

Experiments in Fixed-Wing UAV Perching

Rick Cory* and Russ Tedrake†

MIT Computer Science and Artificial Intelligence Lab, Cambridge, MA, 02139, USA

High-precision maneuvers at high angles-of-attack are not properly addressed by even the most advanced aircraft control systems. Here we present our control design procedure and indoor experimental results with a small fixed-wing autonomous glider which is capable of executing an aggressive high angle-of-attack maneuver in order to land on a perch. We first acquire a surprisingly accurate aircraft model through unsteady flight regimes from real kinematic flight data taken in a motion-capture arena. This model is then used in a numerical nonlinear (approximate) optimal control procedure which designs a feedback control policy for the elevator deflection. Finally, we report our experimental data demonstrating that this simple glider can exploit pressure drag to achieve a high-speed perching maneuver.

Nomenclature

x	Horizontal position, m
z	Vertical position, m
θ	Pitch angle of fuselage and wing, in the lab frame, radians
ψ	Elevator deflection angle, in the fuselage frame, relative to trim, radians
\mathbf{x}	Full state vector, $\mathbf{x} = [x, z, \theta, \psi, \dot{x}, \dot{z}, \dot{\theta}, \dot{\psi}]^T$,
u	Elevator command, $\in [-1, 1]$
α	Angle-of-attack of the wing
F	Force, N
M	Moment, Nm
w	Coefficient model parameters
v	Coefficient model nonlinear feature
c	Mean aerodynamic chord
π	Feedback policy
J	Cost-to-go function
\mathbf{Q}	Symmetric positive-definite quadratic penalty matrix
<i>Subscript</i>	
i	Variable number
L	Lift
D	Drag
<i>Superscript</i>	
d	Desired value
$*$	Optimal (as in π^* or J^*)

I. Introduction

Birds routinely execute aerial maneuvers that are still well beyond the capabilities of our best aircraft control systems. During a perching maneuver, birds often exceed 90 degrees in angle-of-attack, exploiting both viscous and pressure drag for rapid deceleration. Despite the complexity of the fluid dynamics in these unsteady regimes, the inevitable loss of control authority on the wings as the velocity of the bird reaches zero,

*Ph.D. Candidate, Electrical Engineering and Computer Science, rcory@csail.mit.edu.

†Assistant Professor, Electrical Engineering and Computer Science, russt@mit.edu, Member AIAA.

and a stochastic, partially-observable aerodynamic environment including wind gusts (which effect both the bird and the perch), birds are remarkably adept at accomplishing this high-precision task.

Perching, one of the most common aerobatic maneuvers executed by birds, is representative of a large and important class of aggressive aerial maneuvers that take advantage of unsteady aerodynamics. Perching with a fixed-wing unmanned aerial vehicle (UAV) represents a formidable challenge for nonlinear controls. Following,¹ we distinguish a *dynamic* perching maneuver from V/STOL maneuvers by the use of a separation-induced pressure drag to decelerate the vehicle. Executing these dynamic maneuvers allows birds to perch in much less time and with much less energy (scaled) than man-made V/STOL equivalents. A lack of compact, design-accessible models in this flow regime (with chord Reynolds numbers exceeding 15,000) hinders the use of traditional control design techniques. However, recent advances from the machine learning community may offer a solution. Statistical learning theory provides new opportunities for unsteady system identification, and statistical approaches to approximate optimal control provide practical algorithms for developing feedback controllers. In this paper we present our initial experiments using these techniques, which have allowed us to develop a controller capable of perching a fixed-wing glider in an indoor motion capture environment.

II. Related Work

The most notable project which shares our goals is the perching plane project at Cornell.¹⁻³ The aim of this project is to design a morphing aircraft which pitches the body up to a high angle-of-attack but rotates the wings back down into the traditional envelope of attached airflow. Similarly, the tail is rotated down and out of the unsteady wake of the body in order to ensure attached flow. This is a clever mechanism for maintaining control authority, but requires a redesign of the vehicle and sacrifices performance (only a portion of the vehicle contributes to pressure drag) in order to allow the use of linear flight control techniques. Our approach is different. Here we describe the design of a *nonlinear* flight control system capable of executing the maneuver with a more conventional vehicle, exploiting pressure drag on the body and the wings.

There have also been numerous studies on control systems for UAV acrobatics. For example, the work presented in⁴ proposed an approximate inverse dynamics model for agile maneuvering where the resultant control system was tested in simulation. However, the maneuver performed was slow and well behaved, not representative of human piloted maneuvers that take the vehicle to the perimeter of its flight envelope. Work in⁵⁻⁹ all propose a linear control system for performing an autonomous prop-hang, where a fixed-wing vehicle is oriented vertically (i.e. nose up) maintaining close to zero translational velocity. These maneuvers, although impressive demonstrations, all rely on tremendous thrust-to-weight ratios (and therefore sacrifice efficiency) to dominate the dynamics of the air at high angles-of-attack.

The most notable work in autonomous UAV acrobatics has been on helicopters. Feron et al. were one of the first to achieve autonomous acrobatic flight with a hobby helicopter¹⁰. In this work an analytical dynamic model was derived from basic principles and force coefficients were measured through flight experiments. Their control system used an intuitive control approach where state machines were developed based on human pilot control data. Through use of piecewise linear commands with specified switching times, the authors were able to perform various maneuvers. Feron's subsequent work added feedback through tight angular rate controllers on the pilot's reference trajectory.^{15,16} Feron et al¹⁷ also present a control architecture for helicopters landing at unusual attitudes. In this work they use a hand designed state-machine to execute a perching maneuver that eventually lands on a large velcro launch pad.

There is also some precedent for the application of machine learning control technology to aerial acrobatics. Work by Bagnell¹⁸ describes a unique approach to helicopter control, namely a computational optimal control method, similar to the method described in this paper, known as reinforcement learning. The authors presented a control policy search algorithm for stabilizing slow moving trajectories. Later work by Ng et al.¹⁹⁻²² used a similar approach for acrobatic helicopter control. The authors first fit a stochastic nonlinear model from pilot data, i.e. a model predicting either next state or accelerations as a function of current state and control inputs. Next, a human pilot flight trajectory for a desired acrobatic maneuver was used as the baseline for an offline control policy search algorithm. Using this approach, an autonomous helicopter achieved stable inverted hover in addition to a few acrobatic maneuvers at slow speeds. We note that (with the exception of inverted hovering) Ng's and Feron's acrobatic control systems focused on stabilizing either a human pilot's trajectories or a state machine inspired by these trajectories.

III. Task Formulation

In this paper we consider the task of making a small autonomously controlled glider land on a laboratory perch. Our focus on a glider (no onboard power plant) helps to emphasize the dynamics of the task and avoids control solutions which exploit the large thrust-to-weight ratios available on some small hobby aircraft. These thrust-to-weight ratios are energetically expensive to maintain, and are not necessary for high-performance perching.

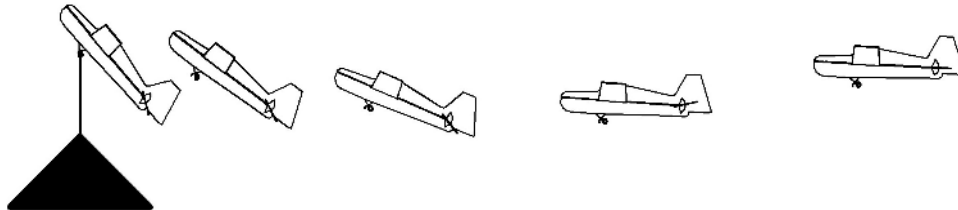


Figure 1. The perching task. This trajectory of the glider illustrated here is actually derived from real flight data of a successful perch.

Specifically, we launch a small 77 g glider with a 98mm mean chord and a 8:3 aspect ratio out of a small custom launching device towards a suspended string perch. The vehicle enters our experimental setup at approximately 6 m/s ($Re \approx 53,000$), and about 3.5 m away from the perch. The vehicle must decelerate to almost zero velocity and come to rest on the perch in less than one second. We claim that, with these specifications, the glider *must* execute a high angle-of-attack maneuver, exploiting both viscous and pressure drag to achieve the required rapid deceleration. Despite the obvious nonlinearity and complexity of this dynamic regime and the short-term loss of control authority on the stalled control surfaces, the task requirements demand a high-precision landing on the string. Figure 1 shows a feasible perching trajectory. Note the initial dip in altitude, which compensates for the boost in altitude right before stalling the wings at the end of the trajectory.

IV. Experimental Setup

IV.A. Offboard Sensing and Control

For this pilot study, we have designed an indoor experimental facility which mitigates many of the complications of sensing, computation, and repeatability. Our facility is equipped with a Vicon MX motion capture system consisting of 16 cameras, which use infrared light to track reflective markers on the vehicle. The system provides real-time sub-millimeter tracking of the vehicle and its control surface deflections, as well as the perch location. The basic setup is illustrated in Figure 2. Our system has a capture volume of approximately 27 m³ with the real-time hardware interface communicating new state information at approximately 119 frames per second. To protect the vehicle, we hang a large safety net just above the floor under the entire experiment.

The only sensing equipment onboard the vehicle is a set of small reflective markers attached to the fuselage and elevator. The glider is launched from just outside the capture volume of the Vicon system. Upon entering the volume, the real-time tracking software locks onto the plane and begins relaying position and orientation information over ethernet to the ground control station, a Linux PC running Matlab. The ground control station evaluates the control policy at 50 Hz and sends a command by serial port to the buddy-box interface of a hobby radio transmitter.

This indoor flight environment providing off-board sensing, computation, and control has proven to be an extremely effective experimental paradigm. The protective environment for the vehicle allows us to explore maneuvers well outside the typical flight regime with little risk of damaging the vehicle, and as we report in the following sections, the flight data attained from the motion capture system provides incredibly clean data for system identification.

IV.B. Vehicle Description

Our glider design (see Figure 3, left) was largely inspired by commercially available aerobatic hobby airplanes. The glider is made of laser-cut 2.8mm thick Dapron foam sheets, carries a GWS four channel micro receiver,

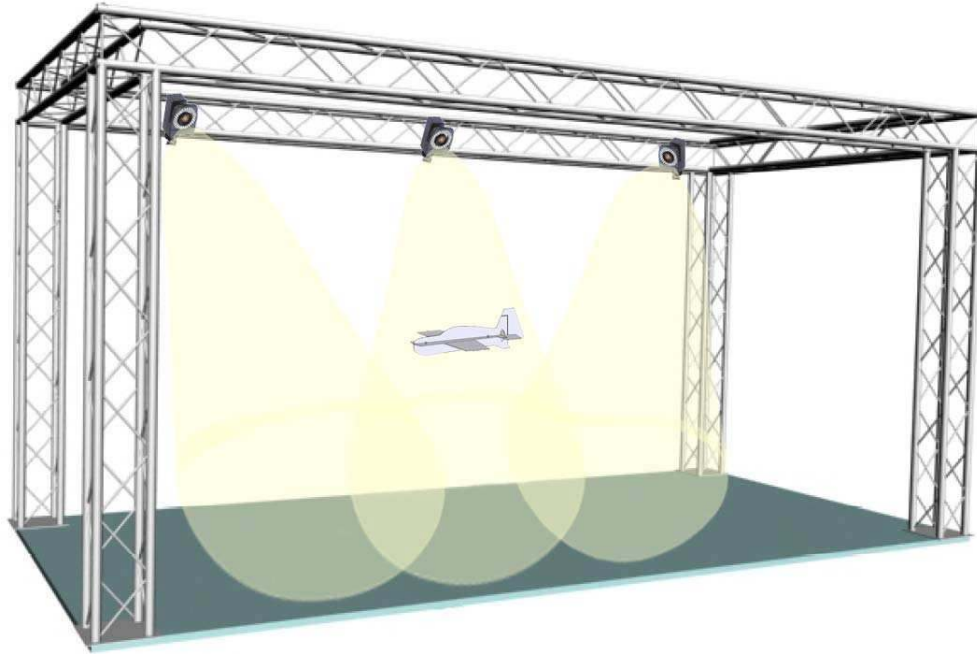


Figure 2. Vicon Motion Capture Environment. The Vicon motion capture setup makes use of 16 infrared cameras to track small reflective markers along the body and control surface of the glider. These marker positions allow the system to reconstruct the position and orientation of the glider and the control surfaces with high accuracy.

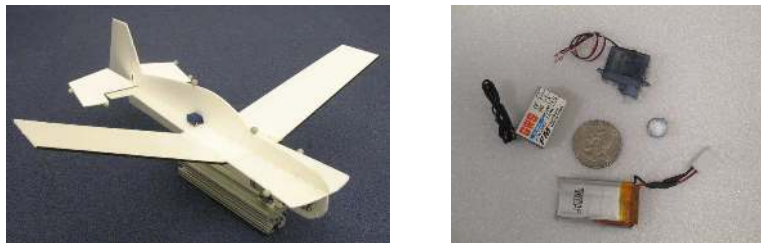


Figure 3. (left) Our experimental glider with a 20° dihedral on the wings for passive lateral stability. Reflective markers are placed along the body and the elevator in order to reconstruct absolute position and orientation using the Vicon motion capture system. **(right)** Electronics and components: (clockwise from top) hobby servo, motion capture marker, battery, micro receiver. (Quarter shown for scale).

an HS-55 Hitech hobby servo, and a Full River 250mah 2-cell lithium polymer battery (see Figure 3, right), giving a total combined weight of 77g. For ease of fabrication, the wings are a symmetric foam flat plate (i.e. no camber) with a carbon fiber reinforced leading edge having a 98mm mean chord ($\approx 3\%$ thickness-to-chord ratio), a 8:3 aspect ratio, with a maximum lift to drag ratio of approximately 3.5 (at zero elevator angle). The wings are slightly tapered (113mm root chord, 83mm tip chord), and have a 20° dihedral for passive lateral stability. The total surface area of all lifting surfaces (including the fuselage) is 0.1022m^2 for a total wing/surface loading of $0.7534 \frac{\text{kg}}{\text{m}^2}$. Four 10mm reflective markers placed along the fuselage and four on the elevator control surface allow motion capture reconstruction. The vehicle grabs the perch with a passive mechanical one-way latch mounted just below the center of mass, with a capture window of $\approx 2\text{cm}$.

The most basic trajectory of a glider landing on a perch can be sufficiently described by the glider's longitudinal dynamics. During this maneuver, these dynamics are mostly controlled by the elevator (controlling pitch) and minor corrections must be made by the ailerons and rudder for deviations outside the longitudinal plane, i.e. roll and yaw. However, given that our glider is unpropelled during flight, the extra drag induced by these control surface deflections creates a non-negligible loss in speed early in the glider's trajectory. A fixed vertical stabilizer and our wing dihedral produce similar lateral corrective forces without the added aerodynamic drag.

V. Unsteady System Identification

V.A. The Flight Data

Rather than approximating a quasi-steady model in a wind tunnel, we obtained our model directly from real flight data. Our primary interest is in the model dynamics in very high angle-of-attack (post-stall) conditions, so we hand-designed a simple feedback policy which brought the nose up and then attempted to bring the nose back down. This was accomplished by defining two deflection planes in the x -axis, the first 2.5m from the perch and the second 1.5m from the perch. Starting with a trimmed elevator angle at launch, the elevator was then deflected up to a specified angle, ψ_1^d , upon crossing the first deflection plane, and down to a different angle, ψ_2^d , upon crossing the second deflection plane. By hand-tuning the parameters, ψ_1^d and ψ_2^d , of this simple controller, we obtained good coverage of the post-stall configuration space. These trials typically ended with the plane tumbling towards the ground (e.g., in a falling-leaf mode), to be caught by the safety net.

In total, we recorded 240 flight trajectories for model identification. Data during these trials were logged at the ground station, and included only the elevator commands and the feedback from the motion capture system, which provided high-accuracy kinematic information about the trajectory of the plane, including actual elevator deflection, at approximately 119 Hz. The accuracy of each marker estimate is sub-mm, resulting in a sub-mm accuracy in position estimate and < 0.01 rad error in vehicle orientation and elevator angle. We independently estimated the vehicle mass, center of mass position, and moment of inertia (assuming that the mass of the elevator was negligible). The flight data was low-pass filtered (by a 10th order Butterworth filter with a 15Hz cut-off) and differentiated twice. Instantaneous angle-of-attack, $\alpha[n]$, lift and drag forces, $F_L[n]$ and $F_d[n]$, and pitching moment, $M[n]$, were computed directly from the model and estimated accelerations (all at time n). Finally, we estimated the instantaneous coefficients,

$$C_L[n] = \frac{2F_L[n]}{\rho V[n]^2 S} \quad C_D[n] = \frac{2F_D[n]}{\rho V[n]^2 S} \quad C_M[n] = \frac{2M[n]}{\rho V[n]^2 S c}, \quad (1)$$

where C_L , C_D , and C_M represent lift, drag, and moment coefficients, respectively, and c is the mean chord.

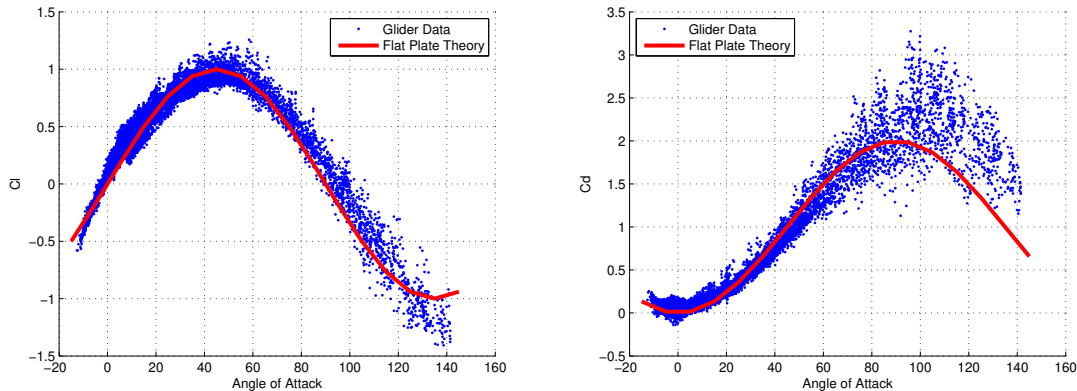


Figure 4. Scatter plots showing the general trend of the computed coefficients from flight data. Shown are lift and drag coefficients with their respective flat plate theory predictions.

The goal of identification here is to generate a predictive model of the flight coefficients that can be used to design the control system. The general trend of the coefficient data is shown in Figure 4 and Figure 5. Notice that, although data from any individual trial showed time-varying coefficients due to the vortex shedding at high angles of attack (see Figure 12), the coefficient data averaged over many trials agrees incredibly well with that predicted by flat plate theory.²³ In general, the coefficients are a function of the angle-of-attack of the wings, the elevator deflection angle, the Reynolds number, and the Mach number, as well as additional fluid-related terms in the unsteady regime. In our data, although the Reynolds number (relative to the mean chord) varies characteristically during the trajectory starting around 53,000 at launch and decreasing to around 14,000 at termination, the elevator deflection angle was the more significant variable for both prediction and control. Therefore, we model our aerodynamic coefficients as nonlinear functions of two variables - the angle attack of the wings and the deflection angle of the elevator.

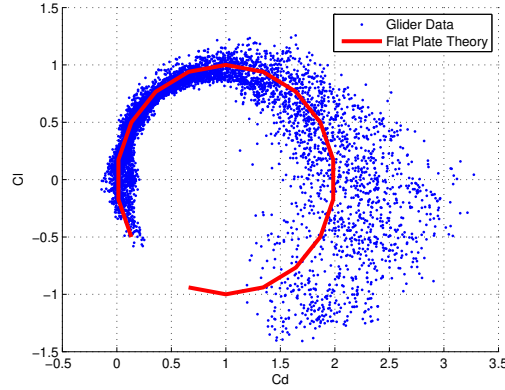


Figure 5. Lift to Drag polar plots and its respective flat plate theory prediction.

V.B. Estimating Aerodynamic Coefficients

In addition to limiting the dependence on state parameters, our model makes the additional assumption that the aerodynamic contributions from the wings and elevator are independent and additive - this approximate domain knowledge dramatically improves the generalization of the approximator outside of the flight data. We parameterize a nonlinear function approximator as a linear combination of nonlinear basis functions:

$$C_L(\alpha, \psi) = \sum_{i=1}^N w_i^\alpha v_i^\alpha(\alpha) + \sum_{i=1}^M w_i^\psi v_i^\psi(\psi) = \mathbf{w}^T \mathbf{v}(\alpha, \psi), \quad (2)$$

where w_i represents the i th parameter, v_i represents the i th nonlinear feature, and the superscript specifies whether the parameter or feature is for the angle-of-attack model or the elevator angle model. In this study, v^α and v^ψ are the features of a barycentric interpolator,²⁴ a representation commonly used in machine learning optimal control studies. For notational simplicity, these parameters and features can be collected into the vectors $\mathbf{w} = [w_1^\alpha, w_2^\alpha, \dots, w_N^\alpha, w_1^\psi, \dots, w_M^\psi]^T$ and $\mathbf{v}(\alpha, \psi) = [v_1^\alpha(\alpha), \dots, v_1^\psi(\psi), \dots]^T$. The model parameter vector, \mathbf{w} , was found using least-squares ridge-regression to minimize the error,

$$\min_{\mathbf{w}} \left[\sum_{\text{trials}} \sum_n [C_L[n] - \mathbf{w}^T \mathbf{v}(\alpha[n], \psi[n])]^2 + \lambda \mathbf{w}^T \mathbf{w} \right]. \quad (3)$$

Ridge regression shrinks the parameters by imposing a penalty on their size, eliminating large parameter magnitudes in regions where data is sparse (typically at the boundaries of the approximation). This regression was combined with a recursive outlier detection scheme that removed data points outside five standard deviations of the current best fit; the model was re-fit until no outliers were found. The procedure described here for C_L was then repeated (using the same parameters) for C_D and C_M . Figures 6 through 8 show the approximated lift and drag coefficient models with overlays of the actual data points. Note that the domain assumption of additive and independent contributions from wing and elevator allowed us to effectively generalize the model beyond the configurations found in the training data.

Figures 6 and 7 suggest that the lift and drag coefficients of our glider do not heavily depend on elevator angle and can be sufficiently described as a function of angle-of-attack alone. Moreover, the comparisons in Figure 4 suggests that flat-plate theory provides sufficient (and surprisingly accurate) models of the lift/drag characteristics of our glider and can be written in closed form as²³

$$C_L = 2 \sin(\alpha) \cos(\alpha) \quad (4)$$

$$C_D = 2 \sin^2(\alpha) \quad (5)$$

Figure 9 shows the lift to drag characteristics of our glider. Note that because our design was largely inspired by that of an aerobatic propeller driven model airplane, our lift to drag ratio is low, having a maximum value of approximately 4.5. This was completely satisfactory for our perching experiments, requiring high maneuverability and short glide distances.

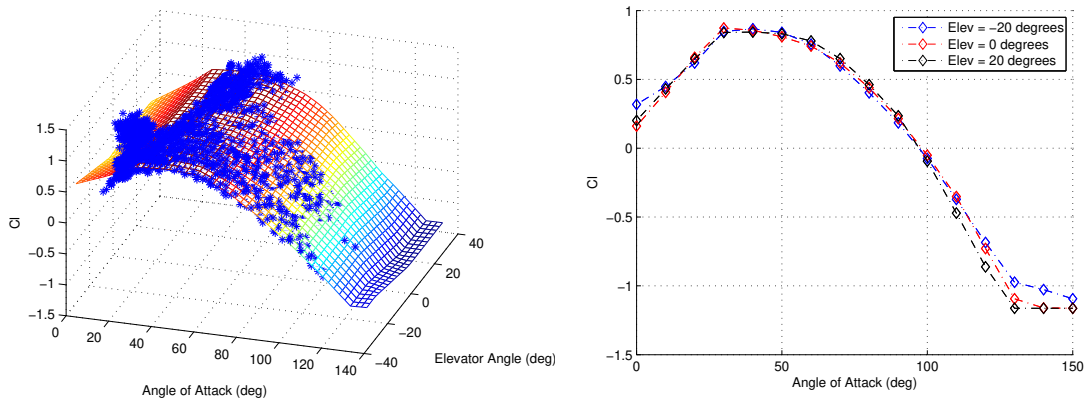


Figure 6. (left) Learned lift coefficient model overlaid with real flight data points. Note that the axes are atypical; they are the two inputs to our coefficient model, angle-of-attack and elevator deflection. (right) 2D projection of lift curves for varying elevator angles.

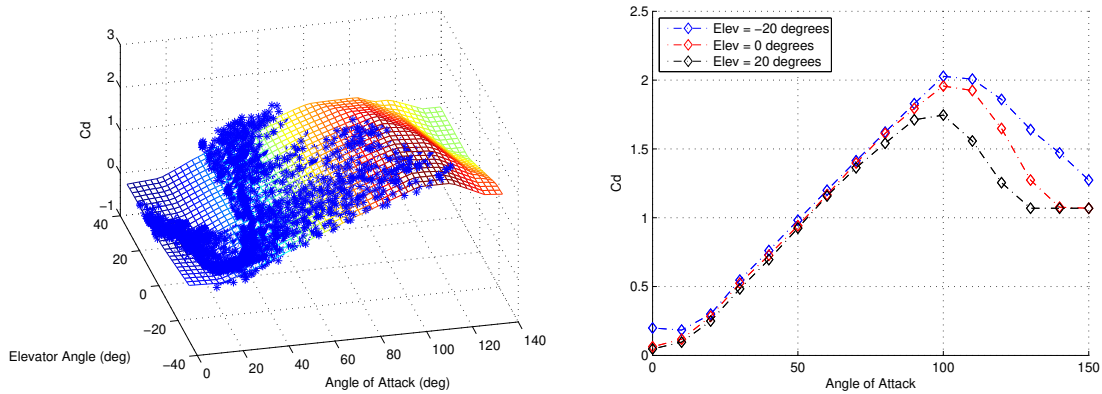


Figure 7. (left) Learned drag coefficient model overlaid with real flight data points. (right) 2D projection of drag coefficient curves for varying elevator angles.

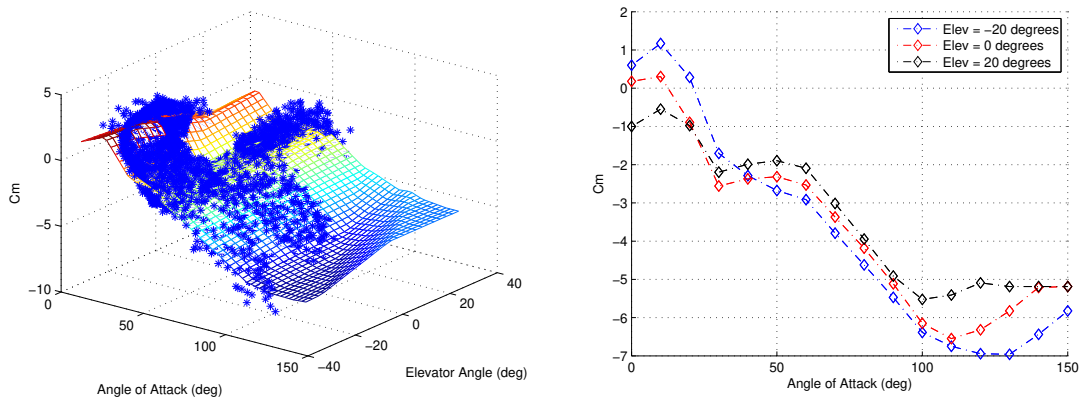


Figure 8. (left) Learned moment coefficient model overlaid with real flight data points. (right) 2D projection of moment coefficient curves for varying elevator angles.

V.C. The Complete Model

In addition to the nonlinear coefficient model, we identified the parameters of a nonlinear rigid-body model of the vehicle, and an additional linear model to account for the dynamics of the elevator actuator and control loop delay. The actuator model was estimated from step-response experiments with the glider held

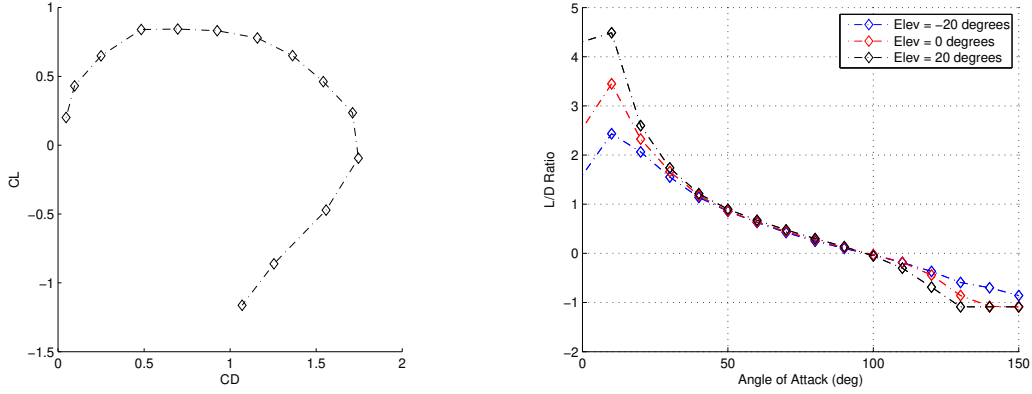


Figure 9. (left) Polar plot for C_l vs. C_d . (right) $\frac{L}{D}$ ratio.

stationary in the motion capture space as a fixed delay followed by a second-order auto-regression model. Identification revealed a 35 ms loop delay (from step command at control station to onset of movement at the control surface as communicated by the motion capture system) - a consequence of the Vicon processing and sequential communication channels.

VI. Approximate Optimal Feedback Control

Solving the glider perching control task requires a controller design procedure which is capable of reasoning about the long-term nonlinear dynamics of the aircraft, the effects of control delay, and the intermittent losses of control authority. Furthermore, because the system has no potential for energy input (no propeller), it is not possible to stabilize a trajectory of the glider's center of mass; corrective control actions increase drag and decrease energy. Therefore, we require a robust feedback policy over the state space. An optimal feedback control formulation of the problem is sufficient to describe a solution to these challenges, but closed-form solutions to nonlinear optimal control problems are rarely obtainable in practice. Fortunately, recent advances in approximate numerical solutions²⁵⁻²⁷ make it possible to efficiently design approximately optimal controllers for relatively low-dimensional problems like the perching problem we have formulated here.

VI.A. Optimal Control Formulation

We formulate the perching problem as an infinite-horizon optimal feedback control problem. Let us define the full-state vector of the system, $\mathbf{x} = [x, z, \theta, \psi, \dot{x}, \dot{z}, \dot{\theta}, \dot{\psi}]^T$. The goal of the controller design procedure is to design a full-state feedback policy,

$$u[n] = \pi(\mathbf{x}[n]), \quad (6)$$

where u is the command sent to the elevator, which is limited by saturation to the (scaled) interval $[-1, 1]$, and $\pi : \mathbb{R}^6 \rightarrow \mathbb{R}$. Together with the dynamic model described in the last section, which we summarize with the discrete-time equation,

$$\mathbf{x}[n+1] = f(\mathbf{x}[n], u[n]), \quad (7)$$

the feedback policy completely describes the closed-loop dynamics of the system.

In order to steer the vehicle as close as possible to the desired state, \mathbf{x}^d , the perching configuration, we define the infinite-horizon cost-to-go,

$$J^\pi(\mathbf{x}) = \min_{n=0, \dots, \infty} [(\mathbf{x}[n] - \mathbf{x}^d)^T \mathbf{Q}(\mathbf{x}[n] - \mathbf{x}^d)], \text{ s.t. } \mathbf{x}[0] = \mathbf{x}, \quad \mathbf{x}[n+1] = f(\mathbf{x}[n], \pi(\mathbf{x}[n])). \quad (8)$$

The goal of the numerical optimal control algorithm is to find the feedback policy, π , which minimizes this cost-to-go for all \mathbf{x} .

VI.B. A Value Iteration Algorithm

The cost-to-go function described above does not have the standard additive cost structure which is so often exploited in optimal control algorithms.²⁶ However, a solution to this cost function can be approximated with equally simple algorithms by noting the recurrence relation:

$$J^\pi(\mathbf{x}) = \min [(\mathbf{x}[n] - \mathbf{x}^d)^T \mathbf{Q}(\mathbf{x}[n] - \mathbf{x}^d), \quad J^\pi(\mathbf{x}')], \quad \mathbf{x}' = f(\mathbf{x}, \pi(\mathbf{x})). \quad (9)$$

Note that the ‘min’ in this equation returns the minimum value of the two arguments. Similarly, the optimal policy, π^* and optimal cost-to-go, J^* , subscribe to the relation:

$$J^*(\mathbf{x}) = \min [(\mathbf{x}[n] - \mathbf{x}^d)^T \mathbf{Q}(\mathbf{x}[n] - \mathbf{x}^d), \quad J^*(\mathbf{x}')], \quad \mathbf{x}' = f(\mathbf{x}, \pi^*(\mathbf{x})). \quad (10)$$

By discretizing the state space, this relation provides a recursive algorithm which, for any initial guess $\hat{J}^*(\mathbf{x})$, will converge to the optimal solution $J^*(\mathbf{x})$ of the discretized system. We perform this discretization, and the interpolation back into a continuous full-state policy, using another barycentric function approximator. In order to be computationally tractable, the discretization in this high-dimensional space must be very coarse; we use 15 bins on x and z , and 8 bins each on θ , \dot{x} , \dot{z} , and $\dot{\theta}$. We assume that the current state of the elevator, ψ and $\dot{\psi}$, do not have a major impact on the optimal feedback policy.

VI.C. Coping With Delay

In fact, the dynamics of the system are not quite as ideal as described in the control derivation above. Recall that our linear actuator model also takes into account the feedback delay in the system. Naively, this delay will increase the number of state variables for our optimal control algorithm. However, we can take advantage of the structure of this pure delay, which happens to be 4 in our discrete-time model, by observing that the delayed dynamics can be written as

$$\mathbf{x}[n+1] = f(\mathbf{x}[n], u[n-4]). \quad (11)$$

Therefore, if we solve the optimal control equations above, assuming no delay, we can (through a change of variables) write the optimal feedback policy as

$$u[n] = \pi^*(\mathbf{x}[n+4]). \quad (12)$$

Thanks to the high-fidelity of our model, we can execute this policy on the real plane by simulating our state estimate forward in time:

$$\hat{\mathbf{x}}[n+4] = f_4(\mathbf{x}[n], u[n-4], u[n-3], u[n-2], u[n-1]), \quad (13)$$

where f_4 is the 4-step dynamics of f .

VI.D. Simulation Results

Our value iteration control optimization was able to successfully reproduce the perching maneuver in simulation. Figure 10 shows the translational and pitch trajectories of the glider during the execution of the maneuver.

VII. Experimental Results

Our preliminary experiments using simulation-designed policies on the real aircraft have produced exciting results. Figure 11 shows frames from a high-speed image sequence from a successful perch, playing control trajectories from our optimal control solution. Based on our initial controller design procedure, which used a fairly coarse resolution mesh over the high-dimensional state-space (due to memory constraints), the perching maneuver has a relatively small basin of attraction; the launcher put the plane into this basin about 1 in every 5 trials, resulting in a successful perch. The model was able to accurately predict, based on only initial conditions, whether the perch would be successful, and unsuccessful trajectories only missed the perch by a very small margin; recall that we designed a capture window of $< 2\text{cm}$, in order to focus our attention on high-precision maneuvers.

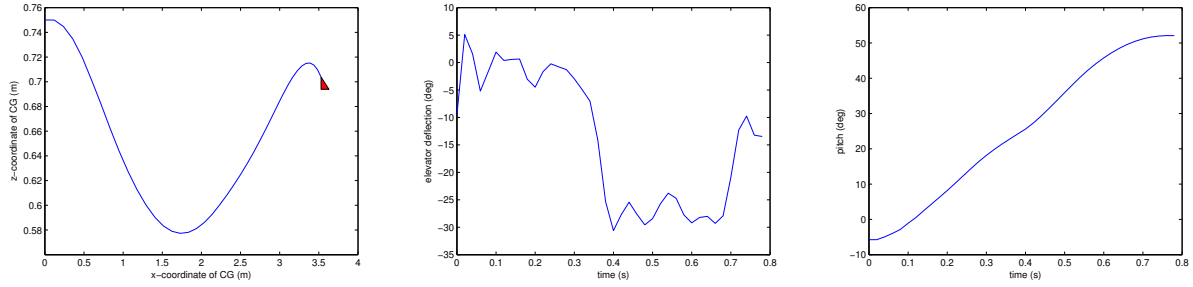


Figure 10. (left) The simulated trajectory of the perching maneuver starting at the nominal condition of 6.0m/s 3.5m away from the perch (in red). (middle) The elevator command policy over the perching trajectory. (right) The corresponding simulated pitch trajectory for the perching task.

VIII. Discussion

The results presented here are surprising in a number of ways. First, we were able to estimate a very high fidelity flight model, which included post-stall configurations, using only kinematic flight data from our motion capture environment. In fact, close inspection of the time trajectories of C_L and C_D reveals a clear oscillation in the high angle-of-attack configuration at around 11 Hz (see Figure 12). This frequency is in close agreement with the expected vortex shedding frequency predicted by flat plate theory.²⁸ There is no question that a vortex shedding is occurring in these trajectories, but the authors did not expect to find this evidence in the differentiated kinematic data!

This work also demonstrates the potential for approximate optimal control solutions to generate high-performance control systems for unsteady flight conditions. The optimal control formulation specifically accounts for the long-term dynamics of the aircraft, and can “reason” through intermittent losses of control authority during a stall. Recent advances in computational optimal control algorithms, in conjunction with a number of domain specific simplifications, made it possible to compute a coarse feedback policy with these same long-term control characteristics.

Finally, this work demonstrates the importance of dynamics in high-performance control. Unlike feedback-linearization control design, which attempts to cancel out nonlinear dynamics with high-gain feedback, the optimal control designed here *exploits* the nonlinear dynamics of the plant in order to execute the perching maneuver. Indeed, it is precisely the separation at the wing which creates the pressure drag needed for rapid deceleration.

IX. Conclusions

Here we have presented the modeling and control of a small autonomous glider which can land on a perch. The success in this task can be attributed to an experimental setup which simplified both the vehicle and the task. Our modeling and control processes made further use of simplifications and statistical approximations in order to, ultimately, generate a feedback policy which mapped motion capture information about the state of the plane into a command sent to the elevator actuator.

Perching is representative of a large and exciting class of dynamic maneuvers which will involve a delicate interaction with unsteady flow regimes. Mastery of these flow regimes will dramatically increase the capabilities of UAV technology.

Acknowledgments

The authors would like to acknowledge labmates John Roberts, Zack Jackowski, and Jim Holzberger for their support with the experiments. This research was supported by MIT, MIT CSAIL, and NEC.

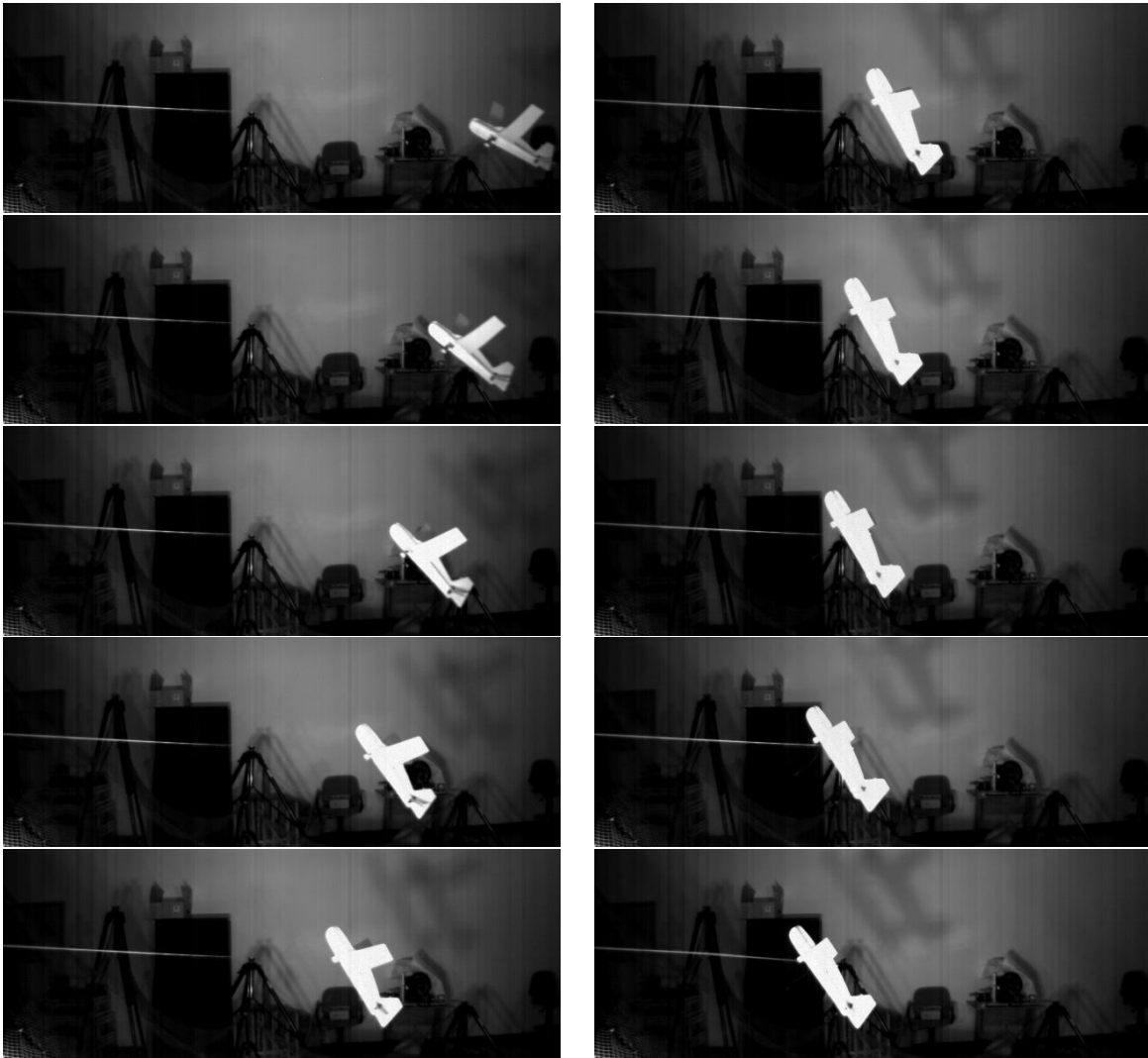


Figure 11. Stills from a successful perching trajectory

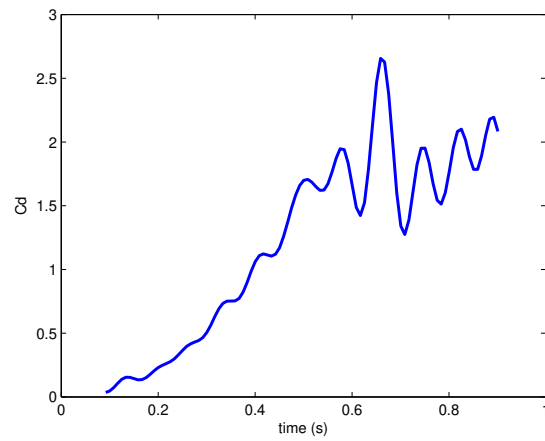


Figure 12. A sample time trajectory of the drag coefficient revealing a clear carrier oscillation around 11 Hz. This may be an artifact of vortex shedding.

References

- ¹Wickenheiser, A. M. and Garcia, E., “Longitudinal Dynamics of a Perching Aircraft,” *Journal of Aircraft*, Vol. 43, No. 5, 2006, pp. 1386–1392.
- ²Wickenheiser, A., Garcia, E., Waszak, and M., “Longitudinal dynamics of a perching aircraft concept,” *Proc. SPIE - Int. Soc. Opt. Eng. (USA)*, Vol. 5764, No. 1, 2005, pp. 192 – 202.
- ³Wickenheiser, A. M. and Garcia, E., “Optimization of Perching Maneuvers Through Vehicle Morphing,” *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 4, July-August 2008, pp. 815–824.
- ⁴Jang, J. S. and Tomlin, C. J., “Autopilot Design for the Stanford DragonFly UAV: Validation through Hardware-in-the-Loop Simulation,” 2001.
- ⁵Green, W. E. and Oh, P. Y., “A MAV That Flies Like an Airplane and Hovers Like a Helicopter,” July 2005.
- ⁶Green, W. E. and Oh, P. Y., “A Fixed-Wing Aircraft for Hovering in Caves, Tunnels, and Buildings,” June 2006.
- ⁷Cory, R. and Tedrake, R., “On the controllability of Agile Fixed-Wing Flight,” *Proceedings of the 2007 Symposium on Flying Insects and Robots (FIR)*, August 2007.
- ⁸Frank, A., McGrew, J. S., Valenti, M., Levine, D., and How, J., “Hover, Transition, and Level Flight Control Design for a Single-Propeller Indoor Airplane,” *AIAA Guidance, Navigation and Control Conference and Exhibit*, AIAA, 2007.
- ⁹Blauwe, H. D., Bayraktar, S., Feron, E., and Lokumcu, F., “Flight Modeling and Experimental Autonomous Hover Control of a Fixed Wing Mini-UAV at High Angle of Attack,” *AIAA Guidance, Navigation and Control Conference and Exhibit*, 2007.
- ¹⁰Piedmonte, M. and Feron, E., “Aggressive Maneuvering of Autonomous Aerial Vehicles: A Human-Centered Approach,” October 1999.
- ¹¹Gavrilets, V., Frazzoli, E., Mettler, B., Piedmonte, M., and Feron, E., “Aggressive Maneuvering of Small Autonomous Helicopters: A Human-Centered Approach,” *The International Journal of Robotics Research*, 2001.
- ¹²Sprague, K., Gavrilets, V., Dugail, D., Mettler, B., Feron, E., Martinos, and I., “Design and applications of an avionics system for a miniature acrobatic helicopter,” October 2001.
- ¹³Gavrilets, V., Mettler, B., and Feron, E., “Nonlinear Model for a Small-Size Acrobatic Helicopter,” August 2001.
- ¹⁴Gavrilets, V., Mettler, B., and Feron, E., “Dynamic Model for a Miniature Aerobatic Helicopter,” .
- ¹⁵Gavrilets, V., Martinos, I., Mettler, B., and Feron, E., “Flight test and simulation results for an autonomous aerobatic helicopter,” 2002.
- ¹⁶Gavrilets, V., Martinos, I., Mettler, B., and Feron, E., “Control Logic for Automated Aerobatic Flight of a Miniature Helicopter,” August 2002.
- ¹⁷Bayraktar, S. and Feron, E., “Experiments with small helicopter automated landings at unusual attitudes,” *arXiv*, September 2007.
- ¹⁸Bagnell, J. A. and Schneider, J. G., “Autonomous Helicopter Control Using Reinforcement Learning Policy Search Methods,” May 2001.
- ¹⁹Ng, A. Y., Kim, H. J., Jordan, M. I., and Sastry, S., “Autonomous helicopter flight via Reinforcement Learning,” *Advances in Neural Information Processing Systems (NIPS)*, Vol. 16, 2003.
- ²⁰Ng, Y., A., Coates, A., Diel, M., Ganapathi, V., Schulte, J., Tse, B., Berger, E., Liang, and E., “Autonomous inverted helicopter flight via reinforcement learning,” *International Symposium on Experimental Robotics*, 2004.
- ²¹Abbeel, P., Ganapathi, V., and Ng, A. Y., “Learning vehicular dynamics, with application to modeling helicopters,” *NIPS*, 2006.
- ²²Abbeel, P., Coates, A., Quigley, M., and Ng, A. Y., “An Application of Reinforcement Learning to Aerobatic Helicopter Flight,” *Proceedings of the Neural Information Processing Systems (NIPS '07)*, Vol. 19, December 2006.
- ²³Tangler, J. and Kocurek, J. D., “Wind Turbine Post-Stall Airfoil Performance Characteristics Guidelines for Blade-Element Momentum Methods,” *43rd AIAA Aerospace Sciences Meeting and Exhibit*, AIAA, 2005.
- ²⁴Munos, R. and Moore, A., “Variable Resolution Discretization in Optimal Control,” *Machine Learning*, Vol. 49, No. 2/3, November/December 2002, pp. 291–323.
- ²⁵Sutton, R. S. and Barto, A. G., *Reinforcement Learning: An Introduction*, MIT Press, 1998.
- ²⁶Bertsekas, D. P., *Dynamic Programming and Optimal Control*, Athena Scientific, 2nd ed., 2000.
- ²⁷Tedrake, R. L., *Applied Optimal Control for Dynamically Stable Legged Locomotion*, Ph.D. thesis, Massachusetts Institute of Technology, 2004.
- ²⁸Lam, K. and Leung, M., “Asymmetric vortex shedding flow past an inclined flat plate at high incidence,” *European Journal of Mechanics B/Fluids*, Vol. 24, 2005.