathbf{R}_t)$ reduces to a 3×3 matrix, so the filter runs rapidly.

While the filter is developed here in north/east/down coordinates, any three-component system could be used to model the three-dimensional wind field. Modeling in cylindrical coordinates has been previously demonstrated,[8] and offers natural intuition in the way we typically think of wind (ie. speed and direction). The wind estimation algorithm however, naturally develops wind in Cartesian coordinates. Conversion to cylindrical coordinates can introduce a non-linearity if one of the wind components fluctuates around zero as both positive and negative wind components contribute to the magnitude estimate.

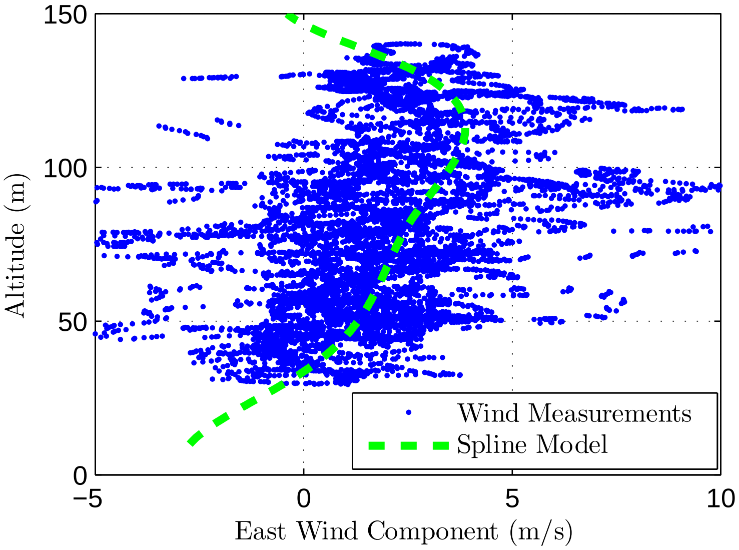


Figure 2. Spline model estimated for boundary layer winds from flight data recorded during a flight test. The aircraft was under manual control so the flight path was highly erratic, contributing to the noise seen in the wind measurements

III.C. Wind Shear in the Lee of a Ridge

The separated region in the lee side ridges offers very strong shear layers and has been the preferred location for pilots of radio controlled sailplanes to practice dynamic soaring. For initial testing of dynamic soaring, the strong shear, easy access, and relatively obstruction-free (compared to "albatross style" dynamic soaring) environment makes ridge shears attractive. The ridge shear layer is modeled using a sigmoid function to capture the strong shear remaining from the surface boundary layer which separated from the hill, as well as the mixing between the boundary layer and separated region which occurs after separation, the form adopted is specified in Equation 6, similar to logistic curves used in other dynamic soaring investigations.[11,13]

$$u(z) = \frac{u_{ref}}{1 + e^{\frac{14}{\Delta h}\left(\frac{\Delta h}{2} - (z - h_0)\right)}} \tag{6}$$

III.D. Combining Spline Models and *a priori* Selected Structures

For wind fields with very sharp curvatures, such as the thin and strong shears in the lee of ridges, even splines alone may not resolve the shear with sufficient accuracy to plan paths without overfitting, as seen in Figure 3(d). A combined approach is adopted, using the model-free approach described in Section III.A and a nonlinear state estimator to determine parameters of a pre-selected form for the shear. The parameter estimation approach has been demonstrated in simulation by Bencatel et al,[13] though here an unscented Kalman filter is used rather than a particle filter.

While the parameter estimation technique can identify the characteristics of the wind field, it is unsuited to determine the wind direction, which is equally critical to successful dynamic soaring. The sigmoid shear model is a scalar one, intended to be applied in the direction of the shear. Modeling the wind direction as a parameter of the model offers a solution, but a noisy estimate of the wind direction in the quiescent region of the wind profile will introduce large fluctuations in the wind direction that would be weighted equally with the more critical strong wind above the shear. A combined approach, using the spline model in addition to the parameter estimation filter offers a solution. Creating a block diagonal matrix of the spline coefficients:

$$\mathbf{M} = \begin{bmatrix} \mathbf{c}_n & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{c}_e & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{c}_d \end{bmatrix} \tag{7}$$

Where each component $\mathbf{c}_{(\cdot)}$ is a $(g + k - 1) \times 1$ column vector of the spline coefficients in the appropriate coordinate direction. A principal component analysis can then be applied to the matrix $\mathbf{M}$, and a 3×3 matrix is recovered. Taking the inverse tangent of the first two values of the first column (the horizontal coefficients of the first principal component) yields the effective wind direction of the spline model. In addition to providing a weighting based on wind magnitude, this approach also offers a means to restrict the wind estimation to a desired altitude range by including only those spline coefficients corresponding to knots in the region of interest.

This modeling approach allows the wind mapper to produce estimates of the characteristics needed by the trajectory planner to select the most appropriate trajectory. The final output of the wind mapper are the wind shear parameters

h_0, the bottom of the shear layer

h_f, the top of the shear layer

w_{max}, the shear magnitude

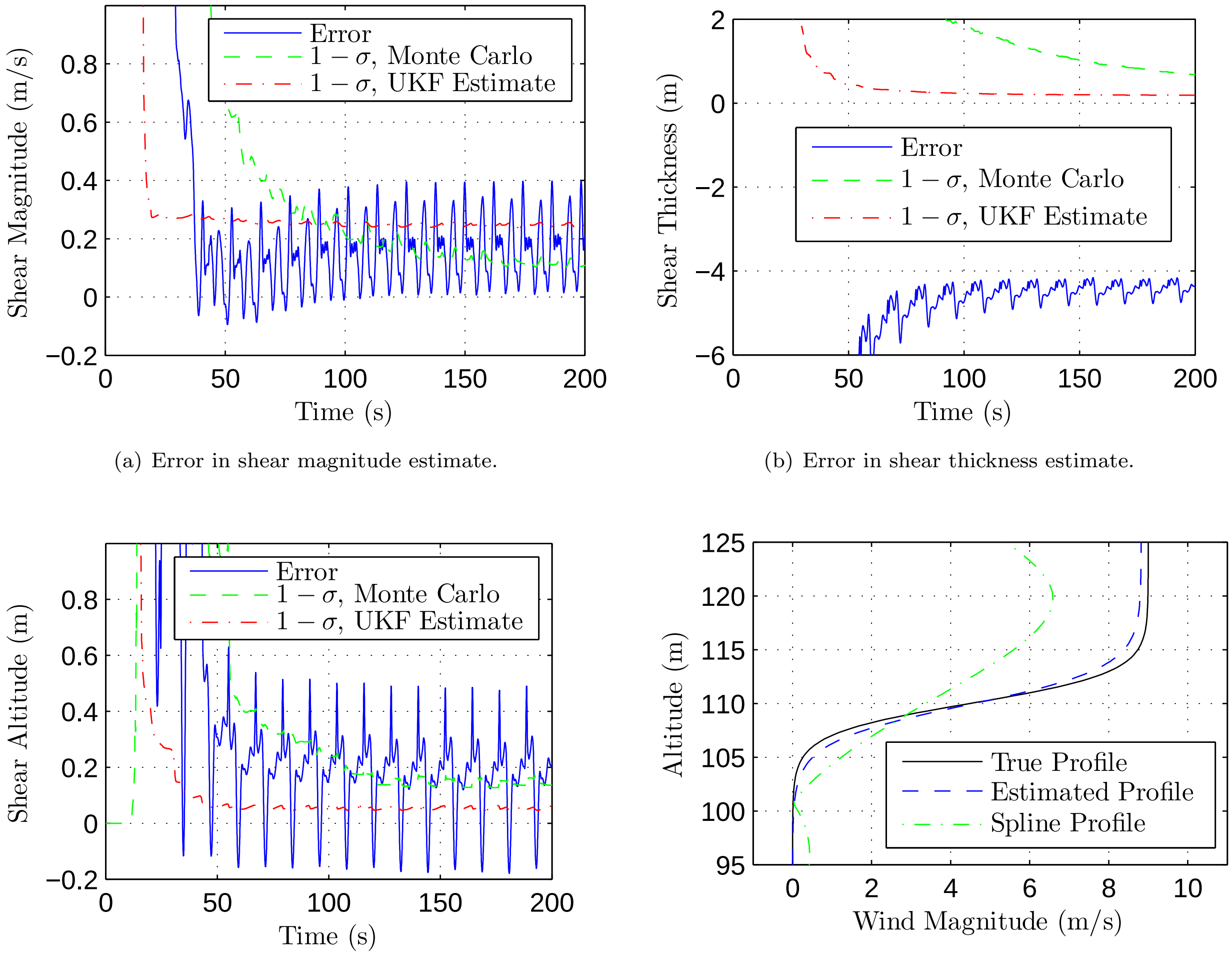
χ, the wind shear direction

These values are communicated to the trajectory planner and used to select the appropriate trajectory and align it with the wind, described in Section IV.

III.E. Assessing Wind Mapper Performance in Simulation

A batch simulation is implemented in MATLAB to assess the reliability of the wind mapper. The aircraft is simulated using the dynamic soaring equations of motion used in trajectory planning.[14] Since the simulation is only concerned with testing the wind mapper, the aircraft is provided with direct thrust control to ensure it can properly fly the DS trajectory. In the simulation the aircraft takes off, intercepts the shear and flies a pre-specified soaring cycle. A 500 run Monte-Carlo set, corrupted with a random level of Dryden turbulence[15] is used to assess the wind map in a variety of conditions for 5 and 20 meter thick shears.

Figure 3 shows that the wind map is able to estimate the magnitude and location of the wind shear to within 2%, but there is significant error in estimation of the thickness. This is potentially due to the effect



(a) Error in shear magnitude estimate.

(b) Error in shear thickness estimate.

(c) Error in shear position estimate.

(d) Estimated wind profiles.

Figure 3. Mean wind map results for 500 run Monte Carlo simulation with a shear thickness of 20 meters and magnitude of 9 m/s. Twelve runs were not included in the mean results as the filter diverged.

of the non-linear measurement model, or the weak signal in the extreme regions of the shear. Simulations with other shear conditions reveals the error in shear thickness to be relatively independent of true shear thickness, indicating that in relative terms the wind map should be expected to perform better for thicker shears.

IV. Trajectory Planning

The trajectory planner used in this investigation is developed from that used by Grenestedt and Spletzer,[14] which optimizes a vector of state variables:

$$\mathbf{X} = \begin{bmatrix} V(0) & V(\Delta t) & \cdots & V(T) \\ \psi(0) & \psi(\Delta t) & \cdots & \psi(T) \\ \gamma(0) & \gamma(\Delta t) & \cdots & \gamma(T) \\ \mu(0) & \mu(\Delta t) & \cdots & \mu(T) \\ h(0) & h(\Delta t) & \cdots & h(T) \\ x(0) & x(\Delta t) & \cdots & x(T) \\ y(0) & y(\Delta t) & \cdots & y(T) \\ C_L(0) & C_L(\Delta t) & \cdots & C_L(T) \end{bmatrix} \tag{8}$$

to minimize the objective function:

$$J = V_a^2(0) - V_a^2(T) \tag{9}$$

where V is the aircraft airspeed, ψ is the aircraft heading angle, γ is the aircraft pitch angle, μ is the aircraft bank angle, h is the aircraft altitide, x is the aircraft longitude, y is the aircraft latitude, C_L is the coefficient of Lift, Δt is the simulation time granularity, and T is the total time to fly the trajectory.

This choice of objective attempts to maximize the energy extracted during each cycle. The optimization constraints are constructed such that the beginning and ending altitudes are identical, so minimizing the function represents maximizing aircraft total energy. The function is left unbounded below, indicating that the aircraft should attempt to gain as much energy as possible during a cycle. Constraints are placed on the optimization which ensure that the path observes the aircraft equations of motion, periodicity of the trajectory, and keeps the trajectory within constraints determined by limitations of the autopilot and aircraft itself.

IV.A. Trajectory Constraints

Grenestedt and Spletzer discuss the constraints required for periodicity, energy extraction, and to enforce the equations of motion and gross aircraft limitations (C_L, velocity, and attitude limitations).[14] This work has been extended to restrict the trajectory to a path which can be flown repeatedly by a specific aircraft and control system. Constraints are constructed to more fully represent the capabilities of the Escale Fox sailplane and Piccolo SL autopilot system. The combination of large RC sailplanes and Piccolo autopilots has been used commonly in past autonomous soaring demonstrations, making this configuration a natural choice to use in this investigation. Aircraft and autopilot command limits, as well as experience gained in simulating dynamic soaring with this system warranted the inclusion of a number of additional constraints:

$$|\phi_{max}| <= 50^\circ$$

$$|\gamma_{max}| <= 30^\circ$$

$$|\dot{\phi}_{max}| <= 50^\circ/s$$

$$|\dot{\gamma}_{max}| <= 50^\circ/s$$

$$|\ddot{\phi}_{max}| <= 114^\circ/s^2$$

$$|\ddot{\gamma}_{max}| <= 114^\circ/s^2$$

$$\mathbf{r}(t_0) = \mathbf{r}(t_0 + T)$$

where ϕ and γ are respectively the bank and flight path angles and $\mathbf{r}$ is the 3-dimensional aircraft position vector in the fixed world frame.

This constraint set allows a path to be determined which is dynamically feasible and within the capabilities of aircraft and autopilot. Further, it ensures that the aircraft path remains stationary with respect to the ground so that successive paths can be flown in the same location (important for local phenomena such as ridge shears).

In addition to the imposition of additional constraints, the trajectory planner is modified to relax the assumption of constant vertical wind shear, β. The trajectory planner now evaluates the model wind shear equation at each point, computing the wind and shear value analytically. This allows trajectories to be planned in more complex wind fields as well as providing a logical integration with the wind modeling.

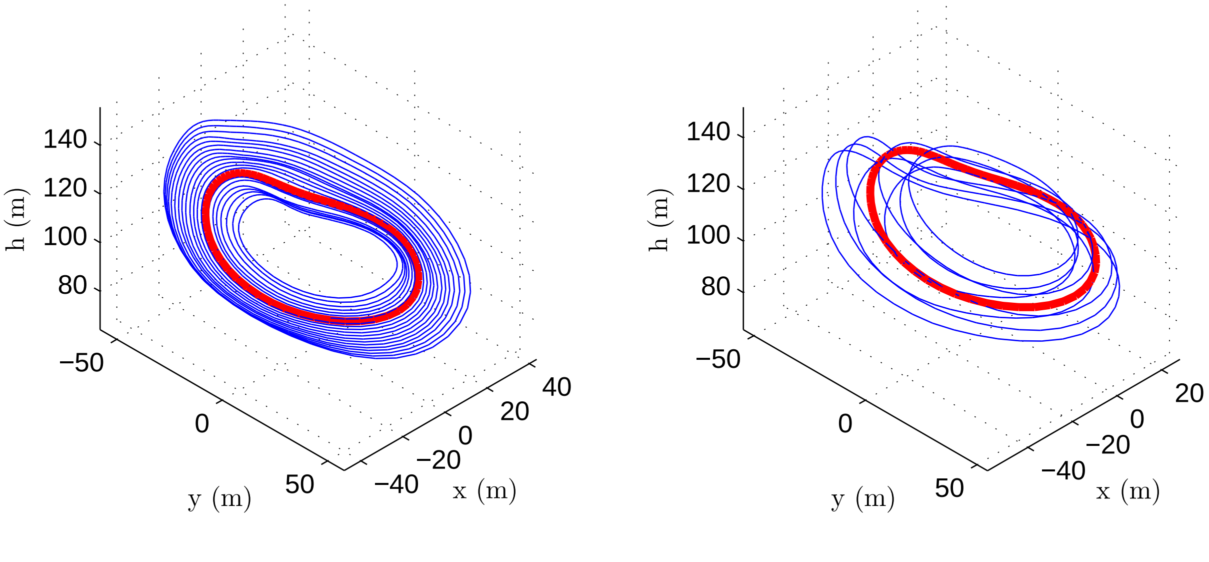
The trajectory planner outputs a series of aircraft states as in Equation 8, defining the path to be followed, and passed to the trajectory controller application.

IV.B. Trajectory Selection

If the aircraft is operating in an environment where the form of the shear can be anticipated, the trajectory planning problem can be simplified. Using the planner outlined in Section IV, locally optimal trajectories can be pre-computed *a priori* across the range of expected aircraft speeds and shear conditions. In effect, we are trading off computational complexity for storage as we would no longer need to compute the trajectories in real-time. This is motivated by the large size and high speed of conventional memory systems in comparison to the limited computational resources typically available on-board the aircraft.

In this vein, we constructed a database of locally optimal trajectories indexed by just three parameters: the aircraft airspeed at the lowest point of the trajectory v_0, the thickness of the shear layer Δh, and the free stream wind speed w_{max}. Since we could not cover a continuous range of wind conditions and aircraft air speeds, we generated trajectories for a fixed granularity and used a nearest neighbor interpolation scheme to select the most appropriate trajectory using feedback from the windmapper and aircraft telemetry. Once the trajectory was chosen, it was shifted to the world frame using the coordinate transformation inferred from the windmapper, and then reported to the DS controller. We should note that to prevent constant changes in the commanded trajectory, a new trajectory was selected only once per cycle.

Trajectory sets illustrating the effect of changing shear conditions and aircraft initial velocities are shown in Figure 4.



(a) Trajectory set for constant shear conditions.

(b) Trajectory set for constant initial velocity.

Figure 4. Figure 4(a) shows a set of trajectories for fixed $w_{max} = 9$ m/s and $h_0 = 10$ m with varying entry velocities ranging from 15 m/s to 35 m/s in steps of 1 m/s. Figure 4(b) shows a set of trajectories for fixed $v_0 = 22$ m/s, with w_{max} varying between 9 m/s and 12 m/s, and h_0 varying between 5 m and 20 m. The bold trajectory highlighted in red is the same in both figures ($w_{max} = 9$ m/s, $h_0 = 10$ m, $v_0 = 22$m/s).

V. Path Following Controller

Controlling a dynamic soaring aircraft requires tracking more than just position – the planner generates a trajectory of both geographic position and aircraft state (C_L, ϕ, γ). As dynamic soaring force is dependent on the entire aircraft state during a shear encounter, the controller must track a trajectory not just in geographic space, but in the entire aircraft state-space in order to provide adequate performance. Wind gusts, uncertainty in the shear environment, and aircraft modeling error renders open-loop scheduling of the nominal controls an ineffective method. The rapid maneuvering required for dynamic soaring makes pure feedback problematic as well, requiring impractically high gain in order to provide a steering command with small track errors.

This challenge is managed by use of a feedforward-feedback control architecture which linearizes the trajectory about a point through a feedforward command from the nominal flight path and uses a feedback controller to fly out error in the path. The architecture for such a system is depicted in Figure 5.

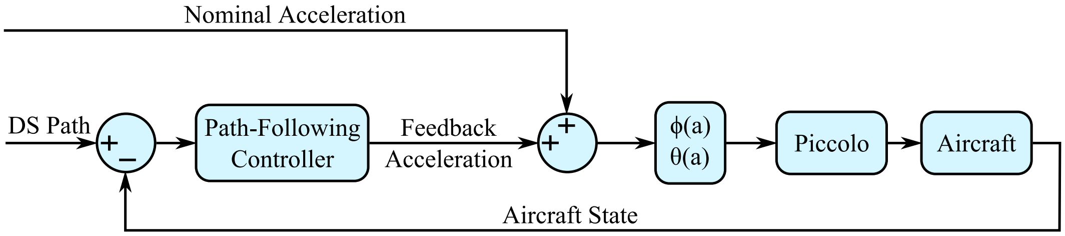


Figure 5. Feedforward-feedback control structure for dynamic soaring

Since the trajectory planner optimizes over a series of aircraft states, the path is readily available as a series of "waypoints" in the aircraft state-space. At each of these points the acceleration required for an aircraft flying the path perfectly is easily computed. The feedforward signal is then determined by returning the nominal acceleration at the nearest waypoint.

The feedback signal is computed based on the error between aircraft location and the desired path. The feedback control uses the non-linear path following controller described by Park et al[16] and depicted in Figure 6.

$$\mathbf{a}_{cmd} = \frac{2}{|\mathbf{L}|^2} (\mathbf{V} \times \mathbf{L}) \times \mathbf{V} \tag{10}$$

Where $\mathbf{V}$ is the vehicle velocity vector and $\mathbf{L}$ is a vector from the aircraft to a desired "look ahead" point on the path. The controller commands an acceleration which drives the aircraft smoothly back to the desired path. Choice of acceleration as the commanded value is somewhat unconventional, but offers the advantage of being readily measurable on-board the aircraft (as opposed to flight path angle, for example) as well as natural integration with the feedback commands.

Determining the location of the goal point can become problematic for tightly curved trajectories, if the look ahead distance is larger than the diameter of a circular path then a goal point can not be defined at all. In the control system described here, feedback is required only to fly out errors in trajectory. The nominal flight path curvature is provided via feedforward, so an alternate definition of the goal point is chosen.[9] A tangent line is extended from the point on the path closest to the aircraft, and the goal point located on this line at the look ahead distance from the aircraft.

Viewing the feedforward as a linearization, if the aircraft is not on the trajectory then the nominal feedforward command will be in error. For example, if the aircraft is "outside" the trajectory (i.e. further from the trajectory centroid than its nominal position), the acceleration command will be greater than required to fly parallel to the path at that radius and it will turn toward the circle. While this seems at first glance to be desirable behavior, this violates the feedforward-feedback architecture structure which requires the feedforward only to linearize the aircraft about the path, providing a signal to continue parallel to the nominal path. A scale factor is thus applied to the feedforward:

$$\mathbf{a}_{feedforward} = \mathbf{a}_{nominal} \frac{\|\mathbf{r}_{nominal} - \mathbf{r}_{centroid}\|}{\|\mathbf{r}_{aircraft} - \mathbf{r}_{centroid}\|} \tag{11}$$

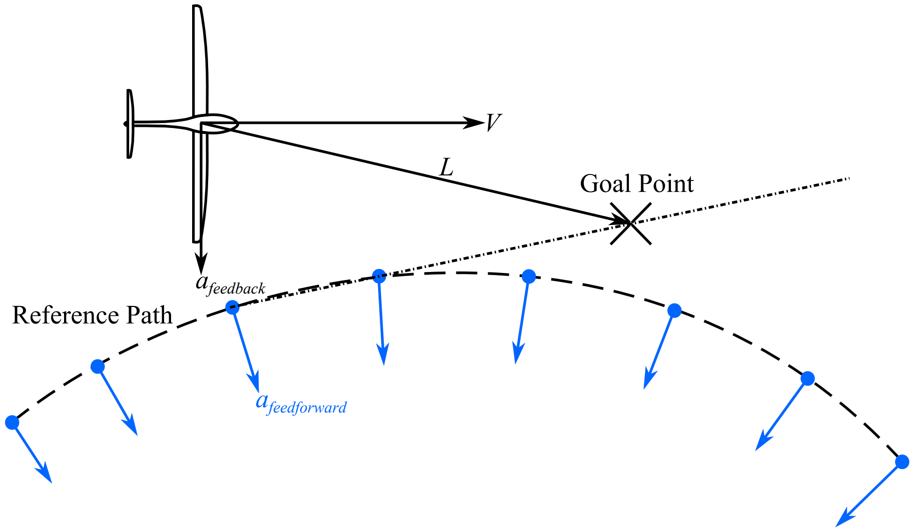


Figure 6. Geometry of the nonlinear guidance law and feedforward linearization signal.

Since both controllers yield acceleration commands in inertial axes, they can be easily combined by simply summing into a total acceleration command. While making command composition simple, this still presents a difficulty as the acceleration must be converted to Euler angle commands to be accepted by the Piccolo.

To make acceleration actionable by the Piccolo, first the commanded accelerations are rotated by the heading angle into a coordinate frame with x in the direction of flight. Making the assumption that $\dot{\gamma} = \dot{\theta}$, the desired pitch rate is determined through kinematics:

$$\dot{\theta} = \frac{-a_z}{V_a} \tag{12}$$

Observing that lateral acceleration is controlled by tilting the lift vector, the required lateral acceleration can be expressed:

$$a_y = (a_z + g)\tan\phi \tag{13}$$

Which can easily be rearranged to give a bank angle which will provide the proper y to z acceleration ratio.

$$\phi = \tan^{-1}(a_y, a_z + g) \tag{14}$$

The achieved y acceleration will then be as commanded provided that the required z axis acceleration is achieved. Referring to Equation 12, the required z acceleration can be achieved through $\dot{\theta}$. Unfortunately, $\dot{\theta}$ cannot be directly commanded to the Piccolo. The autopilot will however, attempt to achieve a first-order response to pitch angle commands. The time constant is known as it is a specified autopilot configuration parameter, so a pitch step command can be constructed to achieve the desired pitch rate.

$$\theta_{cmd} = \frac{1}{T_\theta}\dot{\theta}_{cmd} + \theta \tag{15}$$

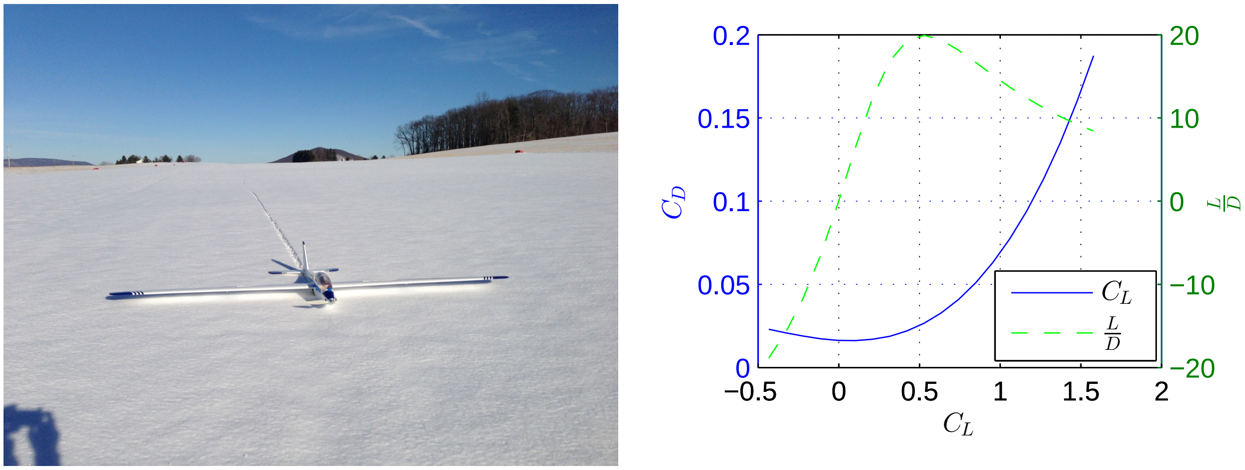
As a practical consideration, it is important to note that there is no guarantee that the solution provided will be a feasible one for the autopilot to fly – for example, a sufficiently strong downward acceleration may require the aircraft to roll inverted, however the Piccolo autopilot will not allow roll angles greater than 90°. Such limitations makes it critical that the trajectory generated is not only feasible for the aircraft to fly, but also not at the limits of autopilot capability in order to ensure sufficient maneuvering margin exists to fly out trajectory error.

Another concern in the implementation of this path following controller is selection of the look-ahead distance, L. This serves as a convergence tuning parameter for the controller – smaller values of L lead to more rapid correction of flight path error, but can also de-stabilize the controller. In simulation, the large variations in speed experienced during a dynamic soaring loop made selecting a look-ahead distance difficult as no single value provided adequate performance. A short look ahead causes the aircraft to overshoot descending portions of the trajectory and destabilize at the upper limits of the trajectory, while a long look ahead cuts off the upper portion and prevents fine trajectory control. A time based look-ahead proved more successful, a value of 4 seconds is used. In simulation this value proved to be a good balance of the competing requirements.

VI. Hardware in the Loop Simulations

The CloudCap Piccolo simulation environment is used to test components of the dynamic soaring system. CloudCap provides a 6DoF vehicle simulation, which generates sensor measurements which can be fed to the Piccolo autopilot (either software simulator or hardware autopilot). Initially, it was intended to start with the software-simulated Piccolo autopilot, but excessive latency in the software autopilot necessitated use of the hardware simulation which interfaces with the Piccolo autopilot hardware itself. Wind shear is implemented in the simulator by discretizing the wind field by altitude. To accurately resolve shear features, wind is specified at vertical positions spaced between one and ten meters apart throughout the shear region. The Piccolo simulator interpolates wind values between layers.

VI.A. Aircraft Model



(a) Fox aircraft after test flight.

(b) Predicted drag polar and L/D.

Figure 7. Fox aircraft model characteristics

The aircraft used in this simulation series is an Escale Fox radio controlled sailplane. The aircraft has a 2.8 meter wingspan and an empty weight of 4.15 kg, though the trajectory planner and simulation allow for ballasting the aircraft to achieve higher wingloading. The aircraft model is created using AVL and a component-based drag buildup. The performance of the aircraft is inferior to typical dynamic soaring RC aircraft, however the flight profiles intended in this investigation are also far less demanding as they are focused solely on sustaining flight and not on all-out speed.

VI.B. Dynamic Soaring With *a priori* Windfield Knowledge

The trajectory following controller was first tested in HiL simulation given *a priori* knowledge of a wind field in the lee of a ridge. This had two objectives: first, to assess the capability of the trajectory following controller in following typical DS trajectories; second, to assess methods for initializing DS from steady state flight. Initialization of DS can be non-trivial: there is no point on the cycle when the aircraft is in a steady-state condition. However, entry steady orbit is possible by providing the aircraft with sufficient excess initial energy: this excess initial energy is used to "fly out" the initial trajectory error.

Figure 8 shows the flight path for entry into a planned DS trajectory and the sixty seconds following entry. After a few cycles the aircraft converges to a steady dynamic soaring trajectory that does not follow the planned path exactly, but that is sustainable (i.e. the resulting flight path remains energy neutral).

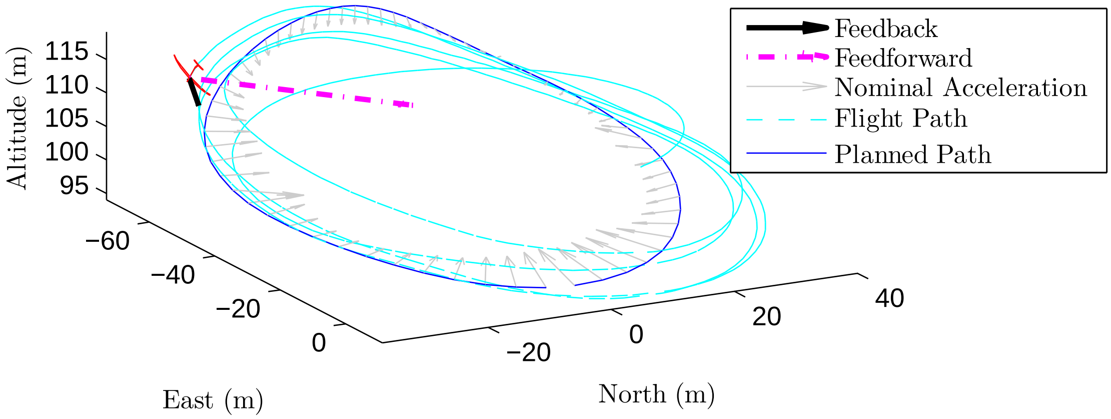


Figure 8. Flight path for 60 seconds after ds initiation.

Figure 9(a) illustrates the aircraft total energy converging toward the trajectory plan over the first four dynamic soaring cycles. Note the discontinuities in planned total energy: this reflects the replanning that occurs at the bottom of each DS cycle. Each re-plan occurs to in response to the actual aircraft state at the bottom of the cycle: Figure 4(a) shows trajectory variations due to changes in entry speed. The remainder of Figure 9 depicts the aircraft state during the fourth cycle after initialization.

Simulations of dynamic soaring in a HiL environment make clear an intuitive principle – optimal cyclic dynamic soaring paths are not realistic operational paths. In practice, attempts to simulate dynamic soaring with energy neutral trajectories slowly lose energy until the aircraft falls below the shear region and can no longer track the desired trajectory. This is due to the presence of trajectory errors: correcting an error in the flight path requires an expenditure of energy which leaves the aircraft with insufficient energy to continue dynamic soaring. This problem has spurred recent interest in planning robust dynamic soaring trajectories.[7] Here the aircraft is given a surplus energy budget for maneuvering as it flied initially energy positive paths. As it gains speed it eventually reaches a path that is optimally energy positive but becomes neutral through maneuvering losses. While the resulting paths are not strictly periodic and introduce error into the flight path, the system is capable of dynamic soaring.

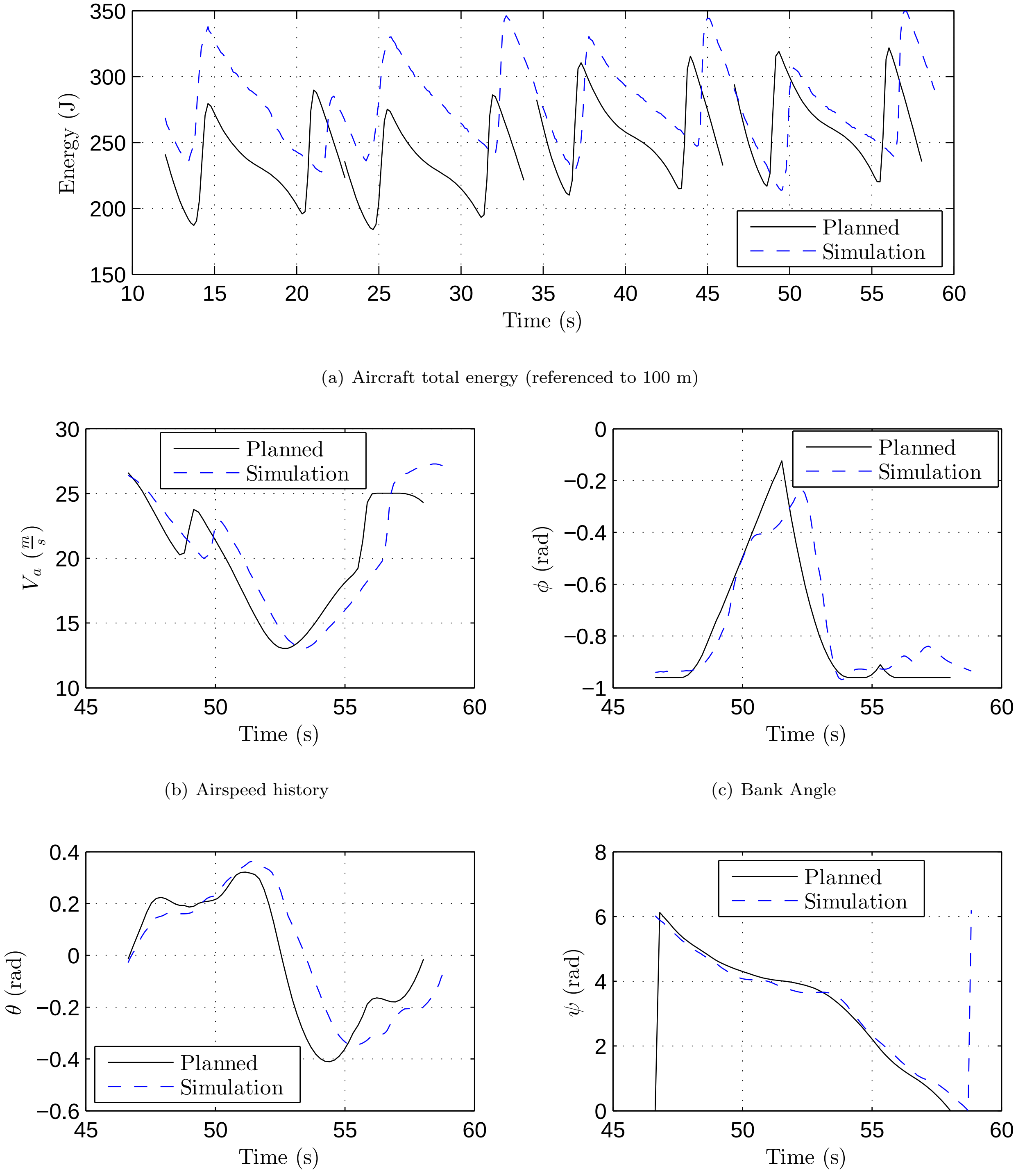
VII. Closed Loop Dynamic Soaring

As the CloudCap HiL simulation is used for all initial simulations, moving to a full closed loop simulation requires simply “plugging” the Piccolo into the rest of the aircraft, connecting an antenna, and routing communication through the ground station. For this investigation the Piccolo interface and dynamic soaring MATLAB applications are run on a desktop computer but it communicates with the Piccolo using the serial link available to a computer on the vehicle. In this configuration, the entire system required for autonomous dynamic soaring can be tested, allowing the system to be verified as thoroughly as possible on the ground. To run fully self-contained all that would be required is to run the applications on a small form factor single-board computer within the aircraft.

Before simulating the flights we generated a set of dynamic soaring trajecories as in Section IV to cover the combination of simulation conditions as shown in Table 1. For each of the 9 boundary layer conditions, trajectories with v_0 ranging from 15 m/s to 35 m/s in steps of 0.1 m/s were generated, for a total trajectory data base of 1809 trajectories.

For each run, the aircraft is started at an altitude of 0 m, with the bottom of the simulated shear layer located at 100 m. The wind map is initialized with a shear location of 110 m and a thickness of 30 m, simulating a condition where forecast conditions, a pilot balloon, or prior flights have given a rough indication of the shear parameters.

The aircraft is launched and flies to a circling waypoint near the desired DS start position. It flies alternating orbits above and below the suspected shear region, allowing the wind mapper to converge. Wind map convergence is assessed through the trace of the parameter estimator covariance matrix; simulations



(a) Aircraft total energy (referenced to 100 m)

(b) Airspeed history

(c) Bank Angle

(d) Pitch Angle

(e) Heading Angle

Figure 9. Flight path and state time histories for Fox dynamic soaring in a lee of ridge shear with *a priori* known wind field. Shear magnitude is 6 m/s and thickness is 5 m. The aircraft gains energy on an optimally energy-negative trajectory by flying the ascending and descending portions of the trajectory more aggressively than planned as it flies out remaining error from trajectory initiation. The planned trajectories were limited to ±30°, but the controller may fly closer to the optimal 45° angle to fly out error.

showed that reaching a value of 0.2 consistently indicates sufficient convergence for soaring. Once the wind map has converged, the aircraft returns to the original orbit and attempts to reach a target airspeed. A trajectory is selected according to the procedure in Section IV, using parameters w_{max} and h_0 as reported by the wind mapper, and the current airspeed reported by the Piccolo. With a trajectory selected, the aircraft is commanded to an orbit tangent to the top of the DS trajectory. When the peak of the trajectory is the closest waypoint to the aircraft, the engine is cut and the dynamic soaring controller assumes command of the aircraft. Simulations are run for 30 minutes or until the aircraft fails to continue dynamic soaring.

		Wind Magnitude		
		6 m/s	9 m/s	12 m/s
Thickness	5 m	0,10 m/s	0,10 m/s	0,10 m/s
	10 m	0,10 m/s	0,10 m/s	0,10 m/s
	20 m	0,10 m/s	0,10 m/s	0,10 m/s

Table 1. Wind and turbulence simulation conditions used to test the ds controller, the center fields indicate the 20 foot wind parameter used to generate the turbulence magnitude. To vary turbulence independently of the shear strength, the 20 foot wind is treated solely as a turbulence parameter, not as a wind speed.

VII.A. Simulations in Smooth Air

To test the closed loop dynamic soaring system in the absence of external disturbances, a set of simulations are run without turbulence. During this initial set of simulations the wind map convergence criterion was developed and look-ahead time tuned for the path following controller. Table 2 illustrates the simulation outcomes for the shear conditions tested.

		Wind Magnitude		
		6 m/s	9 m/s	12 m/s
Thickness	5 m	30 min	30 min	Failed to DS
	10 m	30 min	30 min	30 cycles
	20 m	30 min*	20 min	Failed to DS

Table 2. Dynamic soaring system performance with no turbulence. The 12 m/s condition typically failed after the Piccolo attitude filter diverged. (*–the wind map did not reach the convergence criterion and dynamic soaring was manually initiated)

During this sequence of simulations the wind mapper was discovered to under-estimate shear thickness. This was a major contributor to the diminished stability of the system in 20 meter shears, often the windmap estimated shear thickness to be approximately 15 m. Since trajectories were only available for the conditions tested, trajectory changes would be dramatic as the planner switched between trajectories for 10 m and 20 m thick shears. Improving the resolution of trajectories available and the performance of the wind mapper would likely improve stability of the system.

Another factor discovered which impacted performance is the wind direction estimation. Since the Piccolo heading is driven by the GPS ground track, the heading estimated does not converge until several seconds into flight. If the wind mapper is started before launching the aircraft, the Piccolo attitude filter convergence pollutes the spline model and thus the wind direction estimate. Eliminating this effect is the determining factor for the wind map convergence criteria. In finding an appropriate value for the criteria, wind direction errors of greater than 0.2 radians were observed to significantly increase the likelihood of failure.

The GPS driven heading contributes also to the failure to soar in 12 m/s shears. In very strong shears the wind velocity is near or greater than the minimum airspeed during a cycle, without an independent source of heading information the atttiude filter is unable to track the aircraft state as GPS speed and heading shift dramatically upon crossing the shear. The dynamics of shear crossing also disturb the attitude filter, seen most significantly in the 5 m, 12 m/s condition where the attitude filter diverged almost immediately.

VII.B. Light Turbulence

To assess the performance of the dynamic soaring system in a more representative environment, it was tested in a Dryden turbulence field. The MIL-STD-1797A turbulence definitions are used, which defines a range of 20 foot wind magnitude for each turbulence level.[15,17] Here the high end of the reference wind range is used, defining light turbulence with a 20-foot wind magnitude of 10 knots, 5.1 m/s. The standard deviation of the vertical wind is defined as $\sigma_w = 0.1U_{20}$ and horizontal components are defined as functions of σ_w and height. A reference altitude of 30 m is used here for computing turbulence length scales, turbulence length and intensity is shown in Table 3.

	Magnitude (m/s)	Length (m)
u	0.88	153
v	0.88	153
w	0.51	30

Table 3. Dryden parameters for light turbulence.

Simulations in turbulence revealed the need to plan trajectories with an energy reserve. When soaring without turbulence, the aircraft could fly a trajectory near enough to nominal that an optimal trajectory could be flown, provided that it was not strictly periodic (i.e. it nominally had higher final than initial velocity). Eventually the trajectory would stabilize as the aircraft would fly an optimally energy gaining trajectory that became energy neutral after maneuvering losses. With turbulence added, the additional maneuvering and suboptimal C_L from disturbances requires more energy. Planning for a trajectory with a 5% lower than actual airspeed provides energy that significantly improves the reliability of the soaring system in turbulence. Table 4 shows that even with this added margin, the turbulent condition favors more energetic trajectories; the 9 m/s, 5 m shear is the only condition to support extended dynamic soaring. Once again, the 12 m/s shear condition destabilized the Piccolo attitude filter to the extent that it was not possible to soar, this was aggravated by the turbulent conditions.

		Wind Magnitude		
		6 m/s	9 m/s	12 m/s
Thickness	5 m	25 min	30 min	Failed to DS
	10 m	7 min	5 min	Failed to DS
	20 m	Could Not Maintain	Could Not Maintain	Failed to DS

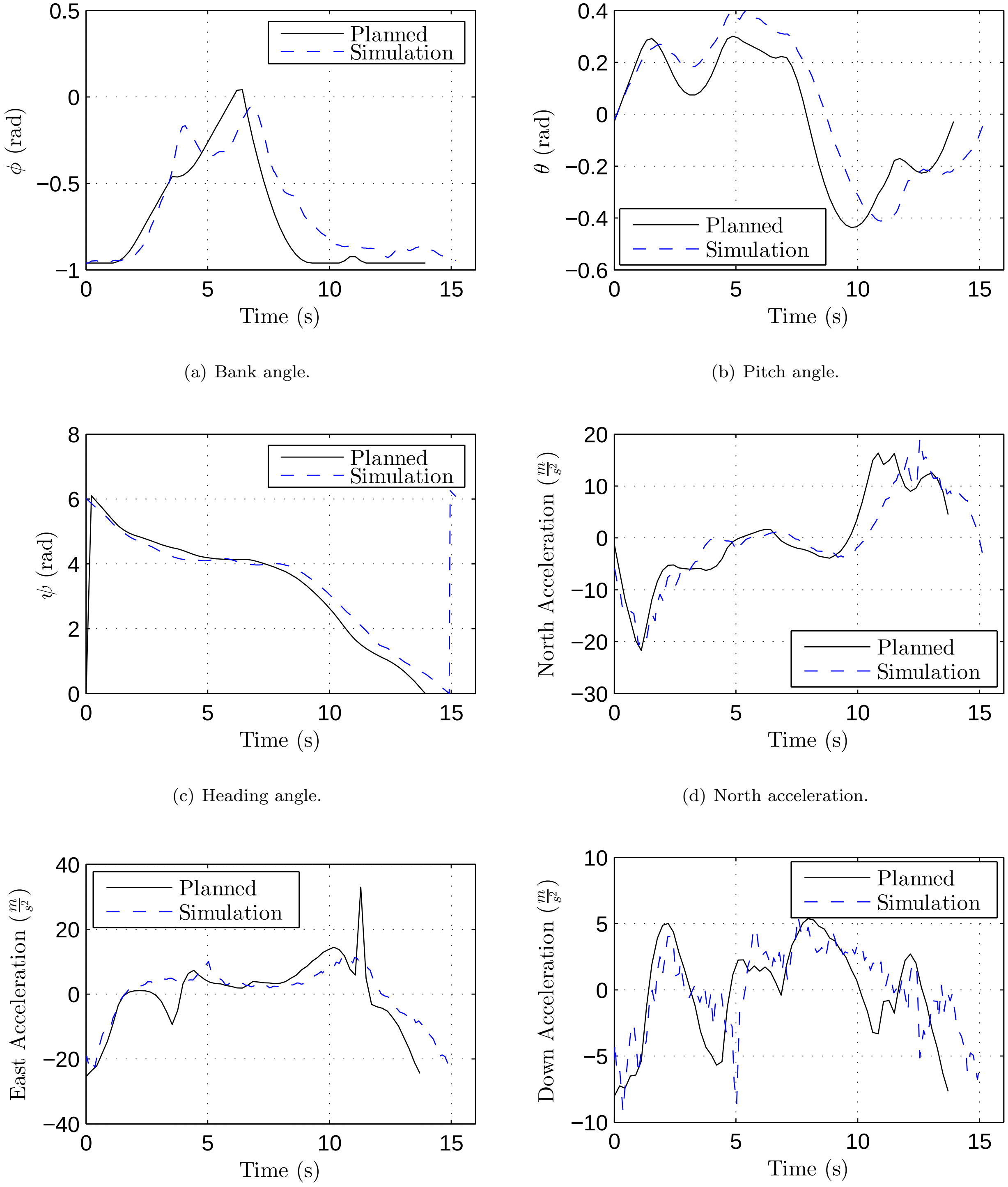
Table 4. Dynamic soaring system performance in light turbulence. The 12 m/s condition typically failed after the Piccolo attitude filter diverged. In the 20 meter thick shear cases, the combination of weak shear and energy lost to turbulence rendered the aircraft unable to fly an energy gaining cycle and dynamic soaring was terminated after just a few cycles.

To achieve a direct comparison between dynamic soaring with and without turbulence, a simulation is initialized without turbulence and after five minutes of dynamic soaring, the turbulence is turned on. The path error and changes in energy state between cycles are used as measures the effect of the disturbance. For an aircraft consistently soaring in a shear with $u_{ref} = 10$ m/s and $\Delta_z = 5$ m, light turbulence increases the RMS path error approximately 10%. The consistency between cycles is also impacted, without turbulence the RMS change in energy between paths updates (subsequent cycles) is 19.5 J, after turbulence is turned on it jumps to 36.3 J, a factor of almost 1.9. The larger disturbances require greater control action to follow the intended path with two effects. First, flying out error decreases the amount of energy harvested. Second, if the path error is sufficient it can destabilize the path following system as the required control action increases. The consistency measures are shown in Figure 12.

VII.C. Wind Mapper Performance

Performance of the wind mapper in simulation reveals that it works as intended but has deficiencies that prove challenging to continuous dynamic soaring. The underestimated wind shear thickness and magnitude is the most significant and obvious challenge. Several factors contribute to this error: the imperfect aircraft aerodynamic model used in wind computations and weak signal in the shear near its limits.

To determine the angle of attack and sideslip to rotate airspeed to inertial axes, a simplified aerodynamic model is used which neglects the elevator and pitch rate contributions to C_L. During dynamic soaring, both experience significant perturbations from their steady flight values, neglecting these factors could be introducing an error in the wind speed computation. Analysis of these factors is complicated however. The



(a) Bank angle.

(b) Pitch angle.

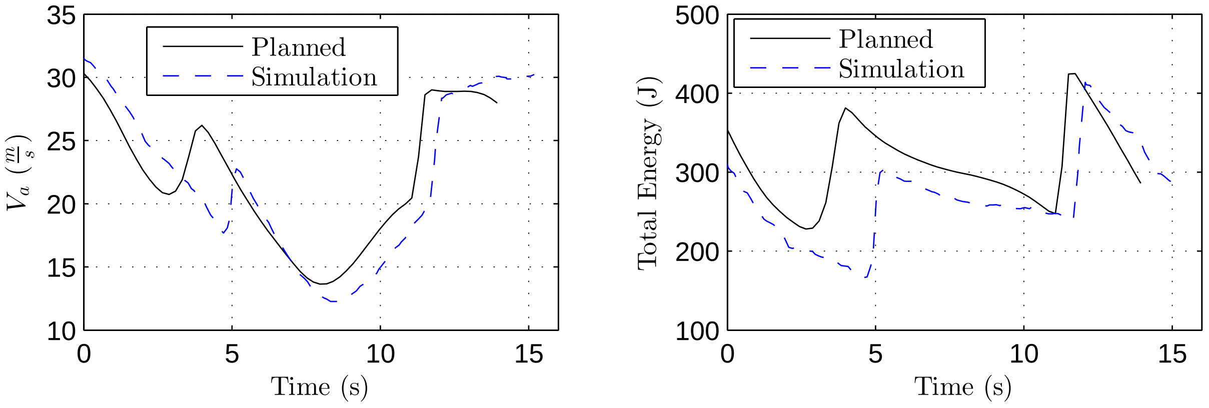
(c) Heading angle.

(d) North acceleration.

(e) East acceleration.

(f) East acceleration.

Figure 10. Nominal and actual state history for one dynamic soaring cycle. Cycle is planned to start at 30.3 m/s in shear with $\Delta h = 5$ m, $u_{ref} = 9$ m/s and light turbulence. The effect of the turbulence on vertical acceleration is most pronounced, indicative of the C_L disturbances that reduce the energy gained in a cycle.



(a) Airspeed.

(b) Total Energy.

Figure 11. Energy measures for one dynamic soaring cycle. Cycle is the same as depicted in Figure 10. The extra energy provided for turbulence and maneuvering losses is visible in the airspeed history. The energy deficit at the cycle start is caused by position error placing the aircraft below the nominal path.

effect of angle of attack error changes sense between ascending and descending legs of the DS cycle. On the ascending leg underestimating the angle of attack will yield a wind measurement smaller than the true wind, while on the descending leg the opposite is true. The pitch rate and elevator effects are also opposite in sense; neglecting pitch rate overestimates angle of attack while neglecting elevator underestimates it.

Compounding the model error is the fact that at the shear edges, the signal available in the wind profile is weak. For instance, 60% of the wind change occurs in 20% of the shear thickness, and 97% occurs within 60%. Identifying the extent of the shear is thus very difficult, especially in the presence of any noise or modeling error. Wind map performance is illustrated in Figure 13.

VIII. Conclusion

This paper presents further progress on the development and integration of the hardware and software components needed to demonstrate practical autonomous dynamic soaring. Building on previous work establishing the basics of planning dynamic soaring trajectories and on estimating wind *in situ* on an aircraft, this paper "closes the loop," estimating the wind field, then planning and flying an appropriate dynamic soaring trajectory.

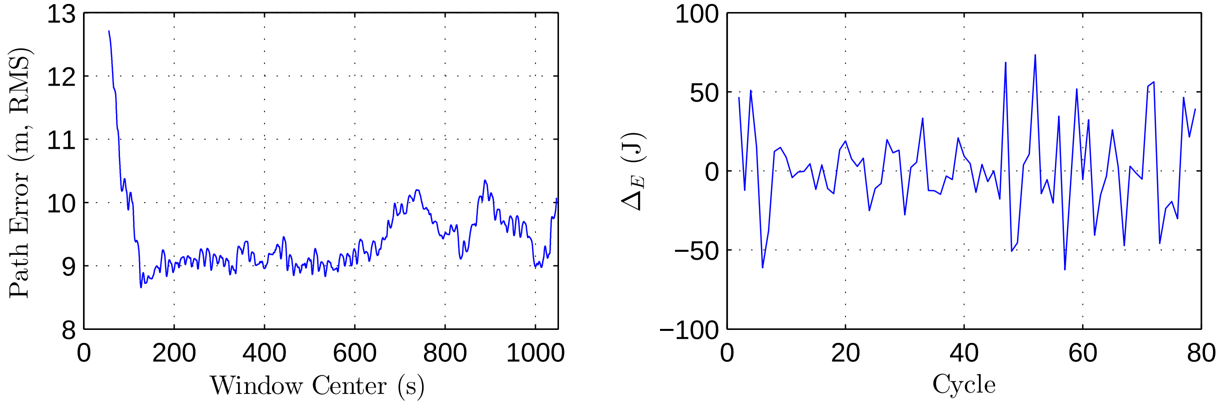
The wind field is estimated from measurements available on board the aircraft and modeled through a combination of basis splines and estimation of the parameters of a model wind structure, allowing complex wind fields to be estimated and represented compactly. The ability the combined method to adequately model wind fields is shown for the lee of ridge shear environment. The utility of the models is established through the capability to plan energy harvesting trajectories using the wind map.

Linearization of the DS trajectory via feedforward acceleration with a feedback law providing error compensation is shown to be an effective method for tracking dynamic soaring trajectories. Use of the Piccolo SL autopilot system as a source of aircraft state information and inner loop controller providing information and actuation for an outer loop trajectory planner is demonstrated using hardware in the loop simulation. Generation of Euler angle commands via aircraft kinematics provides an adequate means of translating commanded accelerations into signals actionable by the Piccolo.

Autonomous dynamic soaring using currently available hardware, trajectory planners, and control policies appears to be viable for strong shear layers such as those found in the lee of ridges. Further development of autonomous dynamic soaring autopilots and state estimators tailored to the rapid maneuvers required for dynamic soaring could improve performance.

Acknowledgements

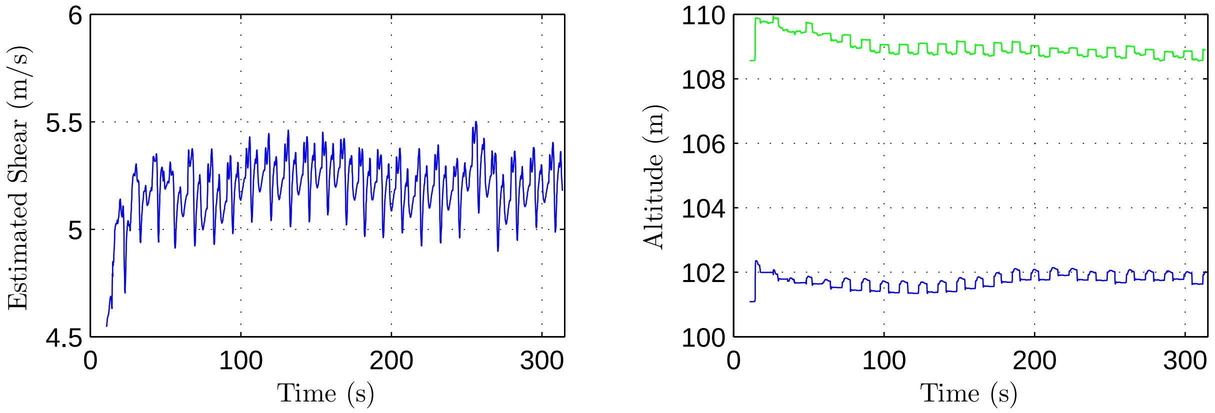
This research was funded by the National Science Foundation under grants IIS-1065024 and IIS-1065202.



(a) RMS path error computed with a 1000 point sliding window.

(b) Change in energy state between subsequent cycles.

Figure 12. Flight path consistency measures, RMS error and energy change between adjacent cycles. Error is computed as the minimum distance between the aircraft location and a point on the flight path, RMS values are computed with a 1000 point sliding window. Simulation is initialized with $\Delta_z = 10$ m, $u_{ref} = 9$ m/s with no turbulence. At approximately $t = 600$ seconds, light turbulence is turned on.



(a) Shear Magnitude

(b) Shear Top and Bottom Location

Figure 13. Time history of wind map parameters for aircraft dynamic soaring in shear with $\Delta_z = 10$ m, $u_{ref} = 6$ m/s, and shear bottom located at $z = 100$ m. Occasional spikes can be seen that appear to originate in the inter-process communication. Data starts when the dynamic soaring initialization command is issued and does not show wind map convergence prior to start.

References

[1] Rayleigh, J. W. S., "The Soaring of Birds," *Nature*, Vol. 27, 1883, pp. 534–535.

[2] Rayleigh, J. W. S., "The Sailing Flight of the Albatross," *Nature*, Vol. 40, 1889, pp. 34.

[3] Sachs, G., "Minimum shear wind strength required for dynamic soaring of albatrosses," *Ibis*, Vol. 147, 2005, pp. 1–10.

[4] Deittert, M., Richards, A., Toomer, C. A., and Pipe, A., "Engineless Unmanned Aerial Vehicle Propulsion by Dynamic Soaring," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 5, 2009, pp. 1446–1457.

[5] Gordon, R. J., *Optimal Dynamic Soaring for Full Size Sailplanes*, Master's thesis, Air Force Institute of Technology, Wright-Patterson Air Force Base, Sept. 2006.

[6] Sachs, G. and da Costa, O., "Dynamic Soaring in Altitude Region below Jet Streams," *AIAA Guidance, Navigation, and Control Conference*, Aug. 2006.

[7] Flanzer, T. C., Bower, G. C., and Kroo, I. M., "Robust Trajectory Optimization for Dynamic Soaring," *AIAA Guidance, Navigation, and Control Conference*, Aug. 2012.

[8] Langelaan, J. W., Spletzer, J., Montella, C., and Grenestedt, J., "Wind field estimation for autonomous dynamic soaring," *IEEE International Conference on Robotics and Automation*, May 2012.

[9] Bird, J. J., *Wind Estimation and Closed-Loop Control of a Soaring Vehicle*, Master's thesis, The Pennsylvania State University, 2013.

[10] Langelaan, J. W., Alley, N., and Neidhoefer, J., “Wind Field Estimation for Small Unmanned Aerial Vehicles,” *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 4, 2011, pp. 1016–1030.

[11] Sachs, G. and da Costa, O., “Optimization of Dynamic Soaring at Ridges,” *AIAA Atmospheric Flight Mechanics Conference*, 2003.

[12] Diercx, P., *Curve and Surface Fitting with Splines*, Oxford University Press, 1993.

[13] Bencatel, R., “Shear Wind Estimation,” *AIAA Guidance, Navigation, and Control Conference*, 2011.

[14] Grenestedt, J. L. and Spletzer, J. R., “Towards Perpetual Flight of a Gliding Unmanned Aerial Vehicle in the Jet Stream,” *IEEE Conference on Decision and Control*, Dec. 2010.

[15] Standard, M., “Flying Qualities of Piloted Aircraft,” Tech. rep., Department of Defence, Washington, DC, 1990.

[16] Park, S., “Autonomous Aerobatic Flight by Three-Dimensional Path-Following with Relaxed Roll Constraint,” *AIAA Guidance, Navigation, and Control Conference*, 2011.

[17] Moorhouse, D. and Woodcock, R., “Background Information and User Guide for MIL-F-8785C, Military Specification-Flying Qualities of Piloted Airplanes,” Tech. rep., Air Force Wright Aeronautical Laboratories, 1982.